

IDENTIFICATION

PRODUCT CODE I	MAINDEC-8E-DBCC-0
PRODUCT NAME I	8E ADDER TESTS
DATE CREATED I	SEPT. 1, 1971
MAINTAINER I	DIAGNOSTIC GROUP
AUTHOR I	M. DAVIS & J. VROBEL

COPYRIGHT © 1971
DIGITAL EQUIPMENT CORPORATION



1. ABSTRACT

THIS PROGRAM TESTS THE ADDER CIRCUITS OF THE PDP-8E. THE PROGRAM IS COMPOSED OF FIVE 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 BSW.

A CARRY GENERATION TEST

A SERIES OF RANDOM NUMBER TESTS

A FIELD RELOCATION ADDER TEST

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 0000-6000 AND UTILIZES LOCATIONS 7775-7777 AS A TEST AREA.

2.3 PRELIMINARY PROGRAMS

MAINDEC=8E-D0AA, 00BA

RUN ALL EXTENDED MEMORY TESTS PRIOR TO RUNNING RELOCATION ADDER TEST.

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
SR04=0 LOOP IN CURRENT MEMORY BANK
SR04=1 RELOCATE TO NEXT EXISTING BANK
SR06=0B AMOUNT OF EXTENDED BANKS OF MEMORY
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 SR=0000

4.3.4 SET SWITCH REGISTER TO DESIRED FUNCTIONS SEE 4.1

4.3.5 PRESS CLEAR AND CONT SWITCHES

5. OPERATING PROCEDURE

5.1 FASI 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 7600 AND START COMPUTER.

5.3 RELOCATION ADDER TEST

IF SR04=1 THE ADDER TEST WILL RELOCATE TO THE NEXT SEQUENTIAL EXISTING MEMORY BANK AT THE COMPLETION OF EVERY PASS, THE EXACT AMOUNT OF EXISTING EXTENDED MEMORY BANKS MUST BE IN SR06=00 TO RUN THIS PORTION OF THE ADDER TEST, PRIOR TO EACH RELOCATION THE PROGRAM WILL COMPARE THE BANKS FOUND UNDER TEST TO THE BANK AMOUNT IN SR06=00 AND START RELOCATION, THE FOLLOWING MESSAGE WILL BE TYPED ON TELETYPE.

***** X EXTENDED BANKS OF MEMORY TO BANK X *****

5.4 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
ARG2 SIMULATED ARG1+ARG2 ARG2+ARG1
XXXXXXXXXXXXX XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX

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

1+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.

6.1.1.2 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 XXXXXXXXXXXXXXXX X XXXXXXXXXXXXXXXX

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

6.1.1.3 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 XXXXXXXXXXXXXXXX

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

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

6.1.1.4 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

6.1.1.5 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

AAA=RAR, 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
5061	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 IYPEOUT
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

EXTENDED MEMORY TESTS SHOULD BE RUN PRIOR TO
RUNNING RELOCATION ADDER TEST.

8. EXECUTION TIME

TIME DEPENDENT ON AMOUNT OF MEMORY, FOR EACH BANK APPROXIMATELY 35 MINUTES, IF SR03=1, AND KXXXX=7777(SEE 5.1) 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
SIMROY
FACT
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 BSM 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,

9.4

RANDOM ADD TEST 1

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

9.5

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.

9.6

RANDOM ROTATE TEST

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

9.6

RELOCATION ADDER TEST

ALL TESTS LISTED ABOVE ARE RELOCATED TO EXTENDED BANKS AND RUN.

10:

LISTING

/ADDER TEST
 /FOR PDP-8/E
 /COPYRIGHT 1970 DIGITAL EQUIPMENT CORP, MAYNARD MASS,
 /V 82 07552

/INSTRUCTION DEFINITIONS

7501 MOA=7501
 7421 MOL=7421
 7002 OSH=7002
 6007 CAF=6007

/SWITCH REGISTER MASK BITS

0103 SR00=K4000
 0104 SR01=K2000
 0105 SR02=K1000
 0106 SR03=K0400
 0107 SR04=K0200
 0110 SR05=K0100
 0111 SR06=K0040
 0112 SR07=K0020
 0113 SR08=K0010
 0114 SR09=K0004
 0115 SR10=K0002
 0116 SR11=K0001

/LOCATION EQUIVALENCIES

0023 RAC=ARG1
 0024 RLNK=ARG2
 0031 RRAC=SUM1
 0033 RRLNK=SUM2
 0025 TEMPAC=SIMAC
 0026 TEMPL=SIMLNK
 0037 TEMPI=END1
 0040 W1=WD1
 0040 W2=WD2
 0035 RHFLG=AHFLG
 0067 NERROP=XLOOP
 7775
 7775 TSTAB; 0
 7776 TSTA1; 0
 7777 TSTA2; 0
 0000
 0000 TSTA3; 0
 5001 TSTA4; JMP
 0002 TSTA5; 2
 0003 TSTA6; 3
 0004 TSTA7; 0

/AC TO BE ROTATED
 /LINK TO BE ROTATED
 /AQ AFTER REAL ROTATE
 /LINK AFTER REAL ROTATE
 /TEMPORARY AC STORAGE
 /TEMPORARY LINK STORAGE
 /TEMPORARY DATA STORAGE
 /
 /
 /
 /ROTATE TEST ERROR HEADER FLAG

```

0010 *10
/INDEX REGISTERS
/
0010 0000 TSTIND, 0
0011 0000 POINT1, 0
0012 0000 POINT2, 0

```

```

0020 *20
0020 0000 CNTRI, 0
0021 0022 ADA2
0022 7777 ADA1, 7777

```

```

/SIMULATION VARIABLES
/
0023 0000 ARG1, 0
0024 0000 ARG2, 0
0025 0000 SIMAG, 0
0026 0000 SIMLNK, 0
0027 0000 AJORAZ, 0
0030 0000 CARRY, 0
0031 0000 SUM1, 0
0032 0000 LINK1, 0
0033 0000 SUM2, 0
0034 0000 LINK2, 0

```

```

/MESSAGE OUTPUT VARIABLES
/
0035 0000 AMPLG, 0
0036 0000 CHAR, 0
0037 0000 HQ1, 0
0040 0000 HQ2, 0

```

```

/RANDOM VARIABLES
/
0041 0037 RANDA, 37
0042 0000 RANDB, 0
0043 0000 RANDC, 0
0044 0000 LINKR, 0
0045 0000 LINKRC, 0

```

```

/INDIRECT POINTERS
/
0046 1600 XPRINT, PRINT
0047 1652 XIYPE, TYPE
0050 1133 XRHD, RHD
/CHARACTER STRING TYPE
/CHARACTER TYPE
/TYPE ROTATE ERROR HEADER

```

```

0051 1200 XSROT, SROTAL
0052 0756 XRALTA, RALTAB=1
0053 1157 XRTLTA, RLTAB=1
0054 1140 XRTRTA, RTRTAB=1
0055 1657 XBSHTA, BSHTAB=1
0056 1000 XCOMRO, COMROT
0057 1031 XNXTRO, NXTROT
0060 0504 XLNKOU, LNKOUT
0061 0523 XWDOUT, WDOUT
0062 3000 XAMEAS, SAMEAS
0063 3780 XAMEA, SAMEA
0064 3017 XAVREG, SAVREG
0065 3037 XDATER, DATER
0066 3027 XHALT2, HALT2
0067 3046 XLOOP, LOOP
0070 7775 XSTAB, TSTAB
0071 7776 XSTA1, TSTA1
0072 7777 XSTA2, TSTA2
0073 3512 XRAND, RANDOM
0074 0410 XLOOP2, HLTA=4
0075 0952 XLOOP1, LOOP1

```

/RANDOM NUMBER GENERATOR

/COMMON ROTATE SIMULATOR
/RAL MASK TABLE
/RTL MASK TABLE
/RTR MASK TABLE
/BYTE SWAP MASK TABLE
/ROTATE COMPARISON FOR SIMULATION
/ROTATE SETUP FOR SIMULATION
/TYPE LINK
/TYPE DATA WORD
/COMPARE DATA
/SAVE AC AND LINK
/DATA ERROR HANDLER FOR FCT
/DATA ERROR HALT FOR FCT
/LOOP ON TEST

/WIDELY USED CONSTANTS

```

0076 0240 K240,
0077 0260 K260,
0090 0261 K261,
0101 6000 K6000,
0102 0102 XRARTA,
0103 4000 K4000,
0104 2000 K2000,
0105 1000 K1000,
0106 0400 K0400,
0107 0200 K0200,
0110 0100 K0100,
0111 0040 K0040,
0112 0020 K0020,
0113 0010 K0010,
0114 0004 K0004,
0115 0002 K0002,
0116 0001 K0001,
0117 0000
0120 4000
0121 0001

```

/TEST POINTERS FOR FCT

```

0122 2004 SEQ1,
0123 2043 SEQ2,
0124 2076 SEQ3,
0125 2200 SEQ4,
0126 2232 SEQ5,
FCT1
FCT2
FCT3
FCT4
FCT5

```

0127	0130	0131	0132	0133	0134	0135
SEQ6,	SEQ7,	SEQ8,	SEQ9,	SEQ10,	SEQ11,	SEQ12,
FCT6	FCT7	FCT8	FCT9	FCT10	FCT11	FCT12

SETUP INSTRUCTIONS FOR FCI

0136	0137	0140	0141	0142	0143	0144	0145	0146	0147	0150	0151	0152	0153	0154	0155
INS1,	INS3,	INS4,	INS5,	INS6,	INS7,	INS8,	INS9,	INS10,	INS11,	INS12,	INS13,	INS14,	INS15,	SEG,	BIN,
1376	7001	5404	5402	7070	2376	2000	2410	4000	4776	4410	5403	5401	4377	FCT1	5301
BTAD	BTAC	BUMP	BUM	BCHA	BISZ	BISZ	BISZ	BJMS	BJMS	BJMS	BJMP	BJMP	BJMS		
01 IN 7777		02 IN 0002	01 IN 0001	CML RAR	01 IN 7777	01 IN 7777	TSTIND	01 IN 7777	01 IN 7777	TSTIND	01 IN 0002	01 IN 0000	01 IN 7777		

TEST FOR FAST TAD SIMULATION

0156	0157	0160	0161	0162	0163	0164	0165	0166	0167	0170	0171	0172	0173	0174	0175	0176	0177	
CAP	LAS	AND	SNA	JMP	CLA	AND	DCA	JMP	RSIMAD	KXXXX	K0	K0007	K0070	FLDNUM	FLDSAV	FLDCNT	*177	GOTEST
6007	7604	0106	7650	5177	7240	0170	3024	5567	0202	0000	0000	0007	0070	0000	0000	0000	0177	7410
SR03	GOTEST	KXXXX	ARG2	01				02										

TEST FOR FAST TAD SIMULATION

0177	0178	0179	0180	0181	0182	0183	0184	0185	0186	0187	0188	0189	0190	0191	0192	0193	0194	0195	0196	0197	0198	0199	
BTAD	BTAC	BUMP	BUM	BCHA	BISZ	BISZ	BISZ	BJMS	BJMS	BJMS	BJMP	BJMP	BJMS										
01 IN 7777		02 IN 0002	01 IN 0001	CML RAR	01 IN 7777	01 IN 7777	TSTIND	01 IN 7777	01 IN 7777	TSTIND	01 IN 0002	01 IN 0000	01 IN 7777										

TEST FOR FAST TAD SIMULATION

```

0200 0200 JMP START /GO TO FAST TEST CHECK
0201 0201 DCA ARG2 /CLEAR SIMULATION VARIABLES
0202 0202 DCA ARG1 /CLEAR ERROR MESSAGE FLAG
0203 0203 DCA AHFLG

```

```

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

```

```

/FORM OR OF ARG1 WITH ARG2

```

```

SIMAD,
0204 7340 CLA CLL CMA /LOAD AC WITH ARG1
0205 0023 AND ARG1 /PLACE IN MQ
0206 7421 MQL /LOAD AC WITH ARG2
0207 7040 CMA AND ARG2 /FORM ARG1 OR ARG2
0210 0024 AND ARG2 /SAVE ARG1 OR ARG2
0211 7501 MQA A10RA2
0212 5027 DCA

```

```

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

```

```

0213 7501 MQA /GET ARG1 FROM MQ
0214 7040 CMA /FORM NOTARG1
0215 0024 AND ARG2 /AND WITH ARG2 TO GET ARG2 AND NOTARG1
0216 7421 MQL /SAVE IN MQ
0217 7040 CMA /LOAD AC WITH ARG2
0220 0024 AND ARG2 /FORM NOTARG2
0221 7040 CMA /AND WITH ARG1 TO GET ARG1 AND NOTARG2
0222 0023 AND ARG1 /OR WITH ARG2 AND NOTARG1
0223 7501 MQA /TO GET ARG1 XOR ARG2
0224 5025 DCA SIMAC
0225 5026 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

```

```

0226 7040 CMA /LOAD AC WITH ARG1
0227 0023 AND ARG1 /AND WITH ARG2
0230 0024 AND ARG2 /ARE THERE ANY CARRIES
0231 7450 SNA ADD /NO, TERMINATE SIMULATION
0232 5274 JMP
/GENERATE CARRIES

```

```

0233 7421 MQL /SAVE FIRST CARRIES
0234 7521 MQA MQL /GET CARRIES FROM MQ
0235 0027 AND A10RA2 /AND WITH A10RA2 TO SEE IF MORE CAPRIES ARE GENERATED

```

```

0236 7450 SNA /ARE THERE ANY MORE CARRIES
0237 5244 JMP ENCAR /NO, END SIMULATION OF CARRIES
0240 7104 CLL RAL /PROPIGATE CARRIES
0241 7521 MQA MQL /GET PREVIOUS CARRIES FROM MQ, SAVE NEW CARRIES
0242 7501 MQA /OR NEW CARRIES WITH PREVIOUS CARRIES
0243 5234 JMP NXTCAR /CONTINUE

```

```

/TEST FOR CARRY INTO LINK
/
ENCAR,
0244 7501 MQA /GET CARRIES
0245 0027 AND A10RA2 /AND WITH A10RA2
0246 0103 AND K4000 /TEST BIT 00
0247 7450 SNA /IS BIT 00 1
0250 5253 JMP ENCAR1 /NO, CARRIES DID NOT PROPAGATE INTO LINK
0251 3026 DCA SIMLNK /YES, SAVE CARRY INTO LINK
0252 5260 JMP XORALL /COMPLETE SIMULATION
0253 7130 CLL CHL RAR /SET AC=4000
0254 0023 AND ARG1 /AND WITH ARG1
0255 0024 AND ARG2 /AND WITH ARG2 TO SEE IF ORIGINAL
0256 7440 SZA /NUMBERS GENERATED CARRY INTO LINK
0257 3026 DCA SIMLNK /SAVE SIMULATED LINK

```

```

/FORM XOR OF ARG1, ARG2 AND CARRIES
/TO GET FINAL SIMULATED SUM
/
XORALL,
0260 7501 MQA /SAVE SIMULATED CARRIES
0261 3030 DCA CARRY
0262 7501 MQA
0263 7040 CMA
0264 0025 AND SIMAC /FORM A10RA2 AND NOTCARRY
0265 7421 MQL /SAVE IN MQ
0266 7040 CMA
0267 0025 AND SIMAC
0270 7040 CMA
0271 0030 AND CARRY /FORM CARRY AND NOTA10RA2
0272 7501 MQA /OR WITH CONTENTS OF MQ
0273 3025 DCA SIMAC /TO GET FINAL SIMULATED SUM

```

```

ADD,
0274 7340 CLA CLL CMA /PERFORM ADDITIONS ARG1+ARG2 AND ARG2+ARG1
0275 0023 AND ARG1 /LOAD AC WITH ARG1
0276 1024 TAD ARG2 /ADD ARG2
0277 7000 NOP
0300 3031 SUM1 /SAVE RESULT
0301 7010 RAR
0302 3032 DCA LINK1 /SAVE LINK
0303 7040 CMA
0304 0024 AND ARG2 /LOAD AC WITH ARG2
0305 1023 TAD ARG1 /ADD ARG1
0306 7000 NOP
0307 3033 DCA SUM2 /SAVE RESULT
0310 7010 RAR

```

PAL10 V141

13-SEP-71

13131

1-6

0311 3034
0312 7000

DCA LINK2 /SAVE LINK
NOP

/COMPARE RESULTS OF REAL ADDS
/IF A=B, A XOR B=0, THIS IS USED TO COMPARE RESULTS

0313	7340	CLA CLL CMA	
0314	0031	AND SUM1	/GET RESULT OF ARG1+ARG2
0315	7040	CMA	/COMPLEMENT
0316	0033	AND SUM2	/AND RESULTS OF ARG2+ARG1
0317	7440	SEA	/IS SUM2 AND NOTSUM1=0
0320	5377	JMP ERROR1	/NO, ERROR
0321	7040	CMA	
0322	0033	AND SUM2	/LOAD AC WITH RESULTS OF ARG2+ARG1
0323	7040	CMA	/COMPLEMENT
0324	0031	AND SUM1	/AND WITH SUM1
0325	7440	SEA	/IS SUM1 AND NOTSUM2=0
0326	5377	JMP ERROR1	/NO, ERROR

/COMPARE REAL AND SIMULATED ADDS

0327	7340	CLA CLL CMA	
0328	0031	AND SUM1	/LOAD AC WITH RESULTS OF ARG1+ARG2
0331	7040	CMA	/COMPLEMENT
0332	0025	AND SIMAC	/AND WITH RESULTS OF SIMULATION
0333	7440	SEA	/IS SIMAC AND NOTSUM1=0
0334	5377	JMP ERROR1	/NO, ERROR
0335	7040	CMA	
0336	0025	AND SIMAC	/LOAD AC WITH SIMULATION RESULTS
0337	7040	CMA	/COMPLEMENT
0340	0031	AND SUM1	/AND WITH RESULTS OF ARG1+ARG2
0341	7440	SEA	/IS SUM1 AND NOTSIMAC=0
0342	5377	JMP ERROR1	/NO, ERROR

/COMPARE LINKS GENERATED BY REAL ADDS

0343	7340	CLA CLL CMA	
0344	0032	AND LINK1	/GET LINK FROM ARG1+ARG2
0345	7004	RAL	
0346	7240	CLA CMA	
0347	0034	AND LINK2	/GET LINK FROM ARG2+ARG1
0350	7640	SEA CLA	
0351	7020	CML	
0352	7430	SEL	/ARE THEY THE SAME
0353	5377	JMP ERROR1	/NO, ERROR

/COMPARE LINKS GENERATED BY REAL AND SIMULATED ADDS

```

0354 7340 /CLA CLL CMA /GET LINK FROM ARG1+ARG2
0355 0032 AND LINK1
0356 7004 RAL
0357 7240 CLA CMA SIMLNK /GET LINK FROM SIMULATION
0360 0026 AND SZA CLA /ARE THEY THE SAME
0361 7640 CML SZA CLA /NO, ERROR
0362 7020 SZA CLA
0363 7430 SZA CLA
0364 5377 JMP ERROR1

```

```

/SET UP FOR NEXT ADDITION
NXTADD, JMP I XLOOP2 /TEST FOR SIMULATION WITH SAME DATA
ISE ARG1 /INCREMENT ARG1
JMP SIMAD /GO TO SIMULATION
ISE ARG2 /INCREMENT ARG2
SKP I XLOOP1 /GO TO SIMULATION
JMP I XLOOP1 /TEST FOR TRANSFER TO NEXT TEST
CLA CMA ARG2 /TRANSFER ARG2 TO ARG1
AND ARG1
DCA ARG1 /CONTINUE SIMULATION
JMP SIMAD
*377
ERROR1, NOP

```

```

/ERROR HANDLER FOR ADDITION TEST
*400
ADERR, LAS /GET SWITCHES
AND SR01 /TEST SR01
SNA CLA ADPRT /SUPPRESS TYPEOUT IF SR01=1
JMS ADPRT /TYPE ERROR MESSAGE
LAS HALTA, /HALT IF SR00=0
AND SR00 /HALT WITH ADDRESS OF TEST IN AC
SNA CLA HALTA
JMS SR02 /TEST SR02
LAS SR02 /LOOP WITH SAME DATA IF SR02=1
AND SZA CLA XADD /LOOP WITH SAME DATA
JMP I XNXTAD
JMP I XNXTAD
XADD, ADD
XNXTAD, NXTADD+1

```

```

/TYPE ERROR MESSAGE FOR ADDITION TEST
ADPRT, 0
0000 CLA CLL CMA /GET FLAG FOR ERROR MESSAGE HEADER TYPEOUT
0420 7340 AND AHFLG /HAS HEADER FOR TEST BEEN TYPED
0421 0035 SNA CLA ABOUT /NO TYPE HEADER
0422 7650 JMS
0423 4267

```



```

0424 CMA 7040
0425 AND 0023
0426 DCA 3037
0427 JMS 4323
0430 CMA 7040
0431 AND 0024
0432 DCA 3037
0433 JMS 4323
0434 CMA 7040
0435 AND 0026
0436 DCA 3037
0437 JMS 4323
0440 AND 0025
0441 DCA 3037
0442 JMS 4304
0443 JMS 4323
0444 CMA 7040
0445 AND 0032
0446 DCA 3040
0447 CMA 7040
0450 AND 0031
0451 DCA 3037
0452 JMS 4304
0453 JMS 4323
0454 CMA 7040
0455 AND 0034
0456 DCA 3040
0457 CMA 7040
0460 AND 0033
0461 DCA 3037
0462 JMS 4304
0463 JMS 4323
0464 CMA 4446
0465 AND 5742
0466 JMP 5204
    ARG1
    WDO1
    WDO2
    ARG2
    WDO1
    WDO2
    SIMLNK
    WDO1
    WDO2
    SIMAC
    WDO1
    LNKOUT
    WDO1
    WDO2
    LINK1
    WDO1
    WDO2
    SUM1
    WDO1
    LNKOUT
    WDO1
    WDO2
    LINK2
    WDO1
    WDO2
    SUM2
    WDO1
    LNKOUT
    WDO1
    WDO2
    XPRINT
    WDO1
    WDO2
    HALT
    /OUTPUT ARG1
    /OUTPUT ARG2
    /OUTPUT SIMULATED LINK
    /OUTPUT SIMULATED SUM
    /OUTPUT LINK1
    /OUTPUT SUM1
    /OUTPUT LINK2
    /OUTPUT SUM2
    /TEST FOR HALT

```

```

/TYPE HEADER FOR ADDITION TEST ERROR MESSAGE
/TYPE "SIMULATED ADD TEST FAILED"
/TYPE ARG1, ARG2, SIMULATED, ARG1+ARG2, ARG2+ARG1
/SET ADD TEST HEADER FLAG
/TO PREVENT MULTIPLE HEADER TYPEOUTS
    AHOUT:
0467 JMS I 0000
0470 JMS I 4446
0471 EM1=1 5417
0472 JMS I 4446
0473 DH1=1 5177
0474 CLA CMA 7240
0475 DCA 3035
0476 JMP I 5667
    HALT:
0477 0000
0500 7240
0501 0351
0502 7402
0503 5677
    /HALT WITH ADDRESS OF TEST IN AC
    /HALT WITH ADDRESS OF ADDITION TEST IN AC

```

```

/ /
/ /TYPE LINK
/ /
LNKOUT, 0000
0504 0000
0505 7340
0506 0040
0507 7640
0510 5320
0511 7040
0512 0077
0513 4447
0514 7040
0515 0076
0516 4447
0517 5704
0520 7040
0521 0100
0522 5313

TYLNK,
0512 0077
0513 4447
0514 7040
0515 0076
0516 4447
0517 5704
0520 7040
0521 0100
0522 5313

OUT1,
0504 0000
0505 7340
0506 0040
0507 7640
0510 5320
0511 7040
0512 0077
0513 4447
0514 7040
0515 0076
0516 4447
0517 5704
0520 7040
0521 0100
0522 5313

```

```

/ /TYPE DATA WORD
/ /
WDOUT, 0000
0523 0000
0524 7340
0525 0102
0526 3011
0527 7040
0530 0411
0531 7450
0532 5345
0533 0037
0534 7640
0535 5342
0536 7040
0537 0077

NXBIT,
0523 0000
0524 7340
0525 0102
0526 3011
0527 7040
0530 0411
0531 7450
0532 5345
0533 0037
0534 7640
0535 5342
0536 7040
0537 0077

```

```

TYBIT, 4447
0540 4447
0541 5327
0542 7040
0543 0100
0544 5340
0545 7040
0546 0076
0547 4447
0550 5723
0551 0204

OUT1A,
0540 4447
0541 5327
0542 7040
0543 0100
0544 5340
0545 7040
0546 0076
0547 4447
0550 5723
0551 0204

SP1,
0540 4447
0541 5327
0542 7040
0543 0100
0544 5340
0545 7040
0546 0076
0547 4447
0550 5723
0551 0204

```

```

ADT,
0540 4447
0541 5327
0542 7040
0543 0100
0544 5340
0545 7040
0546 0076
0547 4447
0550 5723
0551 0204

/ /END OF SIMULATED ADD TEST
/ /
LOOP1, 7604
0552 7604
0553 0115
0554 7650

```

/TEST SR10
/IS SR10=1

```

PAL10      V141      13=SEP=71      13131      1=10
0555      5370      JMP      SADOK
0556      7604      ADHLT,
0557      0114      AND      SR09
0560      7640      SZA CLA
0561      7402      HLT
0562      7604      LAS
0563      0116      AND      SR11
0564      7650      SNA CLA
0565      5377      JMP      SIMR
0566      5767      JMP I
0567      0204      SIMAD
0570      4446      JMS I   XPRINT
0571      5721      OK1=1
0572      5356      JMP
0577      7000      NOP
SADOK,
*577
SIMR,

```

```

/NO, TYPE END OF TEST MESSAGE
/TEST SR09
/IS SR09=1
/YES, HALT AT END OF TEST
/TEST SR11
/IS SR11=1
/NO, GO TO NEXT TEST
/REPEAT SIMAD

```

```

/TEST ROTATION BY COMPARISON OF REAL AND SIMULATED
/ROTATES
/SET UP FOR RAL TEST

```

```

*600
SIMR01, JMS I   XR1
/TEST RAL
/GET MASK TABLE FOR
/SIMULATED RAL
/SIMULATE RAL
/SET UP TO DO REAL ROTATES

```

```

SIMRAL, CLA CLL CMA
AND      XRALTA
DCA      POINT2
JMS I   XSROT
RRAL,   CLA CLL CMA
AND      RLNK
SEA CLA
CML
CMA
AND
RAL
NOP
DCA
SZL
CMA
DCA
JMS I
JMP
JMS I
JMP

```

```

/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

```

```

*625 4753 SIMR02, JMS I   XR2
/TEST RAR

```

```

0626 7340 /
0627 0102 /
0630 3012 /
0631 4451 /
0632 7340 /
0633 0024 /
0634 7640 /
0635 7020 /
0636 7040 /
0637 0023 /
0640 7010 /
0641 7000 /
0642 3031 /
0643 7430 /
0644 7040 /
0645 3033 /
0646 4456 /
0647 5232 /
0650 4457 /
0651 5226 /

SIMRAR, CLA CLL CMA /
AND XRARTA /
DCA POINT2 /
JMS I XSROT /
CLA CLL CMA /
AND RLNK /
SZA CLA /
CML /
CMA /
AND RAC /
RAR /
NOP RRAC /
DCA /
SZL /
CMA /
DCA RRLNK /
JMS I XCOMRO /
JMP RRAR /
JMS I XNXTRO /
JMP SIMRAR /

/GET MASK TABLE FOR
/SIMULATED RAR
/SIMULATED RAR
/SET UP TP DO REAL RAR

/DO REAL RAR
/SAVE ROTATED ACCUMULATOR

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

```

```

0652 4754 /
/TEST RTL /

SIMRTL, CLA CLL CMA /
AND XRILT A /
DCA POINT2 /
JMS I XSROT /
CLA CLL CMA /
AND RLNK /
SZA CLA /
CML /
CMA /
AND RAC /
RTL /
NOP RRAC /
DCA /
SZL /
CMA /
DCA RRLNK /
JMS I XCOMRO /
JMP RRTL /
JMS I XNXTRO /
JMP SIMRTL /

/GET MASK TABLE FOR
/SIMULATED RTL
/SIMULATED RTL
/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 RTL TEST

```

```

0677 4755 /
/TEST RTR /

SIMR04, JMS I XR4 /
/TEST RTR /

```

```

0700 7340 SIMRTR, CLA CLL CMA
0701 0054 AND XRTRTA
0702 3012 DCA POINT2
0703 4451 JMS I XSROT
0704 7340 CLA CLL CMA
0705 0024 AND RLNK
0706 7640 SZA CLA
0707 7020 CHL
0710 7040 CMA
0711 0023 AND RAC
0712 7012 RTR
0713 7000 NOP
0714 3031 DCA RRAC
0715 7430 SZL
0716 7040 CMA
0717 3033 DCA RRLNK
0720 4456 JMS I XCOMRO
0721 5304 JMP RRTR
0722 4457 JMS I XNXTRO
0723 5300 JMP SIMRTR

```

```

/GET MASK TABLE FOR
/SIMULATED RTR
/SIMULATE RTR

```

```

/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

```

```

0725 7340 SIMBSW, CLA CLL CMA
0726 0055 AND XBSWTA
0727 3012 DCA POINT2
0730 4776 JMS I XSBSW
0731 7340 CLA CLL CMA
0732 0004 AND RLNK
0733 7640 SZA CLA
0734 7020 CHL
0735 7040 CMA
0736 0023 AND RAC
0737 7002 BSW RRAC
0740 7000 NOP
0741 3031 DCA RRLNK
0742 7430 SZL
0743 7040 CMA XCOMRO
0744 3033 DCA RBSW
0745 4456 JMS I XNXTRO
0746 5331 JMP SIMBSW
0747 4457 JMS I XR0TDN
0750 5325 JMP
0751 5777 JMP I

```

```

/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

```

```

0752 1400 XR1, R1
0753 1410 XR2, R2
0754 1420 XR3, R3
0755 1430 XR4, R4
0756 1440 XR5, R5
0757 0001 RALTAB, 1

```

```

0760 0002
0761 0004
0762 0010
0763 0020
0764 0040
0765 0100
0766 0200
0767 0400
0770 1000
0771 2000
0772 4000
0773 0000
0774 0001
0775 4000
0776 1236
0777 1323
    
```

XSBSH, SBSSH
XROTON, ROYDNE

/(TAPE 2)
/COMPARE RESULTS OF REAL AND SIMULATED ROTATES

```

1000
1000 0000
    
```

/COMPARE ROTATED ACCUMULATORS

```

1001 7340
1002 0025
1003 7040
1004 0031
1005 7440
1006 5226
1007 7040
1010 0031
1011 7040
1012 0025
1013 7440
1014 5226

CLA CLL CMA
AND SIMAC
CMA
AND RRAC
SEA
JMP ERROT
CMA
AND RRAC
CMA
AND SIMAC
SEA
JMP ERROT

/GET SIMULATED ROTATED ACCUMULATOR
/COMPLEMENT
/AND WITH REAL ROTATED ACCUMULATOR
/IS NOT SIMAC AND RRAC50
/NO, ERROR

/GET REAL ROTATED ACCUMULATOR
/COMPLEMENT
/AND WITH SIMULATED ROTATED ACCUMULATOR
/IS SIMAC AND NOT RRAC50
/NO, ERROR
    
```

/COMPARE ROTATED LINKS

```

1015 7340
1016 0026
1017 7640
1020 7020
1021 7040
1022 0033
1023 7440
1024 7020
1025 7430
1026 5246

CLA CLL CMA
AND SIMLINK
SEA CLA
CML
CMA
AND RRLINK
SEA
CML
SEL
JMP ERROR2

/GET SIMULATED LINK
/GET REAL ROTATED LINK
/ARE THEY THE SAME
/NO, ERROR
    
```

```

1027 2200 ISE COMROT /RETURN HERE IF NO LOOP ON ERROR
1030 5600 JMP I COMROT
/
/SET UP TO DO NEXT ROTATE
/
NXTROT, 0
1031 0000 CLA CLL CMA /GET LINK OF WORD TO BE ROTATED
1032 7340 AND RLNK /IS IT 0
1033 0024 SEA CLA /NO, CLEAR IT
1034 7640 JMP NEWLNK /YES, SET IT
1035 5244 CMA
1036 7040 DCA /INCREMENT NUMBER TO BE ROTATED
1037 3024 ISE RAC /CONTINUE SIMULATION OF PRESENT ROTATE INSTRUCTION
1040 2023 JMP I NXTROT /PRESENT SIMULATION DONE
1041 5631 ISE NXTROT /GO TO NEXT TEST
1042 2231 JMP I NXTROT
1043 5631 DCA RLNK
1044 3024 JMP I NXTROT
1045 5631

```

```

/ERROR HANDLER FOR ROTATE TEST
/
ERROR2, 0
1046 7004 LAS SR01 /TEST SR01
1047 0104 AND CLA /IS SR01=1
1050 7650 JMS ROTPRY /NO, TYPE ERROR MESSAGE
1051 4271 LAS SR00 /TEST SR00
1052 7604 AND CLA /IS SR00=1
1053 0103 JMP HALTB /NO, HALT WITH ADDRESS OF TEST IN AC
1054 7600 LAS SR02 /TEST SR02
1055 5263 AND CLA /IS SR02=1
1056 7604 SNA CLA ERROT+1 /NO, GO TO NEW DATA
1057 0105 AND CLA ERROT+2 /YES, LOOP WITH SAME DATA
1060 7650 JMP CMA
1061 5227 CLA CLL XSROT
1062 5230 AND I M4
1063 7340 TAD HLT
1064 0451 HLT
1065 1270 JMP M4
1066 7402
1067 5256
1070 7774

```

```

/ERROR TYPEOUT FOR SIMULATED ROTATE TEST ERRORS
/
ROTPRY, 0
1071 0000 CLA CLL CMA /GET ROTATE TEST HEADER FLAG
1072 7340 AND RHFLG /HAS HEADER BEEN TYPED
1073 0035 SNA CLA
1074 7650

```

```

1075 4331 JMS RHOUT
1076 7040 CMA
1077 0023 AND RAC
1100 3037 DCA WDI
1101 7040 CMA
1102 0024 AND RLNK
1103 3040 DCA WD2
1104 4460 JMS I XLNKOU
1105 4461 JMS I XWDOUT
1106 7040 CMA
1107 0025 AND SIMAC
1110 3037 DCA WDI
1111 7040 CMA
1112 0026 AND SIMLNK
1113 3040 DCA WD2
1114 4460 JMS I XLNKOU
1115 4461 JMS I XWDOUT
1116 7040 CMA
1117 0031 AND RRAC
1120 3037 DCA WDI
1121 7040 CMA
1122 0033 AND RRLNK
1123 3040 DCA WD2
1124 4460 JMS I XLNKOU
1125 4461 JMS I XWDOUT
1126 4446 JMS I XPRINT
1127 5742 CRLF=1
1130 5671 JMP I ROTPRT

```

/OUTPUT ORIGINAL LINK
/OUTPUT ORIGINAL WORD

/OUTPUT SIMULATED ROTATED LINK
/OUTPUT SIMULATED ROTATED WORD

/OUTPUT ACTUAL ROTATED LINK
/OUTPUT ACTUAL ROTATED WORD

/OUTPUT HEADER FOR ROTATE ERROR MESSAGE

/TYPE SIMULATED XXX TEST FAILED
/WHERE XXX IS THE INSTRUCTION THAT FAILED
/TYPE ORIGINAL, SIMULATED ACTUAL

```

RHOUT, 0000
RHD, 0000
JMS I XPRINT
JMS I XPRINT
DH2=1
CLA CMA
DCA RHFLG
JMP I RHOUT

```

```

RIRTAB, 2000
2000
0400
0100
0020
0004
0001
4000
1000
0200
0040
0010
0002

```



```

1155 0000
1156 2000
1157 0002

1160 0002
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

```

RICTAB, 2

```

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

```

```

1200 1200
1201 0000
1202 7300
1203 3025
1204 3026
1205 7040
1206 0412
1207 3037
1208 7040
1209 0412
1210 7450
1211 5303
1212 3040
1213 7040
1214 0023
1215 0037
1216 7440
1217 4225
1220 7040
1221 0040
1222 3037
1223 0040
1224 5207

1225 0000
1226 7240

```

*1200 SROTAL, 0

NBIT, 0

```

CLA CLL
DCA SIMAC
DCA SIMLNK
CMA AND I
DCA POINT2
CMA WD1
DCA AND I
SNA POINT2
JMP ENDROT
DCA WD2
CMA RAC
AND WD1
AND OR1
SEA JMS
CMA WD2
AND DCA
DCA NBIT
JMP
/
/
/OR BITS TO FORM PARTIALLY ROTATED WORD
/
0 CLA CMA

```

/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

PAL10 V141 13-SEP-71 13131 PAGE 1-17

1227 0040 AND WD2 /GET BIT TO BE INSERTED
 1230 7421 MQL CMA /SAVE IN M0
 1231 7040 CMA AND /GET SIMULATED ROTATED WORD
 1232 0025 AND MQA /DR BIT INTO POSITION
 1233 7501 MQA DCA SIMAC /SAVE PARTIALLY ROTATED WORD
 1234 3025 DCA JMP I OR1
 1235 5625 JMP I OR1

```

/SIMULATE BYTE SNAP
/
0
SBSW,
1236 0000 CLA CLL CMA
1237 7340 AND SBSW
1240 0236 DCA I XSR0T
1241 3451 DCA SIMAC
1242 3025 DCA SIMLNK
1243 3026 CMA
1244 7040 AND I POINT2
1245 0412 SNA ENDBSW
1246 7450 JMP WD1
1247 5277 DCA
1250 3037 CMA AND I POINT2
1251 7040 AND WD2
1252 0412 RAC
1253 3040 WD1
1254 7040 AND OR1
1255 0023 AND WD1
1256 0037 SEA OR1
1257 7440 JMS WD1
1260 4225 CMA
1261 7040 AND MQL
1262 0037 CMA
1263 7421 CMA
1264 7040 AND DCA
1265 0040 DCA MQA
1266 3037 DCA
1267 7501 MQA
1270 3040 DCA
1271 7040 CMA
1272 0023 AND AND
1273 0037 AND AND
1274 7440 SEA
1275 4225 JMS
1276 5244 JMP
1277 7340 CLA CLL
1300 0024 AND
1301 3026 DCA
1302 5636 JMP I
  
```

/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

```

1303 7340  ENDROT,  CLA CLL CMA  /GET BIT TO BE ROTATED FROM LINK
1304 0412  AND I  POINT2
1305 3040  DCA  WD2
1306 7040  CMA
1307 0116  AND K0001 /GET MASK FOR LINK
1310 0024  AND  RLNK /TEST LINK
1311 7440  SZA  OR1  /IS LINK 0
1312 4225  JMS  /PLACE LINK IN NEW POSITION
1313 7040  CMA
1314 0412  AND I  POINT2 /GET MASK FOR BIT TO BE ROTATED INTO LINK
1315 0023  AND  RAC  /TEST BIT IN WORD TO BE ROTATED INTO LINK
1316 7440  SZA
1317 7240  CLA CMA K0001 /IS IT 0
1320 0116  AND  SIMLNK /NO, SET LINK=1
1321 3026  DCA  SROTAL
1322 5600  JMP I

```

```

1323 7604  ROTONE,  LAS
1324 0115  AND  SR10 /TEST SR10
1325 7650  SNA CLA SR10=1 /IS SR10=1
1326 5342  JMP  /NO, TYPE "SIMROT"
1327 7604  LAS
1330 0114  AND  SR09 /TEST SR09
1331 7640  SZA CLA SR09=1 /IS SR09=1
1332 7402  HLT  /YES, HALT AT END OF ROTATE TESTS
1333 7604  LAS
1334 0116  AND  SR11 /TEST SR11
1335 7650  SNA CLA SR11=1 /IS SR11=1
1336 5740  JMP I  /NO, GO TO NEXT TEST
1337 5741  JMP I  /YES, REPEAT ROTATE TESTS
1340 2000  FCT
1341 0600  SIMR01
1342 4446  JMS I  XPRINT
1343 5725  OK2=1
1344 5327  JMP  ROTHLT

```

/SET UP FOR ROTATE TESTS

```

1400 0000  PAGE
1401 7340  R1,
1402 0250  CLA CLL CMA
1403 3450  AND  XM2
1404 3035  DCA I  XRHD
1405 3024  DCA  RHFLG
1406 3023  DCA  RLNK
1407 5600  DCA  RAC
1410 0000  JMP I  R1
1411 7340  CLA CLL CMA

```

/SET UP HEADER
/FOR RAL TEST ERROR MESSAGE
/CLEAR ROTATE HEADER FLAG

1412 0251 AND XM3 /SET UP HEADER
1413 3450 DCA I XRHD /FOR RAR TEST ERROR MESSAGE
1414 3035 DCA RHFLG
1415 3024 DCA RLNK
1416 3023 DCA RAC
1417 5610 JMP I R2
1420 0000
1421 7340 CLA CLL CMA
1422 0252 AND XM4
1423 3450 DCA I XRHD
1424 3035 DCA RHFLG
1425 3024 DCA RLNK
1426 3023 DCA RAC
1427 5620 JMP I R3
1430 0000
1431 7340 CLA CLL CMA
1432 0253 AND XM5
1433 3450 DCA I XRHD
1434 3035 DCA RHFLG
1435 3024 DCA RLNK
1436 3023 DCA RAC
1437 5630 JMP I R4
1440 0000
1441 7340 CLA CLL CMA
1442 0254 AND XM6
1443 3450 DCA I XRHD
1444 3035 DCA RHFLG
1445 3024 DCA RLNK
1446 3023 DCA RAC
1447 5640 JMP I R5
1450 5440 EM2=1
1451 5461 EM3=1
1452 5502 EM4=1
1453 5523 EM5=1
1454 5544 EM6=1

R3,
R4,
R5,
XM2,
XM3,
XM4,
XM5,
XM6,

/SET UP HEADER
/FOR RTR TEST ERROR MESSAGE

/SET UP HEADER
/FOR RIL TEST ERROR MESSAGE

/SET UP HEADER
/FOR BSH TEST ERROR MESSAGE

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

1600 PRINT,
1601
1602
1603
1604
1605
1606
1607
1610
1611
1612
1613
1614
1615
0 CLA CLL PRINT
TAD I POINT1
DCA POINT1
ISE PRINT
TAD I POINT1
DCA CHAR
TAD CHAR
RTR
RTR
RTR
JMS TYPSET
TAD CHAR
JMS TYPSET

```

1616 5205 JMP PRINT+5
1617 0000 TYPSET, 0
1620 0245 AND K0077
1621 7450 SNA
1622 5600 JMP I
1623 1246 TAD PRINT
1624 7510 SPA M40
1625 5230 JMP I=3
1626 1076 TAD K240
1627 5243 JMP MTP
1638 7001 IAC
1639 7440 SZA
1652 5235 JMP I=3
1633 1251 TAD K215
1634 5243 JMP MTP
1635 7001 IAC
1636 7440 SZA
1637 5242 JMP I=3
1640 1250 TAD K212
1641 5243 JMP MTP
1642 1247 TAD K336
1643 4447 JMP I
1644 5617 JMP I
1645 0077 K0077,
1646 7740 M40,
1647 0336 K336,
1650 0212 K212,
1651 0215 K215,
1652 0000 TYPE,
1653 6046 TLS
1654 6041 TSP
1655 5254 JMP I=4
1656 7200 CLA
1657 5652 JMP I TYPE

```

```

BSWTAB, 1
0001 1
0100 100
0002 2
0200 200
0004 4
0400 400
0010 10
1000 1000
0020 20
2000 2000
0040 40
4000 4000
0000 0

```

```

PAGE PAGE
FCT, FCT,
CLA CLL CLA CLL
TAD TAD TAD
DCA DCA DCA
SEQ SEQ SEQ

```

2003 3020 DCA CNTR1

/ / FALSE CARRY TEST#1

2004 7300 FCT1, CLA CLL

/ PLACE INSTRUCTIONS AND DATA IN TEST ADDRESSES

```

2005 7300 CLA CLL /DATA=0000
2006 3471 DCA I XSTA1 /LOC,=7776
2007 1136 TAD INS1 /INSTRUCTION=TAD ;,1
2010 3472 DCA I XSTA2 /LOC,=7777
2011 1332 TAD INS2 /INSTRUCTION=TAD ;,3
2012 3000 DCA TSTA3 /LOC,=0000
2013 1137 TAD INS3 /INSTRUCTION=IAC
2014 3001 DCA TSTA4 /LOC,=0001
2015 1140 TAD INS4 /INSTRUCTION=JMP I ,+2
2016 3002 DCA TSTA5 /LOC,=0002
2017 7240 CLA CMA /DATA=7777
2020 3003 DCA TSTA6 /LOC,=0003
2021 1327 TAD ADI /ADDRESS=RETI
2022 3004 DCA TSTA7 /LOC,=0004

```

/ EXECUTE INSTRUCTIONS PREVIOUSLY ASSEMBLED IN TEST

2023 7300 FCL1, CLA CLL

2024 5472 JMP I XSTA2 /PROVIDED FOR PROGRAM MODIFICATION

2025 7000 NOP

2026 7000 NOP

2027 4464 JMS I XAVREG /SAVE LINK AND AC

/ EXPECTED RESULTS ARE AC=0, LINK=1

```

2030 7430 SZL /COMPUTATION ERROR HAS OCCURED
2031 7440 SEA /TEST FOR HALT
2032 4465 JMS I XDATER /TEST FOR LOOP
2033 7410 SKP /ADDRESS OF NEXT TEST
2034 4466 JMS I XHALT2 /GO TO NEXT TEST
2035 4467 JMS I XLOOP
2036 5223 JMP FCL1
2037 7200 CLA
2040 1123 TAD SEQ2
2041 3154 DCA SEQ
2042 5554 JMP I SEQ

```

/ FALSE CARRY TEST#2

2043 7300 FCT2, CLA CLL

/PLACE INSTRUCTIONS AND DATA IN TEST ADDRESSES

2044	7340	CLA CLL	CMA	/DATA=7777
2045	3471	DCA I	XSTA1	/LOC,=7776
2046	1136	TAD	INS1	/INSTRUCTION=TAD I,=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	CLA CLL		
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	SEL		
2064	7440	SEA		
2065	4465	JMS I	XDATER	
2066	7410	SKP		
2067	4466	JMS I	XHALT2	
2070	4467	JMS I	XLOOP	
2071	5256	JMP	FCL2	
2072	7300	CLA		
2073	1124	TAD	SEQ3	
2074	3154	DCA	SEQ	
2075	5554	JMP I	SEQ	

/FALSE CARRY TEST #3

/CLA CLL

2076 7300

2077	1137	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=RETS
2106	3001	DCA	TSTA4	/LOC,=0001

2107 7300
 2110 5471
 2111 7000
 2112 7000
 2113 4464

FCL3,
 RET3,

CLA CLL
 JMP I XSTA1
 NOP
 NOP
 JMS I XAVREG
 JMS I XAVREG

2114 7430
 2115 7440
 2116 4465
 2117 7410
 2120 4466
 2121 4467
 2122 5307
 2123 7200
 2124 1125
 2125 3154
 2126 5554
 2127 2025
 2130 2060
 2131 2111
 2132 1003
 2133 1421

AD1,
 AD2,
 AD3,
 INS2,
 INS16,

SZL
 SZA
 JMS I XDATER
 SKP
 JMS I XHALT2
 JMS I XLOOP
 JMP FCL3
 CLA
 TAD
 DCA
 JMP I
 RET1
 RET2
 RET3
 1003
 1421

/*TAD ,+3 IN 0000

2200

PAGE

2200 7300

FCL4,

CLA CLL

/*FALSE CARRY TEST #4

2201 7340
 2202 3471
 2203 1136
 2204 3472
 2205 1142
 2206 3000
 2207 1141
 2210 3001
 2211 1324
 2212 3002

FCS4,

CLA CLL CMA
 DCA I XSTA1
 TAD INS1
 DCA I XSTA2
 TAD INS6
 DCA TSTA3
 TAD INS5
 DCA TSTA4
 TAD AD4
 DCA TSTA5

/*DATA=7777
 /*LOC,=7776
 /*INSTRUCTION=TAD ,+1
 /*LOC,=7777
 /*INSTRUCTION=CMA CML RAR
 /*LOC,=0000
 /*INSTRUCTION=JMP I ,+1
 /*LOC,=0001
 /*ADDRESS=RET4
 /*LOC,=0002

2213 7340

FCL4,

CLA CLL CMA


```

2214 5472 JMP I XSTA2
2215 7000 NOP
2216 7000 NOP
2217 4464 JMS I XAVREG
/
/
/
2220 7430 SEL
2221 7440 SZA
2222 4465 JMS I XDATER
2223 7410 SKP I
2224 4466 JMS I XHALT2
2225 4467 JMS I XLOOP
2226 5213 JMP FCL4
2227 1126 TAD SEQ5
2228 3134 DCA SEQ
2231 5534 JMP I SEQ

```

/FALSE CARRY TEST #5

CLA CLL

FCT5,

7300

2232

```

2233 7300 CLA CLL
2234 1143 TAD IN97
2235 3472 DCA I XSTA2
2236 1137 TAD IN93
2237 3000 DCA TSTA3
2240 1137 TAD IN93
2241 3001 DCA TSTA4
2242 1151 TAD IN93
2243 3002 DCA TSTA5
2244 1325 TAD AD5
2245 3003 DCA TSTA6

```

```

/INSTRUCTION=ISE ;=1
/LOC.=7777
/INSTRUCTION=IAC
/LOC.=0000
/INSTRUCTION=IAC
/LOC.=0001
/INSTRUCTION=JMP I ;+1
/LOC.=0002
/ADDRESS=RET5
/LOC.=0003

```

FCL5,

7340

2246

```

2247 3471 DCA I XSTA1
2250 7040 CMA
2251 5472 JMP I XSTA2
2252 7000 NOP
2253 7000 NOP
2254 4464 JMS I XAVREG

```

RET5,

7430

2255

```

2256 7440 SEL
2257 4465 JMS I XDATER

```

```

2260 7410 SKP
2261 4466 JMS I XHALT2
2262 4467 JMS I XLOOP
2263 5246 JMP FCL5
2264 7200 CLA
2265 1127 TAD SEQ6
2266 3154 DCA SEQ
2267 5554 JMP I SEQ

```

```

/ /
/ /FALSE CARRY TEST #6
/ /
/ CLA CLL
/ CLA CLL
TAD INSB
DCA I XSTA2
TAD INSB
DCA TSTA4
TAD INSB
DCA TSTA5
TAD AD6
DCA TSTA6
/
/
/ INSTRUCTION=ISE ,+1
/ LOC,=7777
/ INSTRUCTION=IAC
/ LOC,=0001
/ INSTRUCTION=JMP I ,+1
/ LOC,=0002
/ ADDRESS=RET6
/ LOC,=0003

```

```

2270 7300 FCT6,
2271 7300 CLA CLL
2272 1144 TAD
2273 5472 DCA I
2274 1137 TAD
2275 3001 DCA
2276 1151 TAD
2277 3002 DCA
2300 1326 TAD
2301 3003 DCA

```

```

2302 7340 FCL6,
2303 3000 CLA CLL CMA
2304 7240 DCA TSTA5
2305 5472 CLA CMA XSTA2
2306 7000 JMP I
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
2322 3154 DCA SEQ7
2323 5554 JMP I SEQ
2324 2215 RET4
2325 2252 AD5,
2326 2306 AD6,

```

```

/ /
/ /FALSE CARRY TEST#7
/ /
/ CLA CLL
/ /
/ /
/ CLA CLL
/ TAD INS9
/ DCA I XSTA2
/ TAD INS3
/ DCA TSTA4
/ TAD INS13
/ DCA TSTA5
/ TAD AD7
/ DCA TSTA6
/ /
/ /
/ CLA CLL CMA
/ DCA TSTIND
/ CMA TSTAS
/ DCA
/ CMA
/ JMP I XSTA2
/ NOP
/ NOP
/ JMS I XAVREG
/ /
/ /
/ SZL
/ SZA
/ JMS I XDATE
/ SKP I XHALT2
/ JMS I XLOOP
/ JMP FCL7
/ CLA
/ TAD SEQ8
/ DCA SEQ
/ JMP I SEQ
/ /
/ /FALSE CARRY TEST #8
/ /
/ CLA CLL
/ /
/ /
/ CLA CLL
/ TAD INS3

```

```

/INSTRUCTION=ISE I TSTIND
/LOC,=7777
/INSTRUCTION=IAC
/LOC,=0001
/INSTRUCTION=JMP I ,+1
/LOC,=0002
/ADDRESS=RET7
/LOC,=0003

```

```

2400 7300 FCT7,
2401 7300
2402 1145
2403 3472
2404 1137
2405 3001
2406 1151
2407 3002
2410 1326
2411 3003

```

```

2412 7340
2413 3010
2414 7040
2415 3000
2416 7040
2417 5472
2420 7000
2421 7000
2422 4464

```

```

2423 7430
2424 7440
2425 4465
2426 7410
2427 4466
2430 4467
2431 5212
2432 7200
2433 1131
2434 3134
2435 5554

```

```

2436 7300 FCT8,
2437 7300
2440 1137

```

/INSTRUCTION=IAC


```

2505 3472 DCA I XSTA2
2506 7240 CLA CMA
2507 5472 JMP I XSTA2
2510 7000 NOP
2511 7000 NOP
2512 4464 JMS I XAVREG
/
/
/
2513 7430 SZL
2514 7440 SZA
2515 4465 JMS I XDATE
2516 7410 SKP I
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

```

```

AD7,
AD8,
AD9,

```

2600 PAGE

/FALSE CARRY TEST #10

FCT10, CLA CLL

FCS10, CLA CLL

```

TAD INS12
DCA I XSTA2
TAD INSS
DCA TSTA4

```

```

TAD INS13
DCA TSTA5
TAD AD10
DCA TSTA6

```

```

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

```

```

FCL10, CLA CLL CMA
DCA TSTIND
CMA
JMP I XSTA2
RET10, NOP
NOP

```

```

2612 7340
2613 3010
2614 7040
2615 5472
2616 7000
2617 7000

```

```

2620 4464 JMS I XAVREG
/
/
/
2621 7430 SZL
2622 7440 SZA
2623 4465 JMS I XOATER
2624 7410 SKP I
2625 4466 JMS I XHALT2
2626 4467 JMS I XLOOP
2627 5212 JMP FCL10
2630 7200 CLA
2631 1134 TAD SEQ11
2632 3134 DCA SEQ
2633 5554 JMP I SEQ

```

/ / FALSE CARRY TEST #11

```

2634 7300 FCL11, CLA CLL
/
/
/
2635 7300 FCS11, CLA CLL
2636 1137 TAD INS3
2637 3000 DCA TSTA3
2640 1141 TAD INS5
2641 3001 DCA TSTA4
2642 1316 TAD AD11
2643 3002 DCA TSTA5
/
/
/
/ INSTRUCTION=IAC
/ LOC.=0000
/ INSTRUCTION=JMP I,+1
/ ADDRESS=0001
/ ADDRESS=RETI1
/ LOC.=0002

```

```

2644 7300 FCL11, CLA CLL
2645 1133 TAD INS15
2646 3472 DCA I XSTA2
2647 7240 CLA CMA
2650 5472 JMP I XSTA2
2651 7000 NOP
2652 7000 NOP
2653 4464 JMS I XAVREG
/
/
/
/ INSTRUCTION=JMS
/ LOC.=7777

```

```

2654 7430 SZL
2655 7440 SZA
2656 4465 JMS I XOATER
2657 7410 SKP I
2660 4466 JMS I XHALT2
2661 4467 JMS I XLOOP
2662 5244 JMP FCL11
2663 7200 CLA
2664 1135 TAD SEQ12

```

13-SEP-71 13131 1-30

V141 13-SEP-71 13131

PAL10 V141 13-SEP-71 13131

2665 3154
2666 5554

/ /
/ FALSE CARRY TEST #12
/

2667 7300 FCT12:

CLA CLL
TAD I
DCA I
DCA
TAD
DCA

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

INS3
XSTA2
INS14
TSTA3
AD12
TSTA4
/ INSTRUCTION=IAC
/ LOC.=7777
/ INSTRUCTION=JMP I ,+1
/ LOC.=0000
/ ADDRESS=RET12
/ LOC.=0001

CLA CLL CMA
JMP I XSTA2
NOP
NOP
JMS I XAVREG

2677 7340
2700 5472
2701 7000
2702 7000
2703 4464

FCL12:
RET12:

SEL
SEA
JMS I XDATA
SKP I
JMS I XHALT2
JMS I XLOOP
JMP I FCL12
JMP I ,+1
ENDFCT
RET10
RET11
RET12

2704 7430
2705 7440
2706 4465
2707 7410
2710 4466
2711 4467
2712 5277
2713 5714
2714 3200
2715 2616
2716 2051
2717 2701

AD10:
AD11:
AD12:

3000 PAGE

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

SAMEAS, 0
CLA CLL CMA
AND W2
CMA

3000 0000
3001 7340
3002 0040
3003 7040

```

3004 0037 AND W1
3005 7640 SEA CLA
3006 5600 JMP I SAMEAS
3007 7040 CMA
3010 0037 AND W1
3011 7040 CMA
3012 0040 AND W2
3013 7640 SEA CLA
3014 5600 JMP I SAMEAS
3015 2200 ISE
3016 5600 JMP I SAMEAS

```

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

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

/W1=W2

/SAVE AC AND LINK

```

3017 0000 SAVREG, 0
3020 3025 DCA
3021 7430 SZL
3022 7040 CMA
3023 3026 DCA
3024 7040 CMA
3025 0025 AND
3026 5617 JMP I SAVREG

```

TEMPAC
TEMPL
TEMPAC
SAVREG

/HALT ON ERROR; DISPLAY ADDRESS OF FAILED TEST IN AC

```

3027 0000 HALT2, 0
3030 7604 LAS
3031 0103 AND SR00
3032 7640 SEA CLA
3033 5627 JMP I HALT2
3034 1154 TAD
3035 7402 HLT
3036 5627 JMP I HALT2

```

/TEST SR00
/SUPPRESS HALT IF SR00=1
/PUT ADDRESS OF FAILED TEST IN
/AC AND STOP
/CONTINUE TESTING

/DATA ERROR HAS OCCURED

```

3037 0000 DATER, 0
3040 7604 LAS
3041 0104 AND SR01
3042 7450 SNA
3043 4256 JMS
3044 2237 ISE
3045 5637 JMP I DATER

```

/TEST SR01
/SUPPRESS ERROR TYPE IF SR01=1
/SET UP FOR ERROR TYPE

/LOOP ON DATA ERROR

```

3046 0000 LOOP, 0
3047 7604 LAS

```


PAL10

V141

13=SEP=71

13131

1=32

```

3050 0105 AND SR02
3051 7650 SNA CLA
3052 5254 JMP NLOOP
3053 5646 JMP I LOOP
3054 2246 ISZ LOOP
3055 5646 JMP I LOOP

```

NLOOP,

/TYPE DATA ERROR MESSAGE

```

3056 0000
3057 4446 JMS I XPRINT
3060 5744 DATE=1
3061 1037 TAD W1
3062 4673 JMS I XADOUT
3063 7340 CLA CLL CMA
3064 0025 AND TEMPAC
3065 3037 DCA WD1
3066 0026 AND TEMPL
3067 3040 DCA WD2
3070 4460 JMS I XLNKOU
3071 4461 JMS I XADOUT
3072 5656 JMP I TYP2
3073 3227 XADOUT,

```

TYP2,

/TYPE "DATA ERROR"

/TYPE TEST ADDRESS

/OUTPUT RECEIVED LINK
/OUTPUT RECEIVED AC

/END OF PASS

```

3200 7300 PAGE
3201 2020 ENDFCT,
3202 5224 CLA CLL CNTR1
3203 7604 JMP OUT
3204 0115 AND SR10
3205 7650 SNA CLA FCTOK
3206 5221 JMP SR09
3207 7604 LAS
3210 0114 AND SR09
3211 7640 SZA CLA
3212 7402 HLT
3213 7604 LAS
3214 0116 AND SR11
3215 7640 SZA CLA
3216 5224 JMP OUT
3217 5620 JMP I,+1
3220 3400 RNAD1
3221 4446 JMS I XPRINT
3222 5732 OK3=1
3223 5207 JMP FCTHLT
3224 1122 TAD SEQ1
3225 3154 DCA SEQ
3226 5554 JMP I SEQ

```

PAGE

ENDFCT,

CNTR1
OUT

SR10
FCTOK

SR09

SR11

OUT
+1

XPRINT

FCTHLT
SEQ1
SEQ

/INCREMENT PASS COUNT
/PASS NOT COMPLETE

/TEST SR10
/IS SR10=1
/NO, TYPE FCT

/TEST SR09
/IS SR09=1
/YES, HALT

/TEST SR11
/IS SR11=1
/YES, LOOP ON FCT
/NO, GO TO NEXT TEST

/ CONVERT ADDRESS TO ASCII AND OUTPUT

3227	0000	
3230	3037	ADOUT,
3231	1037	DCA
3232	0172	TAD
3233	3264	AND
3234	1037	DCA
3235	7006	RTL
3236	7004	RAL
3237	0266	AND
3240	1264	TAD
3241	1267	TAD
3242	3264	DCA
3243	1037	TAD
3244	7012	RTR
3245	7012	RTR
3246	7012	RTR
3247	0172	AND
3250	3263	DCA
3251	1037	TAD
3252	7012	RTR
3253	7010	RAR
3254	0266	AND
3255	1263	TAD
3256	1267	TAD
3257	3263	DCA
3260	4446	JMS I
3261	3262	A1=1
3262	5627	JMP I
3263	0000	0
3264	0000	0
3265	4000	A1
3266	0700	A2
3267	6060	K0700,
		K6060,

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

3400			
3401	7300	CLA CLL	XRAND
3402	4473	JMS I	
3403	7300	CLA CLL	
3404	1041	TAD	RANDA
3405	1043	TAD	RANDC
3406	1043	TAD	RANDC
3407	1041	TAD	RANDA
3410	1041	TAD	RANDA
3411	1043	TAD	RANDC
3412	1043	TAD	RANDC
3413	1041	TAD	RANDA
3414	1041	TAD	RANDA
3415	1043	TAD	RANDC
3416	1041	TAD	RANDA
3417	1043	TAD	RANDC
3420	1043	TAD	RANDC

PAGE
RNA01,

/ GENERATE RANDOM NUMBERS

/AC=0

/AC=0

/AC=0

/AC=0

```

3421 1041 TAD RANDA
3422 1041 TAD RANDA
3423 1043 TAD RANDC
3424 1043 TAD RANDC
3425 1043 TAD RANDC
3426 1041 TAD RANDA
3427 1043 TAD RANDC
3430 1041 TAD RANDA
3431 1041 TAD RANDA
3432 1041 TAD RANDA
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
    
```

/AC=0

/AC=0

/AC=0

/SAVE AC AND LINK
/IS LINK=1, AND AC=0

/ERROR, AC NOT 0, OR LINK NOT 1 OR BOTH
/RESULTS OK

/RANDOM ADD TEST & ERROR HANDLER

```

3447 0000 RN1ER,
3450 7604 LAS
3451 0104 AND SR01
3452 7640 SZA CLA
3453 5302 JMP SKHLT
3454 4446 JMS I XPRINT
3455 5565 EM10-1
3456 4446 JMS I XPRINT
3457 5316 DH4-1
3460 7340 CLA CLL CMA
3461 0041 AND RANDA
3462 3037 DCA WD1
3463 4461 JMS I XWDOUT
3464 7340 CLA CLL CMA
3465 0043 AND RANDC
3466 3037 DCA WD1
3467 4461 JMS I XWDOUT
3470 7340 CLA CLL CMA
3471 0025 AND TEMPAC
3472 3037 DCA WD1
3473 7040 CMA
3474 0026 AND
3475 5040 DCA WD2
3476 4460 JMS I XLNKOU
3477 4461 JMS I XWDOUT
3500 4446 JMS I XPRINT
3501 5742 CRLF-1

    /TEST SR01
    /IS SR01=1
    /YES, SUPPRESS ERROR TYPEOUT
    /TYPE "RANDOM ADD TEST1 FAILED"
    /TYPE "RANDA, RANDC, RESULT"

    /OUTPUT RANDA

    /OUTPUT RANDC

    /OUTPUT RESULTANT LINK
    /OUTPUT RESULTANT AC
    
```

PAL10 V141 13-SEP-71 13131 PAGE 1-35
 SKHLT, LAS SR00 /TEST SR00
 AND SZA CLA RNIER /IS SR00=1
 JMP I RNIER /YES, SUPPRESS ERROR HALT
 CLA CLL RNIER
 TAD HLT
 JMP I RNIER /HALT WITH ADDRESS OF RNAD1 IN AC

/RANDOM NUMBER GENERATOR
 RANDOM, 0
 CLA CLL RANDA
 TAD RAL
 SEL K0003
 TAD RANDA
 DCA RANDA
 CIA RANDC
 DCA R2A
 CLL RAL
 TAD SEL
 RAL K0003
 SEL R2A
 TAD RAL
 DCA RANDC
 CIA R2A
 DCA LINKR
 CLL LINKR
 TAD LINKRC
 RAL RANDOM
 SEL R2A
 TAD K0003
 DCA R2A
 CIA LINKR
 DCA LINKR
 CLL LINKRC
 TAD RANDOM
 RAL R2A
 SEL K0003
 TAD R2A
 DCA LINKR
 CIA LINKR
 DCA LINKRC
 CLL RANDOM
 TAD JMP I
 RAL R2A
 SEL 1
 TAD 3
 DCA K0003,

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

PAGE RNAD2, CLA CLL CMA /GET RANDOM NUMBER
 AND RANDA /STORE IT
 DCA APOS
 CMA RANDA
 AND RANDC
 CMA ANEG
 DCA

```

3607 7040 CMA
3610 0103 AND
3611 3352 DCA
3612 7040 CMA
3613 0352 AND
3614 7040 CMA
3615 3353 DCA
3616 7040 CMA
3617 0346 AND
3620 0352 AND
3621 7440 SZA
3622 5232 JMP
3623 7040 CMA
3624 0346 AND
3625 4301 JMS
3626 7040 CMA
3627 0347 AND
3630 3351 DCA
3631 5240 JMP
3632 7240 CMA
3633 0347 AND
3634 4315 JMS
3635 7040 CMA
3636 0346 AND
3637 3351 DCA
3640 7340 CLA
3641 0350 AND
3642 1351 TAD
3643 7430 SZL
3644 7001 IAC
3645 4464 JMS
3646 4463 JMS
3647 7410 SKP
3650 4756 JMS
3651 4467 JMS
3652 5240 JMP
3653 5254 JMP

3654 7340 CBTST2, CLL CLA CMA
3655 0351 AND BNEG
3656 1350 TAD BPOS
3657 7430 SZL
3658 7001 IAC
3659 4464 JMS
3661 4463 JMS
3662 4463 SKP
3663 7410 SKP
3664 4756 JMS
3665 4467 JMS
3666 5254 JMP

3667 7340 MOVMSK, CLA CLL CMA
3670 0352 AND MASK
3671 7010 RAR
3672 3352 DCA MASK

3607 7040 K4000 /GET MASK
3610 0103 MASK
3611 3352 MASK
3612 7040 MASK
3613 0352 NMASK
3614 7040 APOS
3615 3353 MASK
3616 7040 MODNEG
3617 0346 APOS
3620 0352 XOR1
3621 7440 ANEG
3622 5232 BNEG
3623 7040 CBTST1
3624 0346 ANEG
3625 4301 XOR2
3626 7040 APOS
3627 0347 BNEG
3630 3351 CBTST1
3631 5240 ANEG
3632 7240 XOR2
3633 0347 APOS
3634 4315 BNEG
3635 7040 BPOS
3636 0346 BNEG
3637 3351 XAVREG
3640 7340 CMA XAMEA
3641 0350 XRN2ER
3642 1351 NERRORP
3643 7430 CBTST1
3644 7001 CBTST2
3645 4464 /AC AND MASK DIFFERENT, ERROR
3646 4463 /NO ERROR, AC AND MASK THE SAME
3647 7410 /RETURN HERE FOR LOOPING
3650 4756 /LOAD AC WITH UNMODIFIED ARGUMENT
3651 4467 /ADD MODIFIED ARGUMENT
3652 5240 /DID CARRY PROPAGATE INTO LINK
3653 5254 /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
/NO ERROR, AC AND MASK THE SAME
/RETURN HERE FOR LOOPING

```

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

```

3673 7420 SNL
3674 5212 JMP I
3675 4467 JMS I
3676 5200 JMP RNAD2
3677 5700 JMP I,*1
3700 4200 RARR
3701 0000 XQR1,
3702 0353 AND NMASK
3703 7040 CMA ABNOT
3704 3354 DCA ABNOT
3705 7040 CMA ANEG
3706 0347 AND MASK
3707 0352 AND ABNOT
3710 7040 CMA AND
3711 0354 AND ABNOT
3712 7040 CMA BPOS
3713 3350 DCA XOR1
3714 5701 JMP I
3715 0000 XQR2,
3716 0352 AND MASK
3717 7040 CMA ABNOT
3720 3354 DCA APOS
3721 7040 CMA NMASK
3722 0346 AND ABNOT
3723 0353 AND BPOS
3724 7040 CMA XOR2
3725 0354 AND
3726 3350 DCA
3727 5715 JMP I

```

```

3730 0000 SAMEA,
3731 7040 CMA
3732 3355 DCA NOTAC
3733 7040 CMA TEMPAC
3734 0025 AND NMASK
3735 0353 AND EROUT1
3736 7440 SZA
3737 5344 JMP
3740 7040 CMA MASK
3741 0352 AND NOTAC
3742 0355 AND
3743 7440 SZA EROUT1, ISE
3744 2330 JMP I
3745 5730 APOS,
3746 0000 ANEG,
3747 0000 BPOS,
3750 0000 BNEG,
3751 0000 MASK,
3752 0000 NMASK,
3753 0000 ABNOT,
3754 0000 NOTAC,
3755 0000 XRN2ER, RN2ER
3756 4000

```

```

4000 PAGE
/ERROR HANDLER FOR RANDOM ADD TEST 2,
/ RNZER,
0000 LAS
4001 AND SR01
4002 SZA CLA
4003 JMP I XHLT
4004 JMS I XPRINT
4005 EM11=1
4006 JMS I XPRINT
4007 DH6=1
4008 CLA CLL
4009 AND BPOS
4010 DCA I WD1
4011 JMS I XHDOUT
4012 CMA
4013 AND BNEG
4014 DCA I WD1
4015 JMS I XHDOUT
4016 CMA
4017 AND MASK
4018 DCA I WD1
4019 JMS I XHDOUT
4020 CMA
4021 AND TEMPAC
4022 DCA I WD1
4023 JMS I XHDOUT
4024 CMA
4025 AND DCA
4026 DCA I XPRINT
4027 JMS I XPRINT
4028 CRLF=1
4029 LAS
4030 AND SR00
4031 SZA CLA
4032 JMP I RNZER
4033 CLA CLL
4034 TAD
4035 HLT
4036 JMP I RNZER
4037 RNZER
4038 RNZER
4039 RNZER
4040 RNZER
4041 RNZER
4042 RNZER
/TEST SR01
/IS SR01 = 1
/YES SUPPRESS ERROR TYPEOUT
/NO, TYPE "RANDOM ADD TEST 2 FAILED"
/TYPE ARG1, ARG2, ARG1, ARG2, EXPECTED
/OUTPUT ARG1
/OUTPUT ARG2
/OUTPUT EXPECTED RESULT
/OUTPUT RESULTANT IC
/TEST SR00
/IS SR00 = 1
/YES, DO NOT HALT
/NO, HALT WITH ADDRESS IN AC

```

/ROTATE RANDOM NUMBER RIGHT USING RAR

```

4175 3752
4176 3751
4177 3750
4200 4200
4300 7300

```

```

PAGE
RARR, CLA CLL

```

4201 1044 TAD LINKR /GET LINK TO BE ROTATED
 4202 7440 SEA /GET LINK TO BE ROTATED
 4203 7220 CLA CML RANDA /GET NUMBER TO BE ROTATED
 4204 1041 TAD RAR
 4205 7010 RAR
 4206 7010 RAR
 4207 7010 RAR
 4210 7010 RAR
 4211 7010 RAR
 4212 7010 RAR
 4213 7010 RAR
 4214 7010 RAR
 4215 7010 RAR
 4216 7010 RAR
 4217 7010 RAR
 4220 7010 RAR
 4221 7010 RAR
 4222 7010 RAR
 4223 7010 RAR
 4224 7010 RAR
 4225 7010 RAR
 4226 7010 RAR
 4227 7010 RAR
 4230 7010 RAR
 4231 7010 RAR
 4232 7010 RAR
 4233 7010 RAR
 4234 7010 RAR
 4235 7010 RAR
 4236 7010 RAR
 4237 7000 NOP
 4240 7000 NOP
 4241 4464 JMS I XAVREG
 4242 1043 TAD RANDC
 4243 7640 SEA CLA
 4244 5250 JMP I+4
 4245 1044 TAD LINKR
 4246 3037 DCA HD1
 4247 1026 TAD TEMPL
 4250 3040 DCA WD2
 4251 4462 JMS I XAMEAS
 4252 4735 JMS I XRARR
 4253 4467 JMS I NERROP
 4254 5200 JMP RARR

/SAVE AC AND LINK
 /ADD COMPLEMENT OF NUMBER TO AC
 /ARE THEY EQUAL
 /NO, ERROR

/ARE LINKS THE SAME
 /NO, ERROR
 /TEST FOR LOOPING
 /LOOP ON RARR

RALR,
 4255 7300 /ROTATE RANDOM NUMBER LEFT USING RAL
 4256 1044 CLA CCL
 4257 7440 TAD LINKR /GET LINK TO BE ROTATED
 4260 7220 SEA CML RANDA /GET NUMBER TO BE ROTATED
 4261 1041 TAD RAR
 4262 7004 RAR
 4263 7004 RAR
 4264 7004 RAR


```

4265 7004 RAL
4266 7004 RAL
4267 7004 RAL
4270 7004 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
4311 7004 RAL
4312 7004 RAL
4313 7004 RAL
4314 7000 NOP
4315 7000 NOP
4316 4464 JMS I
4317 1043 TAD
4320 7440 SZA
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 5013 XRALR;
4335 5000 XRARR;

```

```

/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

CLA CLL LINKR /GET LINK TO BE ROTATED
SZA
CLA CML RANDA /GET NUMBER TO BE ROTATED
TAD
RTL
RTL
RTL

```

```

PAGE
RTL,
4400
4401 1044
4402 7440
4403 7220
4404 1041
4405 7006
4406 7006
4407 7006
4410 7006

```



```

4474 RTR
4475 RTR
4476 RTR
4477 RTR
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
44
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 2020
4533 5366
4534 7604
4535 0115
4536 7650
4537 5363
4540 7604
4541 0114
4542 7640
4543 7402
4544 7604
4545 0116
4546 7640
4547 5366
4550 7604
4551 0173
4552 7110
4553 7012
4554 3175
4555 7604
4556 0107
4557 7640

CNTR1
ENRN
SR10
SNA CLA
RNDOK
SR09
AND
SEA CLA
HLT
LAS
AND
SEA CLA
ENRN
SR11
ENRN
K0070
RAR CLL
RTR
DCA FLOSAV
LAS
AND SR04
SEA CLA

/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

RNDHLT,
FLOSW,

```

```

/SAVE THE SWITCHES
/MASK FIELD RELOCATION SWITCH

```

```

PAL10      V141      13-SEP-71      13131      PAGE 1-43      /GOT FIELD RELOCATION SWITCH AND GO
4560      JMP I XFLDCK      /NO, GO TO SIMULATED ADDITION TEST
4561      JMP I      ,+1
4562      RSIMAD
4563      JMS I XPRINT
4564      OK4-1
4565      JMP I RNDHLT
4566      JMP I      ,+1
4567      RNAD1
4570      RTRER
4571      RYLER
4572      XFLDCK, FLDCHK
PAGE
/ROUTINE TO SORT AND COMPARE RELOCATION INFORMATION
FLDCHK, JMS FLDEND      /YES, FIND NUMBER OF FIELDS PRESENT
JMS RELOC      /RELOCATE TO NEXT BANK PRESENT OR BANK 0
CLA CLL CHA RTL      /AG TO 7775
JMS LFCR      /PRINT SOME CR-LF
JMS ASTRK      /PRINT SOME *****
JMS FLDNO      /PRINT AMOUNT OF MEMORY
JMS I XPRINT      /PRINT " EXTENDED BANKS OF MEMORY TO BANK "
BKMS      /TEXT FOR EXTENDED BANKS OF MEMORY TO BANK
JMS FLDHR      /PRINT NEW FIELD
JMS ASTRK      /PRINT SOME *****
CLA CLL CHA RAL      /AG TO 7776
JMS LFCR      /PRINT SOME CR = LP
TAD FLD5AV
CIA
TAD FLDNUM
SNA CLA
JMP ,+3
HLT CLA
JMP I XFLD5H
TAD FLDG0
TAD K0002
DCA ,+1
0000
FLDEX, JMP I ,+1
RSIMAD
/ROUTINE TO DETERMINE NUMBER OF BANKS OF MEM.
FLDFND, 0
CLA CLL
DCA FLDNUM
TAD KSTOP
DCA FLD CNT
CDF
DCA I K0
TAD KCDF
TAD K0010
DCA FLD0F
FLD0F, 0
4631      0000
4632      7300
4633      3174
4634      1371
4635      3176
4636      6201
4637      3571
4640      1372
4641      1113
4642      3243
4643      0000

```

```

PAL10      V141      13-SEP-71      13131      1=44
4644      7340      CLA CLL CMA
4645      3571      DCA I K0
4646      1571      TAD I K0
4647      7650      SNA CLA
4650      5255      JMP ,+5
4651      2174      ISZ FLDNUM
4652      1243      TAD FLD0F
4653      2176      ISZ FLD0CNT
4654      5241      JMP FLD0F =2
4655      7300      CLA CLL
4656      4201      CDF 0
4657      1571      TAD I K0
4660      7650      SNA CLA
4661      5631      JMP I FLD0FND
4662      7602      HLT CLA
4663      5274      JMP FLD0FND ,+1

/ROUTINE TO MOVE PROGRAM TO NEXT FIELD OR FIELD 0
/RELOC,
4664      0000
4665      7300      CLA CLL
4666      3176      DCA FLD0CNT
4667      6224      RIF
4670      1113      TAD K0010
4671      0375      AND K0070
4672      3312      DCA FLD0FRM
4673      7301      CLA CLL IAC
4674      1174      YAD FLDNUM
4675      7004      RAL
4676      7006      RTL
4677      7041      CIA
4678      1312      TAD FLD0FRM
4679      7600      SNL CLA
4682      1312      TAD FLD0FRM
4683      1372      TAD K00F
4684      3314      DCA FLD0G
4685      6224      RIF
4686      1372      TAD K00F
4687      3312      DCA FLD0FRM
4688      1312      TAD FLD0FRM
4689      3317      DCA FLD0RM1
4690      0000      FLD0FRM, TAD I FLD0CNT
4691      1576      0000      FLD0G, 0000
4692      3576      DCA I FLD0CNT
4693      1576      TAD I FLD0CNT
4694      0000      FLD0RM1, 0000
4695      7041      CIA
4696      1576      TAD I FLD0CNT
4697      7650      SNA CLA
4698      5326      JMP ,+3
4699      7602      HLT CLA
4700      5312      JMP FLD0FRM
4701      2176      ISZ FLD0CNT
4702      5312      JMP FLD0FRM
4703      5312      JMP FLD0FRM
4704      5312      JMP FLD0FRM
4705      5312      JMP FLD0FRM
4706      5312      JMP FLD0FRM
4707      5312      JMP FLD0FRM
4708      5312      JMP FLD0FRM
4709      5312      JMP FLD0FRM
4710      5312      JMP FLD0FRM
4711      5312      JMP FLD0FRM
4712      5312      JMP FLD0FRM
4713      5312      JMP FLD0FRM
4714      5312      JMP FLD0FRM
4715      5312      JMP FLD0FRM
4716      5312      JMP FLD0FRM
4717      5312      JMP FLD0FRM
4718      5312      JMP FLD0FRM
4719      5312      JMP FLD0FRM
4720      5312      JMP FLD0FRM
4721      5312      JMP FLD0FRM
4722      5312      JMP FLD0FRM
4723      5312      JMP FLD0FRM
4724      5312      JMP FLD0FRM
4725      5312      JMP FLD0FRM
4726      5312      JMP FLD0FRM
4727      5312      JMP FLD0FRM

/TRY EXTENDED FIELD
/SAME IF FIELD PRESENT
/DATE BAD OR FIELD NOT THERE
/UPDATE FIELD COUNT
/GET LAST FIELD CDF
/STOP AFTER 7
/TRY NEXT FIELD
/BACK TO FIELD 0
/DID FIELD 0 CHANGE
/FIELD 0 O.K. EXIT
/FIELD ERROR
/TRY AGAIN

/GET CURRENT FIELD
/UPDATE TO NEXT FIELD
/MASK 0=0
/NEW FIELD POINTER

/MOVE TO 6=0

/COMPARE TO FIELDS PRESENT
/YES, GOOD FIELD
/NO BACK TO FIELD 0
/SET POINTER FOR NEW FIELD
/WHERE IS PROGRAM

/SET POINTER FOR FIELD JUST TESTED

/SAME MOVE
/MODIFIED TO CURRENT FIELD
/GET DATA WORD

/STORE DATA

/THIS THE GOOD ONE
/DID DATA CHANGE
/DATE O.K.
/RELOCATION ERROR
/TRY SAME WORD AGAIN
/UPDATE TO NEXT ADDRESS
/TRANSFER NEXT WORD

```

```

PAL10      V141      13-SEP-71      13131      PAGE 1-45      /CORE LOADED EXIT
4730      5664      /
4731      0000      /
4732      1371      /
4733      3176      /
4734      1376      /
4735      4447      /
4736      2176      /
4737      5334      /
4740      5731      /
4741      0000      /
4742      3176      /
4743      1374      /
4744      4447      /
4745      1373      /
4746      4447      /
4747      2176      /
4750      5343      /
4751      5741      /
4752      0000      /
4753      1174      /
4754      0172      /
4755      1077      /
4756      4447      /
4757      5752      /
4760      0000      /
4761      1314      /
4762      0173      /
4763      7010      /
4764      7012      /
4765      1077      /
4766      4447      /
4767      5760      /
4770      4550      /
4771      7771      /
4772      6201      /
4773      0212      /
4774      0215      /
4775      0170      /
4776      0252      /
5000      5000      /
5001      7604      /
5002      0104      /
5003      7640      /
5004      5210      /
5005      4446      /
5006      5625      /
5007      4264      /

/GET ASTRK CHAR;
/GET ASTRK CHAR;

/ASTRK;
/ASTRK;

/FLDNO;
/FLDNO;

/FLDHR;
/FLDHR;

/XFLDSW;
/XFLDSW;

/PAGE;
/PAGE;

/RARER;
/RARER;

JMP I RELOC
TAD KSTOP
DCA FLDCNT
TAD K252
JMS I XTYPE
ISZ FLDCNT
JMP I *3
JMP I ASTRK

DCA FLDCNT
TAD KCR
JMS I XTYPE
TAD KLF
JMS I XTYPE
ISZ FLDCNT
JMP I *5
JMP I LFCR

TAD FLBNUM
AND K0007
TAD K260
JMS I XTYPE
JMP I FLONO

TAD FLDGO
AND K0070
RAR
RTR
TAD K260
JMS I XTYPE
JMP I FLDHR

FLOSW, FLOSW
KSTOP, 7771
KDF, 6201
KLF, 0212
KCR, 0215
K0170, 0170
K252, 0252

PAGE
RARER,
LAS
AND SR01
SZA CLA
JMP I *4
JMS I XPRINT
EM12-1
JMS ROPRT

```

5010 7300
5011 1200
5012 5253

5013	0000	RALER, 0	CLA CLL
5014	7604	LAS	TAD
5015	0104	AND	JMP
5016	7640	SEA CLA	SR01
5017	5223	JMP	,+4
5020	4446	JMS I	XPRINT
5021	3644	EM13-1	ROPRT
5022	4264	JMS	CLA CLL
5023	7380	TAD	RALER
5024	1213	JMP	ROHLT
5025	5233	0	
5026	0000	RTRER, 0	LAS
5027	7604	LAS	AND
5030	0104	AND	SEA CLA
5031	7640	SEA CLA	JMP
5032	5236	JMP	,+4
5033	4446	JMS I	XPRINT
5034	5663	EM14-1	ROPRT
5035	4264	JMS	CLA CLL
5036	7380	TAD	RTRER
5037	1226	JMP	ROHLT
5040	5253	0	
5041	0000	RTLER, 0	LAS
5042	7604	LAS	AND
5043	0104	AND	SEA CLA
5044	7640	SEA CLA	JMP
5045	5231	JMP	,+4
5046	4446	JMS I	XPRINT
5047	5702	EM15-1	ROPRT
5050	4264	JMS	CLA CLL
5051	7380	TAD	RTLER
5052	1241	DCA	ROBACK
5053	3243	LAS	
5054	7604	LAS	AND
5055	0103	AND	SEA CLA
5056	7640	SEA CLA	JMP
5057	3262	TAD	,+3
5060	1263	HLT	ROBACK
5061	7402	JMP I	ROBACK
5062	5663	0	
5063	0000	ROBACK, 0	

5064	0000	ROPRT, 0	JMS I
5065	4446	JMS I	XPRINT
5066	5347	DM5-1	
5067	7340	CLA CLL	CMA
5070	0044	AND	LINKR
5071	5040	DCA	WD2

5072 7040 CMA
 5073 0041 AND
 5074 3037 DCA
 5075 4460 JMS I
 5076 4461 XLNKOU
 5077 7040 JMS I
 5078 0026 CMA
 5101 3040 AND
 5102 4460 DCA
 5103 7040 JMS I
 5104 0025 CMA
 5105 3037 AND
 5106 4461 DCA
 5107 4446 JMS I
 5110 5742 JMS I
 5111 5664 CRLF=1
 JMP I ROPRT

PAGE	TEXT	ARG1	ARG2	SIMULATED	ARG1+ARG2	ARG2+ARG1+1
5200						
5201						
5202						
5203						
5204						
5205						
5206						
5207						
5210						
5211						
5212						
5213						
5214						
5215						
5216						
5217						
5220						
5221						
5222						
5223						
5224						
5225						
5226						
5227						
5230						
5231						
5232						
5233						
5234						
5235						
5236						
5237						
5240						
5241						

PAL10 V141 13-SEP-71 13131 E 1-48

	DH2,	TEXT	ORIGINAL	SIMULATED	ACTUAL
5242					
5243					
5244					
5245					
5246					
5247					
5250					
5251					
5252					
5253					
5254					
5255					
5256					
5257					
5260					
5261					
5262					
5263					
5264					
5265					
5266					
5267					
5270					
5271					
5272					
5273					
5274					
5275					
5276					
5277					
5300					
5301					
5302					
5303					
5304					
5305					
5306					
5307					
5310					
5311					
5312					
5313					
5314					
5315					
5316					
5317					
5320					
5321					
5322					
5323					
5324					
5325					
5326					
5327					
5330					

	DH3,	TEXT	RANDC	RESULT
5242				
5243				
5244				
5245				
5246				
5247				
5250				
5251				
5252				
5253				
5254				
5255				
5256				
5257				
5260				
5261				
5262				
5263				
5264				
5265				
5266				
5267				
5270				
5271				
5272				
5273				
5274				
5275				
5276				
5277				
5300				
5301				
5302				
5303				
5304				
5305				
5306				
5307				
5310				
5311				
5312				
5313				
5314				
5315				
5316				
5317				
5320				
5321				
5322				
5323				
5324				
5325				
5326				
5327				
5330				

	DH4,	TEXT	BPOS	BNEG	RESULT
5242					
5243					
5244					
5245					
5246					
5247					
5250					
5251					
5252					
5253					
5254					
5255					
5256					
5257					
5260					
5261					
5262					
5263					
5264					
5265					
5266					
5267					
5270					
5271					
5272					
5273					
5274					
5275					
5276					
5277					
5300					
5301					
5302					
5303					
5304					
5305					
5306					
5307					
5310					
5311					
5312					
5313					
5314					
5315					
5316					
5317					
5320					
5321					
5322					
5323					
5324					
5325					
5326					
5327					
5330					

5331	4040
5332	4040
5333	4040
5334	4040
5335	0216
5336	0507
5337	4040
5340	4040
5341	4040
5342	4040
5343	4022
5344	0523
5345	2514
5346	2437
5347	3600
5350	3736
5351	1722
5352	1107
5353	1116
5354	0114
5355	4040
5356	4040
5357	4040
5360	0103
5361	2425
5362	0114
5363	3736
5364	0000
5365	3736
5366	4040
5367	4040
5370	0122
5371	0761
5372	4040
5373	4040
5374	4040
5375	4040
5376	4001
5377	2207
5400	6240
5401	4040
5402	4040
5403	4040
5404	0530
5405	2005
5406	0324
5407	0504
5410	4040
5411	4040
5412	4040
5413	0103
5414	2425
5415	0114
5416	3736
5417	0000

DH5, TEXT / ORIGINAL ACTUAL

DH6, TEXT / ARG1 ARG2 EXPECTED ACTUAL

V141 13-SEP-71 13131 1-50
EM1, TEXT /... SIMULATED ADD TEST FAILED/

PAL10 3736
5420 3736
5421 4040
5422 4040
5423 4023
5424 1115
5425 2514
5426 0124
5427 0504
5430 4001
5431 0404
5432 4024
5433 0523
5434 2440
5435 0601
5436 1114
5437 0504
5440 0000
5441 3736
5442 4040
5443 4040
5444 4023
5445 1115
5446 2514
5447 0124
5450 0504
5451 4022
5452 0114
5453 4024
5454 0523
5455 2440
5456 0601
5457 1114
5460 0504
5461 0000
5462 3736
5463 4040
5464 4040
5465 4023
5466 1115
5467 2514
5470 0124
5471 0504
5472 4022
5473 0122
5474 4024
5475 0523
5476 2440
5477 0601
5500 1114
5501 0504
5502 0000
5503 3736
5504 4040
5505 4040
5506 4023

EM2, TEXT /... SIMULATED RAL TEST FAILED/

EM3, TEXT /... SIMULATED RAR TEST FAILED/

EM4, TEXT /... SIMULATED RTL TEST FAILED/

5507 1115
5510 2514
5511 0124
5512 0504
5513 4022
5514 2414
5515 4024
5516 0523
5517 2440
5520 0601
5521 1114
5522 0504
5523 0000
5524 3736
5525 4040
5526 4040
5527 4023
5530 1115
5531 2514
5532 0124
5533 0504
5534 4022
5535 2422
5536 4024
5537 0523
5540 2440
5541 0601
5542 1114
5543 0504
5544 0000
5545 3736
5546 4040
5547 4040
5550 4023
5551 1115
5552 2514
5553 0124
5554 0504
5555 4002
5556 2327
5557 4024
5560 0523
5561 2440
5562 0601
5563 1114
5564 0504
5565 0000
5566 3736
5567 4040
5570 4040
5571 4022
5572 0116
5573 0417
5574 1540
5575 0104

EM5, TEXT /% SIMULATED RTR TEST FAILED/

EM6, TEXT /% SIMULATED BSH TEST FAILED/

EM10, TEXT /% RANDOM ADD TEST 1 FAILED/

PAL10 V141 13-SEP-71 13131 1-50
EM1, TEXT /... SIMULATED ADD TEST FAILED/

5420 3736
5421 4040
5422 4040
5423 4023
5424 1115
5425 2514
5426 0124
5427 0504
5430 4001
5431 0404
5432 4024
5433 0523
5434 2440
5435 0601
5436 1114
5437 0504
5440 0000
5441 3736
5442 4040
5443 4040
5444 4023
5445 1115
5446 2514
5447 0124
5450 0504
5451 4022
5452 0114
5453 4024
5454 0523
5455 2440
5456 0601
5457 1114
5460 0004
5461 0000
5462 3736
5463 4040
5464 4040
5465 4023
5466 1115
5467 2514
5470 0124
5471 0504
5472 4022
5473 0122
5474 4024
5475 0523
5476 2440
5477 0601
5500 1114
5501 0504
5502 0000
5503 3736
5504 4040
5505 4040
5506 4023

EM2, TEXT /... SIMULATED RAL TEST FAILED/

EM3, TEXT /... SIMULATED RAR TEST FAILED/

EM4, TEXT /... SIMULATED RTL TEST FAILED/

5507	1115		
5510	2514		
5511	0124		
5512	0504		
5513	4022		
5514	2414		
5515	4024		
5516	0523		
5517	2440		
5520	0601		
5521	1114		
5522	0504		
5523	0000		
5524	3736		
5525	4040		
5526	4040		
5527	4023		
5530	1115		
5531	2514		
5532	0124		
5533	0504		
5534	4022		
5535	2422		
5536	4024		
5537	0523		
5540	2440		
5541	0601		
5542	1114		
5543	0504		
5544	0000		
5545	3736		
5546	4040		
5547	4040		
5550	4023		
5551	1115		
5552	2514		
5553	0124		
5554	0504		
5555	4002		
5556	2327		
5557	4024		
5560	0523		
5561	2440		
5562	0601		
5563	1114		
5564	0504		
5565	0000		
5566	3736		
5567	4040		
5570	4040		
5571	4022		
5572	0116		
5573	0417		
5574	1540		
5575	0104		

EM5, TEXT /# SIMULATED RTR TEST FAILED/

EM6, TEXT /# SIMULATED BSW TEST FAILED/

EM10, TEXT /# RANDOM ADD TEST 1 FAILED/

5276 0440
 5277 2405
 5300 2324
 5301 4061
 5302 4006
 5303 0111
 5304 1405
 5305 0400
 5306 3736
 5307 4040
 5310 4040
 5311 4022
 5312 0116
 5313 0417
 5314 1540
 5315 0104
 5316 0440
 5317 2405
 5320 2324
 5321 4062
 5322 4006
 5323 0111
 5324 1405
 5325 0400
 5326 3736
 5327 4040
 5330 4040
 5331 4022
 5332 0116
 5333 0417
 5334 1540
 5335 2201
 5336 2240
 5337 2405
 5340 2324
 5341 4006
 5342 0111
 5343 1405
 5344 0400
 5345 3736
 5346 4040
 5347 4040
 5350 4022
 5351 0116
 5352 0417
 5353 1540
 5354 2201
 5355 1440
 5356 2405
 5357 2324
 5360 4006
 5361 0111
 5362 1405
 5363 0400
 5364 3736

EM11, TEXT /... RANDOM ADD TEST 2 FAILED/

EM12, TEXT /... RANDOM RAR TEST FAILED/

EM13, TEXT /... RANDOM RAL TEST FAILED/

EM14, TEXT /... RANDOM RTL TEST FAILED/

5665 4040
5666 4040
5667 4022
5670 0116
5671 0417
5672 1540
5673 2224
5674 1440
5675 2485
5676 2324
5677 4006
5700 0111
5701 1405
5702 0400
5703 3736
5704 4040
5705 4040
5706 4022
5707 0116
5710 0417
5711 1540
5712 2224
5713 2240
5714 2485
5715 2324
5716 4006
5717 0111
5720 1405
5721 0400
5722 3736
5723 2311
5724 1501
5725 0400
5726 3736
5727 2311
5730 1522
5731 1724
5732 0000
5733 3736
5734 0603
5735 2400
5736 3736
5737 2201
5740 1604
5741 1715
5742 0000
5743 3736
5744 0000
5745 3736
5746 4004
5747 0124
5750 0140
5751 0522
5752 2217
5753 2237

EMIS, TEXT /.. RANDOM RTR TEST FAILED?

OK1, TEXT /..SIMAD/

OK2, TEXT /..SIMROT/

OK3, TEXT /..FCT/

OK4, TEXT /..RANDOM/

CRLF, TEXT /..//

DATE, TEXT /.. DATA ERROR...//

PAL10

V141

13-SEP-71

13131

E 1-54

5754 3600
 5755 7777
 5756 4005
 5757 3024
 5760 0516
 5761 0405
 5762 0440
 5763 0201
 5764 1613
 5765 2340
 5766 1706
 5767 4015
 5770 0515
 5771 1722
 5772 3140
 5773 2417
 5774 4002
 5775 0116
 5776 1340
 5777 0000

BKMS,

7777
 4005
 3024
 0516
 0405
 0440
 0201
 1613
 2340
 1706
 4015
 0515
 1722
 3140
 2417
 4002
 0116
 1340
 0000

/TEXT FOR EXTENDED BANKS OF MEMORY TO BANK

/RESTORE BINARY LOADER AND START LOADER

*7600

7600
 7601
 7602
 7603

CLA CLL
 TAD BIN
 DCA TSTA2
 JMP TSTA2
 S

Label	Value	Label	Value	Label	Value	Label	Value
A1	3263	DH5	5350	FCT12	2667	K0010	0113
A10RA2	0027	DH6	5365	FCT2	2043	K0020	0112
A2	3264	EM1	5420	FCT3	2076	K0040	0111
ABNOT	3754	EM10	5566	FCT4	2200	K0070	0173
AD1	2127	EM11	5606	FCT5	2232	K0077	1645
AD10	2715	EM12	5626	FCT6	2270	K0100	0110
AD11	2716	EM13	5635	FCT7	2400	K0170	4775
AD12	2717	EM14	5664	FCT8	2436	K0200	0107
AD2	2130	EM15	5703	FCT9	2472	K0400	0106
AD3	2131	EM2	5441	FCTHLT	3207	K0700	3266
AD4	2324	EM3	5462	FCTOK	3221	K1000	0105
AD5	2325	EM4	5503	FLOCHK	4600	K2000	0104
AD6	2326	EM5	5524	FLODCNY	0176	K212	1690
AD7	2326	EM6	5545	FLODF	4643	K215	1651
AD8	2527	ENCAR	0244	FLODEX	4627	K240	0076
AD9	2530	ENCAR1	0253	FLODFND	4631	K252	4776
ADA1	0021	ENDBSW	1277	FLODFRM	4712	K260	0077
ADA2	0022	ENDFCT	3200	FLOGO	4714	K261	0100
ADD	0274	ENDROI	1303	FLOGR	4760	K336	1647
ADDERR	0400	ENRN	4566	FLOGNO	4752	K400	0103
ADHLT	0556	ERRUT1	3744	FLOGNUM	0174	K600	0101
ADOUT	3227	ERROR1	0377	FLOGRM1	4717	K6060	3267
ADPRT	0417	ERROR2	1046	FLOGSAV	0175	KCDF	4772
ADT	0551	ERROT	1026	FLOGSW	4590	KCR	4774
ADFLG	0035	ERRT	2023	GOTEST	0177	KLF	4773
ADOUT	0467	FCL1	2612	HALT2	3027	KSTOP	4791
ADTBT	3616	FCL10	2612	HALTA	0477	KXXX	0190
ANEG	3747	FCL11	2644	HALTB	1063	LPOR	4741
APOS	3747	FCL12	2677	HLTA	0404	LINK1	0052
ARG1	0023	FCL2	2056	HLTB	1052	LINK2	0034
ARG2	0024	FCL3	2107	INS1	0136	LINKR	0044
ARG3	0024	FCL4	2213	INS10	0146	LNKRC	0045
ASTRK	4731	FCL5	2246	INS11	0147	LNKOUT	0504
BIN	0155	FCL6	2302	INS12	0150	LOOP	0546
BKMS	5755	FCL7	2412	INS13	0151	LOOP1	0552
BNEG	5755	FCL8	2450	INS14	0152	M4	1070
BPOS	3751	FCL9	2503	INS15	0153	M40	1646
BSH	3750	FCS1	2005	INS16	2153	MASK	3752
BSWIAB	7002	FCS10	2601	INS2	2132	MODNEG	3632
CAF	1660	FCS11	2635	INS3	0137	MOVMSK	3667
CARRY	6007	FCS12	2670	INS4	0140	MGA	7501
CBTST1	0030	FCS2	2044	INS5	0141	MQL	7421
CBTST2	3640	FCS3	2077	INS6	0142	MPP	1643
CHAR	3654	FCS4	2201	INS7	0143	NIBIT	1244
CHAR	0036	FCS5	2233	INS8	0144	NBIT	1207
CNTR1	0020	FCS6	2271	INS9	0145	NERROR	0067
COMHOT	1000	FCS7	2401	K0	0171	NEWLNK	1044
CRLF	5743	FCS8	2437	K0001	0116	NLOOP	3054
DATE	5745	FCS9	2473	K0002	0115	NMASK	3753
DATER	3037	FCT	2000	K0003	3542	NOTAC	3755
DH1	5200	FCT1	2000	K0004	0114	NXBIT	0527
DH2	5245	FCT10	2600	K0007	0172	NXTADD	0365
DH3	5274	FCT11	2634				
DH4	5317						

NXTBT	3612	RNDHLI	4540	SIMRTL	0653	XFLDSH	4770
NXTCAR	0234	RNDOK	4563	SIMRTL	0700	XHALT2	0066
NXTROT	1031	ROBACK	5063	SKHLT	3502	XLNK04	0060
OK1	5722	ROHLT	5053	SP1	0545	XLOOP	0067
OK2	5726	ROPRT	5064	SR00	0103	XLOOP1	0075
OK3	5733	ROTDNE	1323	SR01	0104	XLOOP2	0074
OK4	5736	ROTHLT	1327	SR02	0105	XM2	1450
OR1	1225	ROTPRT	1071	SR03	0106	XM3	1451
OUT	3224	RRAC	0031	SR04	0107	XM4	1452
OUT1	0520	RRAL	0605	SR05	0110	XM5	1453
OUT1A	0542	RRAR	0632	SR06	0111	XM6	1454
POINT1	0011	RRLNK	0033	SR07	0112	XNXTAD	0416
POINT2	0012	RRTL	0037	SR08	0113	XNXTRO	0057
PRINT	1600	RRTR	0657	SR09	0114	XOR1	3701
R1	1400	RSIMAD	0704	SR10	0115	XOR2	3715
R2	1410	RTLER	5041	SR11	0116	XORALL	0260
R2A	3541	RTLRTAB	4400	SROTAL	1200	XPRINT	0046
R3	1420	RTLTAB	1160	SROTOK	1342	XR1	0752
R4	1430	RYRER	5026	START	0156	XR2	0753
R5	1440	RTRR	4455	SUM1	0051	XR3	0754
RAC	0023	RTRTAB	1141	SUM2	0033	XR4	0755
RALER	5013	SADOK	0570	TEMP1	0037	XR5	0756
RALR	4255	SAMEA	3730	TEMPAC	0025	XRALR	4334
RALTAB	0757	SAMEAS	3000	TEMPL	0026	XRALTA	0052
RANDA	0041	SAVREG	3017	TSTAB	7775	XRAND	0093
RANDB	0042	SBSW	1236	TSTA1	7776	XRARR	4335
RANDC	0043	SEQ	0154	TSTA2	7777	XRARTA	0102
RANDD	3512	SEQ1	0122	TSTA3	0000	XRHD	0090
RARER	5000	SEQ10	0133	TSTA4	0001	XRN1ER	3446
RARR	4200	SEQ11	0134	TSTA5	0002	XRN2ER	3756
RBSH	0731	SEQ12	0135	TSTA6	0003	XROTDN	0777
RELOC	4664	SEQ2	0123	TSTA7	0004	XRTL	4571
RETI	2025	SEQ3	0124	TSTIND	0010	XRTLTA	0053
RETI0	2616	SEQ4	0125	TYBIT	0540	XRTTR	4570
RETI1	2031	SEQ5	0126	TYLNK	0513	XRTRTA	0054
RETI2	2701	SEQ6	0127	TYPE	1652	XSBSW	0776
RETI3	2060	SEQ7	0130	TYP52	3056	XSTAB	0051
RETI4	2111	SEQ8	0131	W1	1617	XSTAB	0070
RETI5	2215	SEQ9	0132	W2	0037	XSTA1	0071
RETI6	2306	SHLT	4033	WD1	0040	XSTA2	0072
RETI7	2420	SIMAC	0025	WD2	0037	XTYPE	0007
RETI8	2455	SIMAD	0204	WDOUT	0040	XWDOUT	0061
RETI9	2510	SIMBSH	0725	XADD	0523		
RHD	1133	SIMLNK	0026	XADOUT	0415		
RHFLG	0035	SIMR	0577	XAMEA	3073		
RHOUT	1131	SIMRAL	0601	XAMEAS	0063		
RLNK	0024	SIMRAR	0626	XAVREG	0062		
RN1ER	3447	SIMRO1	0600	XBSWTA	0064		
RN2ER	4000	SIMRO2	0625	XCOMRO	0055		
RNAD1	3400	SIMRO3	0652	XDATER	0056		
RNAD2	3600	SIMRO4	0677	XFLDCK	4572		
		SIMRO5	0724				

ERRORS DETECTED: 0
LINKS GENERATED: 3
RUN-TIME: 32 SECONDS
3K CORE USED