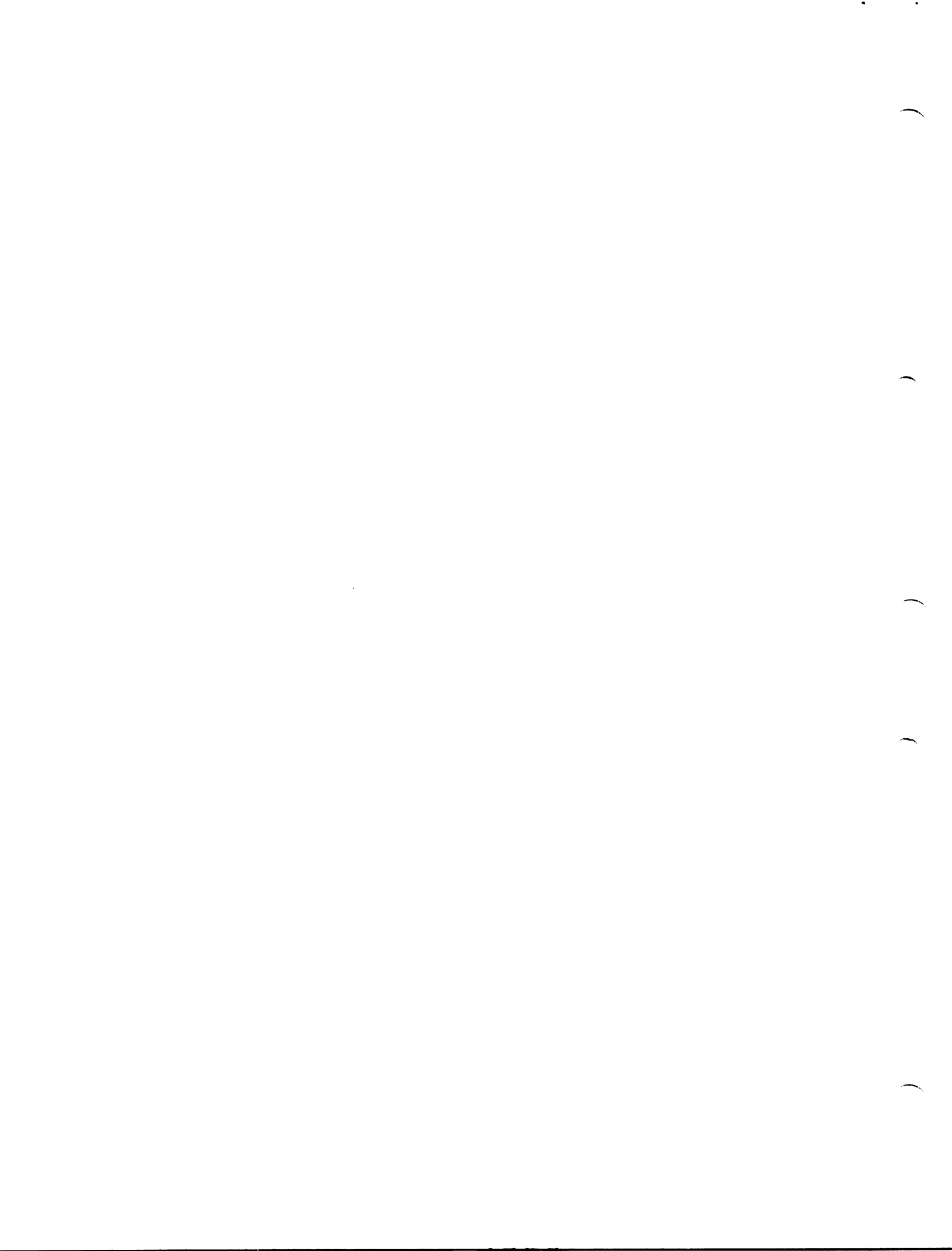


IDENTIFICATION

PRODUCT CODE: MAINDEC-8E-DØEA-D (D)
PRODUCT NAME: RANDOM TAD TEST
DATE CREATED: NOVEMBER 2, 1970
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: MICHAEL DAVIS

COPYRIGHT © 1970
DIGITAL EQUIPMENT CORPORATION



1. ABSTRACT

THIS PROGRAM TESTS THE TAD INSTRUCTING OF THE PDP-8E. THE TAD INSTRUCTION, INSTRUCTION ADDRESS, OPERAND ADDRESS AND BOTH OPERANDS ARE PRODUCED BY RANDOM NUMBER GENERATORS.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-8E EQUIPPED WITH AT LEAST 4K OF MEMORY,

TELETYPE.

2.2 STORAGE

THE PROGRAM IS LOADED INTO LOCATIONS 6600 THRU 7577, THE TEST AREA IS 0000-6577. TEMPORARY STORAGE LOCATIONS ARE LOCATED ON PAGE 0.

2.3 PRELIMINARY PROGRAMS

MAINDEC-8E-D0AA, D0BA, D0CA, D0DA

3. LOADING PROCEDURE

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

4. STARTING PROCEDURE

4.1 STARTING ADDRESS

0200

4.2 CONTROL SWITCH SETTINGS

SR00=1, SUPPRESS HALT ON ERROR
SR03=1, SUPPRESS END OF PASS TYPEOUT
SR09=1, HOLD DATA 1 CONSTANT
SR10=1, HOLD DATA 2 CONSTANT
SR11=1, HOLD INSTRUCTION CONSTANT

4.3 OPERATOR ACTION

4.3.1 SET SR TO 1200

4.3.2 PRESS LOAD ADDRESS SWITCH

4.3.3 SET SR TO 2000

4.3.4 PRESS CLEAR AND CONTINUE SWITCHES

5. OPERATING PROCEDURE

SAME AS 4.

6. ERRORS

6.1 ERROR HALT

IF THE RESULTS OF THE TAD INSTRUCTION ARE INCORRECT, THAT IS IF THE ACTUAL AND SIMULATED LINKS, OR THE ACTUAL AND SIMULATED SUMS DO NOT AGREE, THE PROGRAM WILL HALT AT 7407 WITH DATA1 IN THE AC.

DEPRESS CONTINUE TO DISPLAY DATA2 IN THE AC.
DEPRESS CONTINUE TO DISPLAY TAD INSTRUCTION IN AC.
DEPRESS CONTINUE TO DISPLAY INSTRUCTION ADDRESS IN AC.
DEPRESS CONTINUE TO DISPLAY DATA2 ADDRESS IN AC.
DEPRESS CONTINUE TO DISPLAY INDIRECT POINTER (USED BY INDIRECT TAD) IN AC.
DEPRESS CONTINUE TO RESUME TEST

6.2 ERROR RECOVERY

SEE 6.1

SET SR00=1 TO PREVENT HALT AFTER ERROR,
 SET SR03=1 TO SUPPRESS END OF PASS TYPEOUT,
 SET SR09-SR11=1 TO HOLD INSTRUCTION AND DATA CONSTANT,

7. RESTRICTIONS

NONE

8. EXECUTION TIME

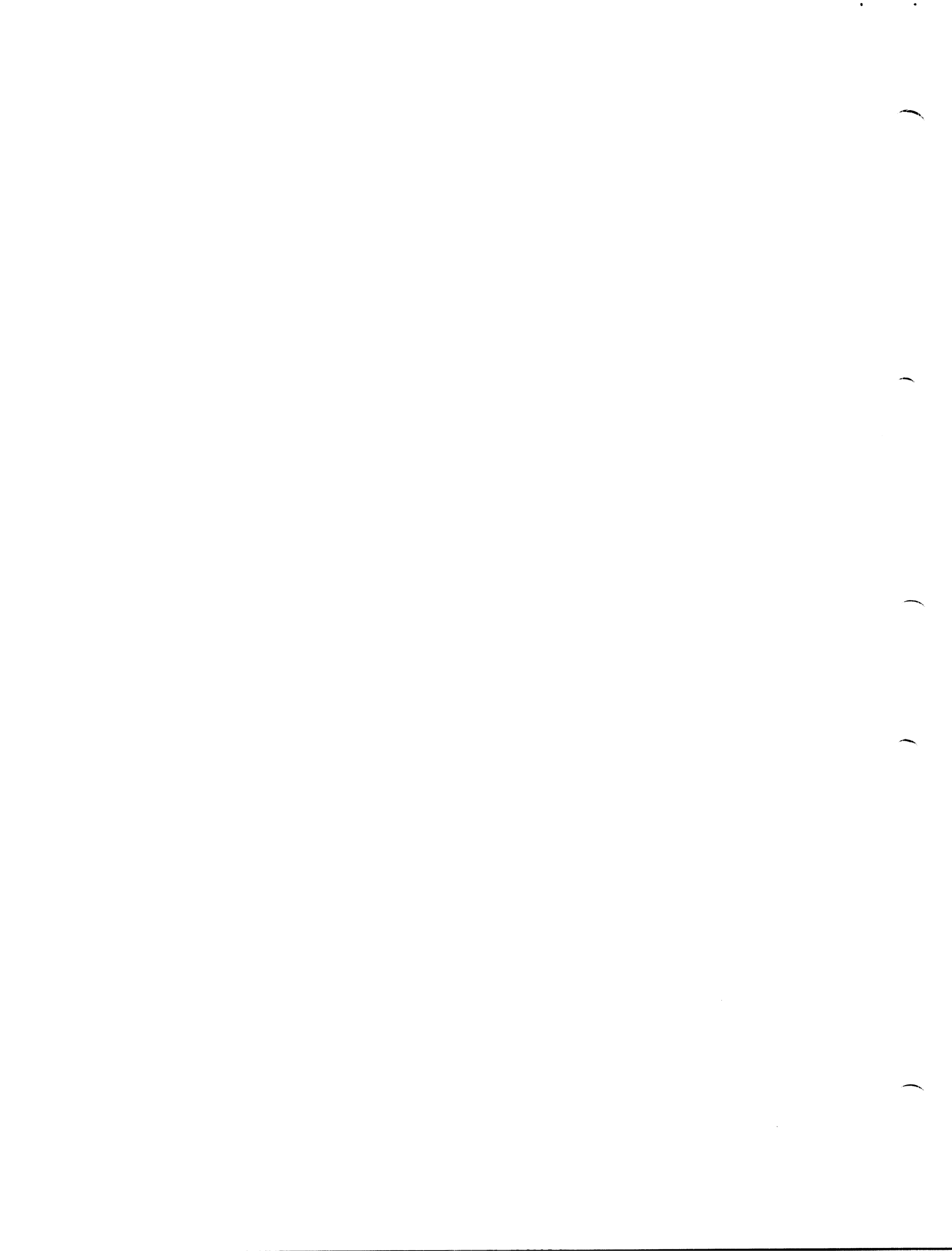
THE PROGRAM PERFORMS 4096 RANDOM TESTS IN APPROXIMATELY 5 SECONDS. THE PROGRAM WILL TYPE "T" AFTER EACH 4096 RANDOM TESTS UNLESS SR03=1.

9. PROGRAM DESCRIPTION

THE PROGRAM IS LOADED INTO LOCATIONS 6600-7577, WITH TEMPORARY STORAGE LOCATIONS ON PAGE 0.

THE PROGRAM USES SEPARATE RANDOM NUMBER GENERATORS TO GENERATE THE TAD INSTRUCTION, INSTRUCTION AND DATA ADDRESSES, AND THE TWO ARGUMENTS TO BE "TADDED". THE INSTRUCTIONS AND DATA ARE STORED IN THEIR PREVIOUSLY GENERATED ADDRESSES. THE PROGRAM TRANSFERS TO THE LOCATION OF THE INSTRUCTION AND EXECUTES IT. THE PROGRAM THEN TRANSFERS TO A COMPARISON ROUTINE WHERE THE ACTUAL RESULT OF THE TAD INSTRUCTION IS COMPARED TO A SIMULATED TAD. THE SIMULATOR IS SIMILAR TO THE ONE USED IN MAINDEC-8E-D0CA-D; NO TADS ARE USED IN THE PROGRAM ITSELF.

AFTER 4096 TESTS, THE PROGRAM TYPES "T" AND CONTINUES TESTING.



```

/ /RANDOM TAD TEST
/ /COPYRIGHT 1970, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
/ /V 82 07552
/ /
/ /TEMPORARY STORAGE LOCATIONS
/ /

```

```

0000
0000
0001 5001 JMP
0002 0002
0003 0003
0004 0004
0005 0005
7501 MGA=7501
7421 MQL=7421
6007 CAF=6007
0200
0200 0200
0201 5602
0202 6600

```

ADVANCE COPY

document subject to change
without notice.

```

*200 START,
CAF JMP I ,*1
STARTL

```

```

/ /GENERATE TEST INSTRUCTION AND DATA
/ /

```

```

6600
6601 7300
6602 3376
6603 7604
6604 0371
6605 7640
6606 5224
6607 4746
6610 3355
6611 7040
6612 0001
6613 3353
6614 7041
6615 0002
6616 3354
6617 7040
6620 0003
6621 3356
6622 7040
6623 0004
6624 3357
6625 7604
0372

*6600 STARTL,
TEST1L,
CLL CTR1
DCA AND
LAS SR11
SEA CLA
JMP TDATA
JMS I TGENL
DCA TIFLGL
CMA INSTL
AND TINSTL
DCA INADDL
CMA TINADL
AND DATADL
DCA TDATA
CMA PADDL
AND TPADDL
LAS SR10
AND

/ /CLEAR PASS COUNTER
/ /TEST SR11
/ /IS SR11=1
/ /SR11=1, DO NOT GENERATE INSTRUCTION
/ /GENERATE INSTRUCTION
/ /SAVE INDIRECT FLA
/ /GET INSTRUCTION
/ /SAVE IT
/ /GET INSTRUCTION ADDRESS
/ /SAVE IT
/ /GET DATA ADDRESS
/ /SAVE IT
/ /GET INDIRECT TO DATA
/ /SAVE IT
/ /TEST SR10

```

```

6626 7647 SZA CLA /IS SR10=1
6627 5234 JMP TDA2L /SR10=1, DO NOT GENERATE DATA1
6630 7040 CMA /GENERATE RANDOM NUMBER
6631 0360 AND TDA1L
6632 4752 JMS I TRANDL
6633 3360 DCA TDA1L
6634 7604 LAS SR09 /TEST SR09
6635 0373 AND SETTL /IS SR09=1
6636 7640 SZA CLA /SR09=1, DO NOT GENERATE DATA2
6637 5244 JMP TDA2L /GENERATE RANDOM NUMBER
6640 7040 CMA
6641 0361 AND TRANDL
6642 4752 JMS I
6643 3361 DCA TDA2L

```

```

/SET UP INSTRUCTION AND DATA AT TEST ADDRESS
/ALONG WITH RETURN TO THIS ROUTINE
/
SETTL,
6644 7340 CLA CLL CMA /GET INSTRUCTION
6645 0353 AND TINSTL /STORE IN TEST LOCATION
6646 3754 DCA I TINADL
6647 7040 CMA /GET INDIRECT FLAG
6650 0355 AND TIFLGL /IS INSTRUCTION INDIRECT
6651 7650 SNA CLA DURL /NO, GET DATA
6652 5302 JMP /ADDRESS IS INDIRECT
6653 7040 CMA /IS ADDRESS AUTO-INDEX REGISTER
6654 0356 AND TDATA1 /NO
6655 0367 AND T7760
6656 7640 SZA CLA NOTAUT
6657 5276 JMP /ADDRESS IS AUTO-INDEX REGISTER
6660 7040 CMA /DECREMENT POINTER TO DATA
6661 0356 AND TDATA1 /STORE IN TEST LOCATION
6662 0375 AND KI0 /GET DATA
6663 7650 SNA CLA NOTAUT /STORE IN TEST LOCATION
6664 5276 JMP /ADDRESS IS AUTO-INDEX REGISTER
6665 7040 CMA /DECREMENT POINTER TO DATA
6666 0357 AND TPADDL /STORE IN TEST LOCATION
6667 7041 CIA /GET DATA
6670 7042 CMA /STORE IN TEST LOCATION
6671 3756 DCA I TDATA1 /GET DATA
6672 7040 CMA /STORE IN TEST LOCATION
6673 0360 AND TDA1L
6674 3757 DCA I TPADDL
6675 5325 JMP DOTSTL
6676 7040 CMA /ADDRESS IS AUTO-INDEX REGISTER
6677 0357 AND TPADDL /DECREMENT POINTER TO DATA
6700 3756 DCA I TDATA1 /STORE IN TEST LOCATION
6701 5272 JMP .-7
6702 7040 CMA
6703 0360 AND TDA1L /GET DATA
6704 3756 DCA I TDATA1 /STORE IN TEST LOCATION

```


/SIMULATE "TAD"

```

6705 7340 DOTSTL, CLA CLL CMA /GET DATA1
6706 0360 AND TDA1L /SAVE IN MQ
6707 7421 MQL CMA /GET DATA2
6710 7040 AND TDA2L /DO SIMULATION
6711 0361 JMS I TSIMAD /SAVE ANSWER
6712 4751 DCA TSIMAC /SAVE LINK
6713 3363 RAR
6714 7010 DCA TSIML
6715 3362

```

/GO TO TEST

```

6716 7040 DOANDL, CMA /GET RETURN ADDRESS
6717 0347 AND TRETTL /SAVE
6720 3000 DCA RETURN /GET INSTRUCTION ADDRESS
6721 7040 CMA AND TINADL /INCREMENT
6722 0354 AND IAC /IS IT 0
6723 7001 SNA /YES, GENERATE NEW INFORMATION
6724 7450 JMP DCA /NO, SAVE
6725 5202 DCA /GET RETURN INSTRUCTION
6726 3345 CMA /PUT IN TEST LOCATION
6727 7040 AND I /GET DATA2
6730 0366 DCA I CLL CMA /EXECUTE "TAD"
6731 3745 AND JMP I /RETURN HERE AFTER EXECUTION
6732 7140 /
6733 0361 /
6734 5754 /

```

/RETURN HERE AFTER EXECUTION

```

6735 3364 TRETUL, DCA TAC /SAVE AC
6736 7010 RAR /SAVE LINK
6737 3365 DCA I TLINK /COMPARE REAL AND SIMULATED ADDITIONS
6740 4774 JMS I TCOMAD /END OF PASS, 4096 TEST COMPLETE
6741 2376 ISZ CNTR1
6742 5202 JMP TEST1L
6743 4750 JMS I TEPASL
6744 5202 JMP TEST1L

```

```

6745 0000 /
6746 7000 /
6747 6735 /
6750 7442 /
6751 7203 /
6752 7430 /
6753 0000 /

```

6754	0000	TINADL, 0
6755	0000	TIFLGL, 0
6756	0000	TDATAL, 0
6757	0000	TPADDL, 0
6760	0021	TDA1L, 21
6761	0037	TDA2L, 37
6762	0000	TSIML, 0
6763	0000	TSIMAC, 0
6764	0000	TAC, 0
6765	0000	TLINK, 0
6766	5400	T5400L, 5400
6767	7760	T7760, 7760
6770	7770	T7770, 7770
6771	0001	SR11, 1
6772	0002	SR10, 2
6773	0004	SR09, 4
6774	7313	TCOMAD, COMAD
6775	0010	K10, 10
6776	0000	CNTR1, 0

/GENERATE INSTRUCTIONS AND ADDRESSES

PAGE
GENL.

7000 0000

/GENERATE "AND" INSTRUCTION

7001	7040	GANDL,	CMĀ	R1L		
7002	0350		AND	SRANDL		/GENERATE RANDOM NUMBER
7003	4762		JMS I	R1L		/SAVE NUMBER
7004	3350		DCA			
7005	7040		CMA	R1L		/GENERATE OP CODE
7006	0350		AND			
7007	7421		MQL			
7010	7040		CMĀ	K1000		
7011	0365		AND			
7012	7501		MQA	K1777		/SAVE INSTRUCTION
7013	0352		AND	INSTL		
7014	3001		DCA			
7015	7040		CMA	INSTL		/GET INSTRUCTION
7016	0001		AND	K0177L		/EXTRACT PAGE ADDRESS OF INSTRUCTION
7017	0355		AND	TEMP3L		/SAVE PAGE ADDRESS OF INSTRUCTION
7020	3361		DCA			

/GENERATE ADDRESS FOR INSTRUCTION

7021	7040	GANADL,	CMA	R2L		
7022	0353		AND	SRANDL		/GENERATE RANDOM NUMBER
7023	4762		JMS I	R2L		/SAVE NUMBER
7024	3353		DCA			
7025	7040		CMA	R2L		/IS ADDRESS WITHIN LIMITS
7026	0353		AND	LIMIT		
7027	4777		JMS			

```

7030 JMP GANADL /NO, GENERATE NEW ADDRESS
7031 CMA /IS ADDRESS ON PAGE 0
7032 AND R2L /GET PAGE ADDRESS OF INSTRUCTION
7033 AND P0L /GET DIFFERENCE BETWEEN PAGE ADDRESSES
7034 SZA CLA /IS DIFFERENCE >2
7035 JMP PAGADL /NO
7036 CMA /GET PAGE ADDRESS OF INSTRUCTION
7037 AND R2L /GET DIFFERENCE BETWEEN PAGE ADDRESSES
7040 JMS ABSL /IS DIFFERENCE >2
7041 SMA CLA /NO
7042 JMP GANADL
7043 JMP PAL

```

```

7044 PAGADL, CMA /GET INSTRUCTION
7045 AND INSTL /IS PAGE BIT SET
7046 AND K200L /NO, USE ADDRESS AS IS
7047 SNA CLA PAL /PAGE BIT SET, EXTRACT PAGE ADDRESS FOR INSTRUCTION
7050 JMP PAL /TEST FOR INTERFERENCE
7051 AND R2L /MAKE SURE DATA WILL
7052 AND K0177L /NOT BE STORED IN LOCATION 0
7053 AND PAGAL /LOCATION ZERO, TRY AGAIN
7054 JMP PAGAL /USE ADDRESS AS IS
7055 CMA
7056 AND TEMP3L
7057 SNA CLA GANDL
7060 JMP GANDL
7061 CMA R2L
7062 AND INADDL
7063 DCA PADL

```

/GENERATE ADDRESS FOR DATA

```

7064 DAADL, CMA /GET INSTRUCTION
7065 AND INSTL /IS PAGE BIT OF INSTRUCTION SET
7066 AND K200L /NO, USE PAGE ADDRESS BITS OF INSTRUCTION FOR DATA ADDRESS
7067 SNA CLA P0AL /EXTRACT PAGE OF INSTRUCTION ADDRESS
7070 JMP CMA
7071 AND INADDL
7072 AND P0L
7073 AND TEMP3L
7074 MQL DATADL
7075 CMA
7076 AND
7077 MQA
7100 DCA
7101 CMA INDIRL,
7102 AND INSTL
7103 AND K400L
7104 SZA CLA
7105 JMP PADL

```

/IS INSTRUCTION INDIRECT
/YES, INSTRUCTION IS INDIRECT

7106 5600 JMP I GENL /EXIT
 7107 7047 CMA /USE PAGE ADDRESS OF INSTRUCTION
 7110 0361 AND /AS DTAT ADDRESS
 7111 3003 DCA
 7112 5301 JMP INDIRL

/GENERATE INDIRECT ADDRESS FOR DATA
 /GENERATE RANDOM NUMBER
 7113 7047 CMA
 7114 0360 AND R3L
 7115 4762 JMS I SRANDL
 7116 3360 DCA R3L
 7117 7040 CMA
 7120 0360 AND R3L
 7121 4777 JMS LIMIT
 7122 5313 JMP PADL
 7123 7040 CMA
 7124 0002 AND INADDL
 7125 4775 JMS ABSL1
 7126 7700 SMA CLA
 7127 5313 JMP PADL
 7130 7040 CMA
 7131 0003 AND DATAOL
 7132 -775 JMS ABSL1
 7133 7700 SMA CLA
 7134 5313 JMP PADL
 7135 7040 CMA
 7136 0360 AND R3L
 7137 7041 CIA
 7140 7040 CMA
 7141 7650 SMA CLA

/IS ADDRESS WITHIN LIMITS
 /NO, TRY AGAIN
 /GET INSTRUCTION ADDRESS
 /GENERATE DIFFERENCE BETWEEN ADDRESSES
 /DO INSTRUCTION AND ADDRESS INTERFERE
 /YES
 /NO, TEST DATA ADDRESS AS ABOVE

/ADDRESSES DO NOT INTERFERE
 /WILL LOCATION BE 0 IF DECREMENTED

/YES, GENERATE NEW POINTER
 /EXIT

JMP PADL
 CMA
 AND R3L
 DCA PADDL
 CMA
 JMP I GENL
 /
 /
 /

7150 0001 R1L,
 7151 0003 K3L,
 7152 1777 K1777,
 7153 0005 R2L,
 7154 7600 POL,
 7155 0177 K0177L,
 7156 0400 K400L,
 7157 0200 K200L,
 7160 0015 R3L,
 7161 0000 TEMP3L,
 7162 7430 SRANDL,
 7163 7200 SRIMAD, RSIMAD

LIML, 1201
K1000, 1000

7164 1201
7165 1000

/SIMULATED ADDITION

7175 7507
7176 7474
7177 7303
7200 7200
7200 0000
7201 3344
7202 7501
7203 3343

PAGE
RSIMAD,

0 ARG2 /SAVE ARGUMENTS
DCA
MGA
DCA ARG1

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

/FORM OR OF ARG1 WITH ARG2

7204 7340
7205 0343
7206 7421
7207 7040
7210 0344
7211 7501
7212 3345

SIMAD,

CLA CLL CMA /LOAD AC WITH ARG1
AND ARG1 /PLACE IN MQ
MQL
CMA ARG2 /LOAD AC WITH ARG2
AND /FORM ARG1 OR ARG2
MGA A10RA2 /SAVE ARG1 OR ARG2
DCA

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

7213 7501
7214 7040
7215 0344
7216 7421
7217 7040
7220 0344
7221 7040
7222 0343
7223 7501
7224 3346
7225 3347

MGA /GET ARG1 FROM MQ
CMA /FORM NOTARG1
AND ARG2 /AND WITH ARG2 TO GET ARG2 AND NOTARG1
MQL /SAVE IN MQ
CMA
AND /LOAD AC WITH ARG2
CMA /FORM NOTARG2
AND ARG1 /AND WITH ARG1 TO GET ARG1 AND NOTARG2
MGA /OR WITH ARG2 AND NOTARG1
DCA /TO GET ARG1 XOR ARG2
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

```

7226 7040 CMA
7227 0343 AND ARG1 /LOAD AC WITH ARG1
7230 0344 AND ARG2 /AND WITH ARG2
7231 7450 SNA /ARE THERE ANY CARRIES
7232 5274 JMP ENDSIM /NO, TERMINATE SIMULATION
/GENERATE CARRIES
/
7233 7421 MQL MQL /SAVE FIRST CARRIES
7234 7521 MQA MQL A10RA2 /GET CARRIES FROM MQ
7235 0345 AND /AND WITH A10RA2 TO SEE IF MORE CARRIES ARE GENERATED
7236 7450 SNA ENCAR /ARE THERE ANY MORE CARRIES
7237 5244 JMP RAL /NO, END SIMULATION OF CARRIES
7240 7104 CLL MQL /PROPIGATE CARRIES
7241 7521 MQA MQL /GET PREVIOUS CARRIES FROM MQ, SAVE NEW CARRIES
7242 7501 MQA /OR NEW CARRIES WITH PREVIOUS CARRIES
7243 5234 JMP NXTCAR /CONTINUE
/TEST FOR CARRY INTO LINK
/
7244 7501 MQA /GET CARRIES
7245 0345 AND A10RA2 /AND WITH A10RA2
7246 0350 AND K4000 /TEST BIT 00
7247 7450 SNA /IS BIT 00 1
7250 5253 JMP ENCAR1 /NO, CARRIES DID NOT PROPAGATE INTO LINK
7251 3347 DCA SIMLNK /YES, SAVE CARRY INTO LINK
7252 5260 JMP XORALL /COMPLETE SIMULATION
7253 7130 CLL CML RAR /SET AC=4000
7254 0343 AND ARG1 /AND WITH ARG1
7255 0344 AND ARG2 /AND WITH ARG2 TO SEE IF ORIGINAL
7256 7440 SZA /NUMBERS GENERATED CARRY INTO LINK
7257 3347 DCA SIMLNK /SAVE SIMULATED LINK

```

```

/FORM XOR OF ARG1, ARG2, AND CARRIES
/TO GET FINAL SIMULATED SUM
/
7260 7501 MQA /SAVE SIMULATED CARRIES
7261 3351 DCA CARRY
7262 7501 MQA
7263 7040 CMA
7264 0346 AND /FORM A10RA2 AND NOTCARRY
7265 7421 MQL /SAVE IN MQ
7266 7040 CMA
7267 0346 AND SIMAC
7270 7040 CMA
7271 0351 AND CARRY
7272 7501 MQA
7273 3346 DCA SIMAC
7274 7340 CLA CLL CMA /FORM CARRY AND NOTA10RA2
7275 0347 AND SIMLNK /OR WITH CONTENTS OF MQ
7276 7640 SZA CLA
7277 7020 CML

```

7300 7040 CMA /TO GET FINAL SIMULATED SUM
7301 5346 AND SIMAC
7302 5600 JMP I RSIMAD

/TEST ADDRESS

7303 0000 LIMIT, /
7304 7421 MQL /SAVE ARGUMENT IN MQ
7305 7040 CMA /LOAD AC WITH LIMIT
7306 0777 AND LIML /DO ADDITION
7307 4200 JMS RSIMAD /LINK SET IF NUMBER TO LARGE
7310 7620 SNL CLA /NUMBER OK
7311 2503 ISZ LIMIT
7312 5703 JMP I LIMIT

/COMPARE SIMULATED AND REAL RESULT

7313 0000 COMAD, /
7314 7340 CLA CLL CMA /GET SIMULATED RESULTANT LINK
7315 0776 AND TSIML
7316 7640 SZA CLA /COMPARE TO REAL LINK
7317 7020 CML
7320 7040 CMA TLINK
7321 0775 AND CLA /IF SAME, LINK=0
7322 7640 SZA CLA /NOT THE SAME, ERROR
7323 7020 CML
7324 7430 SZL /GET ADDITION RESULT
7325 5341 JMP ERROR1 /COMPARE TO COMPLEMENT OF SIMULATION RESULT
7326 7340 CLA CLL CMA
7327 0774 AND TAC
7330 7040 CMA
7331 0773 AND TSIMAC

7332 7440 SZA /NOT 0, ERROR
7333 5341 JMP ERROR1
7334 7040 CMA
7335 0773 AND TSIMAC /GET SIMULATION RESULT
7336 7040 CMA
7337 0774 AND TAC /COMPARE TO COMPLEMENT OF REAL ADDITION
7340 7640 SZA CLA

7341 4752 JMS I ERRORS
7342 5713 JMP I COMAD
7343 0000
7344 0000 ARG1,
7345 0000 ARG2, 0
7346 0000 A10RA2, 0
7347 0000 SIMAC, 0
7350 4000 SIMLNK, 0
7351 0000 K4000, 4000
7352 7400 CARRY, 0
ERRORS, ERROR

```

7373 6763
7374 6764
7375 6765
7376 6762
7377 7164
7400 7400
7401 7604
7402 0267
7403 7640
7404 5600
7405 7240
7406 0777
7407 7402
7410 7240
7411 0776
7412 7402
7413 7240
7414 0775
7415 7402
7416 7240
7417 0774
7420 7402
7421 7240
7422 0773
7423 7402
7424 7240
7425 0772
7426 7402
7427 5600

PAGE ERROR,
0
LAS
AND CLA SR00
SZA I ERROR /IS SR00=1
JMP I ERROR /YES, DO NOT HALT
CLA CMA TDA1L /HALT WITH DATA1 IN AC
AND
HLT CMA TDA2L /HALT WITH DATA2 IN AC
CLA CMA TINSTL /HALT WITH INSTRUCTION IN AC
AND
HLT CMA TINADL /HALT WITH INSTRUCTION ADDRESS IN AC
AND
CLA CMA TDTAL /HALT WITH DATA ADDRESS IN AC
AND
HLT CMA TPADDL /HALT WITH INDIRECT IN AC
AND
JMP I ERROR
/RANDOM NUMBER GENERATOR
0
RANDL,
0
CLL RAL
SNL ENRAN
JMP MQL
CMA K3
AND RSIMAD
JMS JMP I RANDL
ENRAN, K3,
3

/RANDOM NUMBER GENERATOR
0
RANDL,
0
CLL RAL
SNL ENRAN
JMP MQL
CMA K3
AND RSIMAD
JMS JMP I RANDL
ENRAN, K3,
3

/END OF PASS
/
0 LAS
EPASL,
7442 0000
7443 7604

```


V141

PAL10

7444 0270 AND SR03
 7445 7640 SZA CLA
 7446 5642 JMP I EPASL
 7447 7040 CMA
 7450 0271 AND C215
 7451 4261 JMS TYPE
 7452 7040 CMA
 7453 0272 AND C212
 7454 4261 JMS TYPE
 7455 7040 CMA
 7456 0273 AND C324
 7457 4261 JMS TYPE
 7460 5642 JMP I EPASL

7461 0000 TYPE,
 7462 6046 /
 7463 6041 /
 7464 5263 /
 7465 7200 /
 7466 5661 /

7467 4000 SR00,
 7470 0400 SR03,
 7471 0215 C215,
 7472 0212 C212,
 7473 0324 C324,

/ TEST FOR PROPER DIFFERENCE

7474 0000 ABSL,
 7475 7041 CIA
 7476 7421 MQL
 7477 7040 CMA
 7500 0770 AND
 7501 4771 JMS TEMP3L
 7502 7500 SMA RSIMAD
 7503 7041 CIA
 7504 7001 IAC
 7505 7001 IAC
 7506 5674 JMP I ABSL

7507 0000 ABSL1,
 7510 7041 CIA
 7511 7421 MQL
 7512 7040 CMA
 7513 0767 AND R3L

7514	4771	JMS	RSIMAD
7515	7500	SMA	
7516	7041	CIA	
7517	7001	IAC	
7520	7001	IAC	
7521	5707	JMP I	ABSL1
		\$	
7567	7160		
7570	7161		
7571	7200		
7572	6757		
7573	6756		
7574	6754		
7575	6753		
7576	6761		
7577	6760		

0000	11111100	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
0100	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
0200	11100000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
0300	00000000	20010000	00000000	00000000	00000000	00000000	00000000	00000000	00000000

0400
0500
0600
0700

1000
1100
1200
1300

1400
1500
1600
1700

2000
2100
2200
2300

2400
2500
2600
2700

3000
3100
3200
3300

3400
3500
3600
3700

11-NOV-70

V141

PAL10

A10RA2	7345	PAGAL	7040
ABSL	7474	PAL	7055
ABSL1	7507	R1L	7150
ARG1	7343	R2L	7153
ARG2	7344	R3L	7160
C212	7472	RANDL	7430
C215	7471	RETURN	0000
C324	7473	RSIMAD	7200
CAF	6007	SETTL	6644
CARRY	7351	SIMAC	7346
CNTR1	6776	SIMAD	7204
COMAD	7313	SIMLNK	7347
DAADL	7064	SR00	7467
DATADL	0003	SR03	7470
DIRL	6702	SR09	6773
DOANDL	6716	SR10	6772
DOTSTL	6705	SR11	6771
ENCAR	7244	SRANDL	7162
ENCAR1	7253	SRIMAD	7163
ENDSIM	7274	START	0200
ENRAN	7440	STARTL	6600
EPASL	7442	T5400L	6766
ERROR	7400	T7760	6767
ERROR1	7341	T7770	6770
ERRORS	7352	TAC	6764
GANADL	7021	TCOMAD	6774
GANDL	7001	TDA1L	6760
GENL	7000	TDA2L	6761
IFLAGL	0005	TDAT1L	6624
INADDL	0002	TDAT2L	6634
INDIRL	7101	TDATA	6756
INSTL	0001	TEMP2L	6745
K0177L	7155	TEMP3L	7161
K10	6775	TEPASL	6750
K1000	7165	TERROR	6750
K1777	7152	TEST1L	6602
K200L	7157	TGENL	6746
K3	7441	TIFLGL	6755
K3L	7151	TINADL	6754
K4000	7350	TINSTL	6753
K400L	7156	TLINK	6765
LIMIT	7303	TPADDL	6757
LIML	7164	TRANDL	6752
MGA	7501	TRETTL	6747
MQL	7421	TRETUL	6735
NOTAUT	6676	TSIMAC	6763
NXTCAR	7234	TSIMAD	6751
POAL	7107	TSIML	6762
PWL	7154	TYPE	7461
PADDL	0004	XORALL	7260
PADL	7113		
PAGADL	7044		

ERRORS DETECTED: 0
LINKS GENERATED: 23
RUN-TIME: 5 SECONDS
2K CORE USED