

IDENTIFICATION

PRODUCT CODE: MAINDEC-8E-DIEC-D
PRODUCT NAME: MEMORY ADDRESS TEST
DATE CREATED: JUNE 11, 1974
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: BRUCE HANSEN

COPYRIGHT © 1977
INTELL EQUIPMENT CORPORATION

1, ABSTRACT

MEMORY ADDRESS TEST, A RELOCATABLE PROGRAM, CHECKS FOR PROPER
MEMORY ADDRESS SELECTION ON THE PDP-8E.

2, REQUIREMENTS

2.1 EQUIPMENT

PDP-8E EQUIPPED WITH A TELETYPE

2.2 STORAGE

MEMORY ADDRESS TEST OCCUPIES LOCATIONS 7200-7507,
AFTER RELOCATING, THE TEST OCCUPIES LOCATIONS 0000-0307.

2.3 PRELIMINARY PROGRAMS

NONE

3, LOADING PROCEDURE

USE STANDARD BINARY LOADER

4, STARTING PROCEDURE

4.1 INITIAL SWITCH SETTINGS

ALL SR'S = 0 RUN ADDRESS TEST HIGH AND RELOCATE PROGRAM AFTER
1 PASS TO ADDRESS TEST LOW AND THEN RELOCATE PROGRAM TO ADDRESS
TEST HIGH, REPEATEDLY.

SR0(0) HALT AFTER ERROR PRINTOUT
SR1(1) AND SR2(0) RUN ADDRESS TEST HIGH ONLY
SR1(1) AND SR2(1) RELOCATE PROGRAM AND RUN ADDRESS TEST LOW ONLY
SR1(0) PROGRAM WILL RELOCATE AFTER A PASS
SR1(1) PROGRAM WILL STAY IN TEST AND WILL NOT RELOCATE

4.2 SWITCH SETTINGS AFTER PROGRAM IS RUNNING

SR0(0) HALT AFTER ERROR PRINTOUT
SR1(0) RUN TEST AND RELOCATE
SR1(1) RUN SAME TEST, DO NOT RELOCATE

4.3 STARTING ADDRESSES

0200 INITIALLY

RESTART ADDRESS: 0000,7200

4.4 OPERATOR ACTION

A. SET SR TO 0200 AND PRESS LOAD ADDRESS

B. SET SR FOR DESIRED OPERATION (SEE 4.1) PRESS CLEAR, THEN CONTINUE. FOR MOST CASES THE SWITCH REGISTER SHOULD EQUAL ZERO.

5. OPERATING PROCEDURE

ONCE THE PROGRAM IS RUNNING, THE STARTING ROUTINE IS GIVEN UP FOR A TEST AREA. SR0 AND SR1 ARE THE ONLY SWITCHES THAT HAVE ANY AFFECT ON THE PROGRAM. (SEE 4.2) IN ORDER TO RESTART THE PROGRAM, CERTAIN LOCATIONS MUST BE EXAMINED (SEE BELOW) TO DETERMINE WHERE THE PROGRAM IS, SINCE THE PROGRAM RELOCATES ITSELF FROM ADDRESS TEST HIGH TO ADDRESS TEST LOW AND ADDRESS TEST LOW TO ADDRESS TEST HIGH. IF ADDRESS 0000 CONTAINS A 7300 AND ADDRESS 307 CONTAINS A 7200, START THE PROGRAM AT LOCATION 0000 FOR ADDRESS TEST LOW. IF 7200 AND 7507 HAS 7300 AND 7200 RESPECTIVELY, LOAD ADDRESS 7200 AND SET DESIRED SWITCHES AND HIT CLEAR AND THEN CONTINUE.

6. ERRORS

6.1 ERROR PRINTOUTS

A XXXX 0 YYY (ERROR PRINTOUT FORMAT)

A XXXX (ADDRESS) XXXX = ADDRESS CONTAINING WRONG DATA.

C YYY (CONTENTS) YYY = CONTENTS OF LOCATION XXXX

THE CONTENTS OF AN ADDRESS SHOULD EQUAL THE ADDRESS OR THE COMPLEMENT OF THE ADDRESS

6.2 ERROR RECOVERY

ANALYSIS OF SEVERAL ERROR PRINTOUTS SHOULD ESTABLISH A MEANINGFUL PATTERN THAT WILL SINGLE OUT A PARTICULAR ADDRESS SELECTION.

IF IT IS NECESSARY TO SCOPE THE PROBLEM, THE FOLLOWING TWO INSTRUCTIONS MAY BE ENTERED IN MEMORY:

TAD (BAD LOCATION)

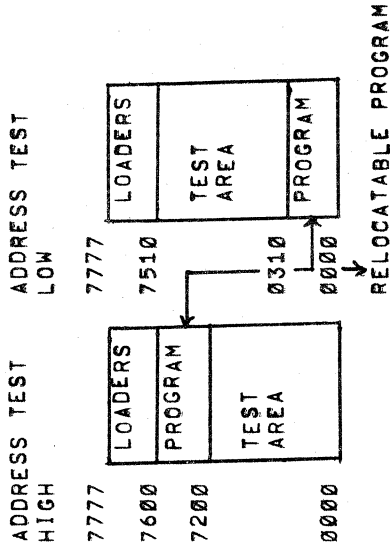
JMP .-1

7. MISCELLANEOUS

7.1 EXECUTION TIME

AFTER EVERY 96 COMPLETE PROGRAM LOOPS AN EC IS PRINTED OUT BEFORE THE PROGRAM RELOCATES, EC IS TYPED OUT TWICE, ONCE AFTER ADDRESS TEST HIGH AND THE SECOND TIME AFTER ADDRESS TEST LOW.

7.2 MEMORY MAPS



8. PROGRAM DESCRIPTION

THE PROGRAM CONSIST OF TWO PHASES WHICH OCCUR IN THE FOLLOWING SEQUENCE:

PHASE 1 LOAD MEMORY SEQUENTIALLY IN THE FORWARD DIRECTION WITH EACH ADDRESS EQUAL TO ITS CONTENTS, THEN READ AND CHECK MEMORY FOR ERRORS.

PHASE 2 LOAD MEMORY SEQUENTIALLY IN THE REVERSE DIRECTION WITH ONE'S COMPLEMENT OF EACH ADDRESS, THEN READ AND CHECK MEMORY FOR ERRORS.

IN PHASE ONE, THE CONTENTS OF EVERY LOCATION IN THE TEST AREA IS EQUAL TO ITS ADDRESS. IF AN ERROR OCCURS, THE CONTENTS WERE PROBABLY DEPOSITED INTO A WRONG ADDRESS OR MULTIPLE ADDRESSES. IN PHASE 2 THE MEMORY IS LOADED WITH THE ONE'S COMPLEMENT OF THE ADDRESS, IF THE ADDRESS OR ITS COMPLEMENT IS WRONG, A ERROR MESSAGE WILL BE TYPED OUT GIVING THE FAILING ADDRESS AND ITS CONTENTS.

BETWEEN PHASE 1 AND PHASE 2 EACH ADDRESS IS CHECKED WITH THE ADDRESS EQUAL TO ITS ADDRESS WITH ALL OTHER BITS A ZERO, AND THEN WITH THE ADDRESS BITS EQUAL TO A ZERO AND ALL OTHER BITS SET TO A ONE. THIS CHECKS EACH ADDRESS FOR BIT DROPOUT OR PICKUP OF ALL BITS OF AN ADDRESS.

SAMPLE ERROR PRINTOUT:

A2560 C2760

EXPLANATION - WHILE ATTEMPTING TO WRITE A 2760 INTO LOCATION 2760, THE DATA WAS WRITTEN INTO LOCATION 2560. BIT FOUR WAS DROPPED.

SAMPLE ERROR PRINTOUT:

A2560 C5207

EXPLANATION - WHILE ATTEMPTING TO WRITE THE COMPLEMENT OF 2560 (5217) INTO LOCATION 2560, 5207 WAS WRITTEN INTO THE LOCATION INSTEAD. BIT 8 WAS DROPPED.

AFTER 96 PROGRAM LOOPS OF PHASES 1-4 THE PROGRAM RELOCATES AND RUNS ANOTHER 96 PROGRAM LOOPS BEFORE IT RELOCATES AGAIN.

ADDRESS TEST HIGH - TEST MEMORY LOCATIONS 0000-7177.

ADDRESS TEST LOW - TEST MEMORY LOCATIONS 310-7510.

```

0000 /PDP-8E MEMORY ADDRESS TEST
0000 *0000
0001 JMP 1
0002 2
0003 3
0004 0
0005 0
0200 *200
0201 7604 BEG,
0202 7440
0203 5204
0204 5615 /SR=0 RUN PROGRAM AND RELOCATE
0205 1217
0206 7640
0207 5210
0210 5615 /RUN MEMORY ADDRESS TEST HIGH
0211 7604
0212 1220
0213 7640
0214 5615 /WRONG SWITCH SETTING RUN HIGH AND RELOCATE
0215 7200 /RELOCATES PROGRAM AND RUNS MEMORY ADDRESS TEST HIGH
0216 7405
0217 6000
0220 5000
7200 *7200

```

/LOAD MEMORY FORWARD DIRECTION

```

7200 LOADUP, CLA CLL
7201 TAD LIMLO
7202 DCA ADRES
7203 TAD M7200
7204 DCA CTR
7205 TAD ADRES
7206 DCA I ADRES
7207 ISZ ADRES
7210 ISZ CTR
7211 JMP LOADUP+5
7212 TAD LIMLO
7213 DCA ADRES
7214 TAD M7200
7215 DCA CTR

```

/SET TEST AREA STARTING ADDRESS

/DEPOSIT ADDRESS IN CONTENTS

/GET CONTENTS FORWARD DIRECTION

```

7216 MEMLUP, TAD I ADRES
7217 CIA
7220 TAD ADRES
7221 SZA CLA
7222 JMS ERROR
7223 ISZ ADRES
7224 ISZ CTR

```

/GET ADDRESS
/SKIP IF EQUAL
/CONTENTS NOT SAME AS ADDRESS
/SELECT NEXT ADDRESS
/SKIP IF END TEST AREA

```

7225 5216 JMP MEMLUP
7226 1276 /LOAD MEMORY REVERSE DIRECTION
7227 3275 LOADWN, TAD LIMHI
7230 1300 DCA ADRES /SET TEST AREA ENDING ADDRESS
7231 3305 TAD M7200
7232 4275 DCA CTR
7233 7040 TAD ADRES
7234 3675 CMA
7235 7240 DCA I ADRES /DEPOSIT 1'S COMPLEMENT OF ADDRESS IN ADDRESS
7236 1275 CLA CMA /AC=-1
7237 3275 TAD ADRES /AC=(ADRES)-1
7240 2305 DCA ADRES /DECREMENT ADDRESS
7241 5232 ISE CTR /SKIP WHEN LOWER LIMIT REACHED
7242 1300 JMP LOADWN+4
7243 3305 TAD M7200
DCA CTR
    
```

```

LOOP2, /SEQUENTIAL LOCATION TEST (DOWN)
7244 1276 TAD LIMHI
7245 3275 DCA ADRES /SET STARTING ADDRESS
7246 1675 TAD I ADRES /GET CONTENTS
7247 7001 IAC
7250 1275 TAD ADRES /GET ADDRESS
7251 7640 SEA CLA /SKIP IF EQUAL
7252 4320 JMS ERROR /CONTENTS NOT COMPLEMENT OF ADDRESS
7253 7240 CLA CMA /AC=-1
7254 1275 TAD ADRES
7255 3275 DCA ADRES /AC=(ADRES)-1
7256 2305 ISE CTR /SELECT NEXT ADDRESS
7257 5246 JMP LOOP2+2 /SKIP IF END TEST AREA
7260 2301 ISE COUNT
7261 5200 JMP LOADUP
7262 1302 TAD RESTOR
7263 3301 DCA COUNT
7264 4312 TAD CR
7265 4343 JMS PRINT
7266 4313 TAD LF
7267 4343 JMS PRINT
7270 1303 TAD K305
7271 4343 JMS PRINT
7272 1316 TAD C
7273 4343 JMS PRINT
7274 5377 JMP BANK1
    
```

```

/CONSTANTS AND VARIABLES
ADRES, 0
LIMHI, 7177
LIMLO, 0
M7200, -7200
COUNT, -140
RESTOR, -140
K305, 305
    
```



```

7304 7774 M4,
7305 0000 CTR,
7306 0007 MSK7,
7307 0260 TW6,
7310 0000 STOR,
7311 7004 NUM,
7312 0215 CR,
7313 0212 LF,
7314 0240 SPACE,
7315 0301 A,
7316 0303 C,
7317 0000 CNT,

```

/ERROR MESSAGE

```

0 TAD CR
JMS PRINT
TAD LF
JMS PRINT
TAD A
JMS PRINT
TAD ADRES
JMS TYPAC
TAD SPACE
JMS PRINT
TAD C
JMS PRINT
TAD I ADRES
JMS TYPAC
LAS CLA
SMA CLA
HLT
JMP I ERROR

```

/HALT ON ERROR (SR0)

ERROR,

```

7320 0000
7321 1312 TAD CR
7322 4343 JMS PRINT
7323 1313 TAD LF
7324 4343 JMS PRINT
7325 1315 TAD A
7326 4343 JMS PRINT
7327 1275 TAD ADRES
7330 4351 JMS TYPAC
7331 1314 TAD SPACE
7332 4343 JMS PRINT
7333 1316 TAD C
7334 4343 JMS PRINT
7335 1675 TAD I ADRES
7336 4351 JMS TYPAC
7337 7604 LAS CLA
7340 7700 SMA CLA
7341 7402 HLT
7342 5720 JMP I ERROR

```

PRINT,

```

0 TLS
TSF
JMP '-1
CLA
JMP I PRINT

```

/TYPE (AC) IN OCTAL

```

0 DCA STOR
TAD BACK+1
DCA BACK+2
TAD M4
DCA CNT
CLL
TAD STOR
RTL
RTL

```

```

7351 0000 TYPAC,
7352 3310
7353 1361 DCA STOR
7354 3362 TAD BACK+1
7355 1304 DCA BACK+2
7356 3317 TAD M4
7357 7100 DCA CNT
7360 1310 CLL
7361 7006 TAD STOR
7362 7006 RTL

```

7363 DCA STOR
 7364 TAD STOR
 7365 AND MSK7
 7366 TAD TW6
 7367 JMS PRINT
 7370 TAD NUM
 7371 DCA BACK+2
 7372 ISE CNT
 7373 JMP BACK
 7374 JMP I TYPAC
 *7377 NOP
 BANK1, 7000

/LOOK AT SR TO SEE IF PROGRAM RELOCATES

7400 LAS
 7401 AND COMP
 7402 SNA CLA
 7403 JMP MOVEH
 7404 JMP LOADP
 7405 TAD STORE
 7406 CMA
 7407 DCA STORE
 7410 TAD STORE
 7411 SMA CLA
 7412 JMP MOVEH
 7413 JMP MOVEH
 7414 CLA CLL
 7415 TAD LIMLOL
 7416 DCA I X1
 7417 TAD LIMHIL
 7420 DCA I X2
 7421 CLA CLL
 7422 DCA CONT1
 7423 TAD CNT2
 7424 DCA CONT2
 7425 TAD HGH
 7426 DCA HIGH
 7427 TAD I HIGH
 7430 DCA I CONT1
 7431 ISE CONT1
 7432 ISE HIGH
 7433 ISE CONT2
 7434 JMP MOVITL
 7435 JMP 0
 7436 TAD LIMLOH
 7437 DCA I X3
 7440 TAD LIMHIL
 7441 DCA I X4
 7442 CLA CLL
 7443 DCA LOW
 7444 TAD CNT2
 7445 DCA CONT2
 7446 TAD HGH
 7447 DCA HIGH
 7450 TAD I LOW
 7451 DCA I HIGH

/JMP TO MOVE ROUTINE
/KEEP PROGRAM IN SAME AREA

/RELOCATES PROGRAM TO HIGH MEMORY
/RELOCATES PROGRAM TO LOW MEMORY

/LOW ADDRESS UNDER TEST=310

/HIGH ADDRESS UNDER TEST=7510
/SETS UP COUNTERS FOR MOVING

/MOVES PROGRAM TO LOWER MEMORY

/IS PROGRAM RELOCATED
/NO
/YES START PROGRAM

/LOW ADDRESS UNDER TEST=0000

/HIGH ADDRESS UNDER TEST=7177
/RESETS COUNTERS

/MOVE PROGRAM TO UPPER MEMORY

7452	2272	ISZ LOW
7453	2267	ISZ HIGH
7454	2266	ISZ CONT2
7455	5250	JMP MOVITH
7456	5663	JMP I HGH
7457	2000	COMP, 2000
7460	0310	LIMLOL, 0310
7461	7510	LIMHIL, 7510
7462	7470	CONT2, 7470
7463	7200	HGH, 7200
7464	0000	STORE, 0
7465	0000	CONT1, 0
7466	7470	CONT2, 7470
7467	7200	HIGH, 7200
7470	0000	LIMLOH, 0
7471	7177	LIMHIL, 7177
7472	0000	LOW, 0
7473	7277	X1, LIMLO
7474	7276	X2, LIMHI
7475	0077	X3, 0077
7476	0076	X4, 0076
7477	7000	LOADP, NOP
7500	4301	JMS, +1
7501	0000	0
7502	1301	TAD, -1
7503	0307	AND STAY
7504	7700	SMA CLA
7505	5000	JMP 0
7506	5707	JMP I STAY
7507	7200	STAY, 7200

/IS PROGRAM RELOCATED
/NO
/YES START PROGRAM

\$

