

11/2

MAINDEC CHANGE
NOTICE

CHANGE NO.
11-DZMME-A - 1 (DLE-)
Sheet 1 of 1

AUTHOR Jim Lacey	PROGRAM DATE MAY 72	PRODUCT LINE PDP-11	MAINDEC NUMBER 11-DZMME-A
DATE 7/7/72	EXT. 3142		

PROGRAM NAME Memory Moving Ones & Zeros Test DEVICE Core Memory.

ITEM Ø.	For information only: Latest revision has been released to program library and will be phased-in when all copies of 11-DLEB are gone. This program supersedes MAINDEC-11-DLEB.						
1. 7/10/72	Problem: The moving Ø disturbance test (last half of part 2) is not performed. Correction: Change the instruction at Location 1Ø56 from "JMP ITERCT" to "JMP ZIPPER" which will be accomplished by the following patch. <table border="1"> <thead> <tr> <th>LOCATION</th> <th>FROM</th> <th>TO</th> </tr> </thead> <tbody> <tr> <td>1060</td> <td>177554</td> <td>177664</td> </tr> </tbody> </table> NOTE: This patch will double the run time for part 2.	LOCATION	FROM	TO	1060	177554	177664
LOCATION	FROM	TO					
1060	177554	177664					

M. C. N. REQUIRED
THIS PROGRAM REQUIRES MON(S)
IN ORDER TO WORK PROPERLY

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DEMME-A-D
(SUPERSEDES MAINDEC-11-D1EB)

PRODUCT NAME: MEMORY MOVING ONES AND ZEROS TEST

DATE REVISED: MAY 19, 1972

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: JOHN RODENHISER/ JIM LACEY

COPYRIGHT © 1970, 1972
DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT

THE MOVING ONES AND ZEROS TEST VERIFIES THE FOLLOWING:

- A) THE SELECTED TEST AREA IS CAPABLE OF WRITING AND READING ALL CONFIGURATIONS OF A 0 BIT MOVED SEQUENTIALLY THROUGH ALL BIT POSITIONS OF ALL TEST ZONE LOCATIONS.
- B) THE SELECTED TEST AREA IS CAPABLE OF WRITING AND READING ALL CONFIGURATIONS OF A 1 BIT MOVED SEQUENTIALLY THROUGH ALL BIT POSITIONS OF A TEST ZONE LOCATIONS.
- C) THE SELECTED TEST AREA IS FREE FROM READ REGENERATE NOISE DISTURBANCES FOR ALL OF THE FOREMENTIONED MOVING 0 AND MOVING 1 CONDITIONS. SCOPE LOOP OPTIONS ARE PROVIDED TO FACILITATE AND ADDITIONAL DIAGNOSTIC PROCEDURES THAT MAY BE USED IN CONJUNCTION WITH THIS TEST.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-11 WITH MINIMUM OF 4K MEMORY

2.2 STORAGE

2.2.1 PROGRAM STORAGE - THE ROUTINE USES MEMORY FROM 200 TO 3244

3. LOADING PROCEDURE

3.1 METHOD

PROCEDURE FOR NORMAL BINARY TAPES SHOULD BE FOLLOWED:

- 1. ABSOLUTE LOADER MUST BE IN MEMORY.
- 2. PLACE BINARY TAPE IN READER.
- 3. LOAD ADDRESS 07500 (* DETERMINED BY ADDRESS OF LOADER)
- 4. PRESS "START" (PROGRAM WILL LOAD)

4. STARTING PROCEDURE

4.1 CONTROL SWITCH SETTING

STARTING AT SA 200 ALL SWITCHES SHOULD BE DOWN OR ZERO.

4.2 STARTING ADDRESS OR ADDRESSES

200 • START FOR AUTOMATIC TEST LIMITS
202 • START FOR SELECTED TESTS LIMITS

4.3 PROGRAM AND/OR OPERATOR ACTION

LOAD PROGRAM INTO MEMORY,
SET SWITCH REGISTER TO STARTING ADDRESS,
LOAD ADDRESS 200,
PRESS START,
THE PROGRAM WILL RUN THROUGH THE SELECTED ADDRESS FIELD
AND LOOP IN PART 1.

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

5.1.1 WITH ALL SWITCHES DOWN, THE PROGRAM
WILL PRINT OUT ON ERRORS AND CONTINUE
IN TEST.

5.1.2 SWITCH SETTINGS ARE

SW15 = 1 OR UP ... HALT ON ERROR
SW14 = 1 OR UP ... SCOPE LOOP
SW13 = 1 OR UP ... INHIBIT PRINTOUT
SW12 = 1 OR UP ... HALT ON END OF PROGRAM
SW11 = 1 OR UP ... PART 2 SWITCH

5.1.3 AUTOMATIC TEST LIMITS

IF THE PROGRAM IS STARTED AT ADDRESS 200 THE PROGRAM WILL
TEST ALL AVAILABLE MEMORY, CARE SHOULD BE TAKEN TO SELECT THE
PROPER OPERATIONAL SWITCH SETTINGS (REFER TO 5.1.2) BEFORE
STARTING.

5.1.4 SELECTED TEST LIMITS

IF THE PROGRAM IS STARTED AT ADDRESS 202 A MESSAGE WILL
BE PRINTED ON THE TELETYPE INSTRUCTING THE OPERATOR ON
THE METHOD OF SELECTING OTHER MEMORY TEST AREAS AND THEN
THE PROGRAM WILL STOP AT THE FIRST OF THREE HALTS.

- A, SET THE LOW TEST LIMIT IN THE SWITCH REGISTER AND
PRESS CONTINUE.
- B, THEN SET THE HIGH TEST LIMIT IN THE SWITCH REGISTER
AND PRESS CONTINUE.
- C, THEN SET THE OPERATIONAL SWITCH SETTINGS (REF 5.1.2)
AND PRESS CONTINUE.

THE PROGRAM WILL NOW BEGIN TESTING THE SELECTED AREA,
THE PROGRAM WILL NOT ALLOW LIMITS TO BE SELECTED THAT
WOULD CAUSE ITSELF TO BE DESTROYED. AN EXCEPTION TO THIS
WOULD BE THE INSTRUCTIONAL TEXT STORAGE AREA, STARTING
THE PROGRAM AT 200 OR SELECTING LIMITS WHICH OVERLAP
THIS AREA WILL CAUSE THE TEXT MESSAGE TO BE DESTROYED.

5.2 SUBROUTINE ABSTRACTS

5.2.1 ERROR1

SUBROUTINE ERROR1 IS CALLED WHENEVER THE MAIN TEST PROGRAM DETECTS A MOVING 0 WRITE-READ ERROR. IF SWITCH 13 IS NOT PRESENT THE SUBROUTINE PRINTS THE ERROR ADDRESS AND ITS CONTENTS. UPON THE END OF PRINTING, OR IN THE EVENT SW13 IS PRESENT, CHECKS THE HALT-SCOPE SWITCHES (SW15 AND 14 RESPECTIVELY). IF NEITHER SWITCH IS SET THE PROGRAM IMMEDIATELY RETURNS TO THE MAIN TEST PROGRAM. IF SW15 IS SET, THE PROGRAM HALTS. AT THIS TIME THE SCOPE LOOP SWITCH (SW14) MAY BE SET (IF NOT SET ALREADY). A CONTINUE FROM THIS HALT WILL CAUSE THE SUBROUTINE TO ENTER THE SCOPE LOOP CHECK. IF SW14 IS SET THE SUBROUTINE WILL PERFORM A WRITE-READ LOOP THROUGH THE ERROR LOCATION. ERROR PRINTOUTS MAY BE ENABLED OR INHIBITED DURING THE SCOPE LOOP AS COMMANDED BY THE POSITION OF SW13. THE SCOPE LOOP IS TERMINATED AT ANY TIME DURING THE LOOP BY RESETTING SW14. IF SW14 WAS RESET PRIOR TO OR AFTER THE ERROR HALT, OR IS RESET DURING THE SCOPE LOOP, THE PROGRAM WILL RETURN TO MAIN TEST PROGRAM AND RESUME TESTING.

5.2.2 ERROR2

SUBROUTINE ERROR2 IS CALLED WHENEVER THE MAIN TEST PROGRAM DETECTS A MOVING 1 WRITE-READ ERROR. EXCEPTION FOR PROVIDING A DIFFERENT STARTING ADDRESS FOR THE ERROR MESSAGE, THIS SUBROUTINE FUNCTIONS EXACTLY AS DESCRIBED UNDER ERROR1.

5.2.3 ERROR3

SUBROUTINE ERROR3 IS CALLED WHENEVER THE MAIN TEST PROGRAM DETECTS A DISTURBANCE ERROR RESULTING FROM AN OPERATION WHICH WRITES AND READS A 0 (10 TIMES) INTO THE LOCATION UNDER TEST. THIS DISTURBANCE IS DETECTED DURING A CHECK SCAN OF THE REST OF THE MEMORY TEST FIELD. EXCEPT FOR PROVIDING A DIFFERENT STARTING ADDRESS FOR THE ERROR MESSAGE, THIS SUBROUTINE FUNCTIONS EXACTLY AS DESCRIBED UNDER ERROR1.

5.2.4 ERROR4

SUBROUTINE ERROR4 IS CALLED WHENEVER THE MAIN TEST PROGRAM DETECTS A DISTURBANCE ERROR RESULTING FROM AN OPERATION WHICH WRITES AND READS A -1 DATA CONFIGURATION (10 TIMES) INTO THE LOCATION UNDER TEST. THIS DISTURBANCE IS DETECTED DURING A CHECK SCAN OF THE REST OF THE MEMORY TEST FIELD. EXCEPT FOR PROVIDING A DIFFERENT STARTING ADDRESS FOR THE ERROR MESSAGE, THIS SUBROUTINE FUNCTIONS EXACTLY AS DESCRIBED UNDER ERROR1.

6, ERRORS

6,1 ERROR PRINTOUT

PRINTS ALL ERRORS UNLESS INHIBITED BY SW13.

6,2 ERROR RECOVERY

A) IF IN A SCOPE LOOP, RESET SW14.

B) TO RECOVER FROM AN ERROR HALT, MAKE SURE SW14 IS RESET THEN DEPRESS CONTINUE.

C) RELOAD SA AND START.

7, RESTRICTIONS

7,1 STARTING RESTRICTION

NONE

7,2 OPERATIONAL RESTRICTION

NONE

8, MISCELLANEOUS

8,1 EXECUTION TIME

THE PROGRAM WILL RING THE TELETYPE BELL AFTER EVERY 21 PASSES THRU THE PROGRAM WHICH IS APPROXIMATELY ONCE PER MINUTE FOR PART 1 AND ONCE EVERY FOUR MINUTES FOR PART 2 WITH 4K OF MEMORY AVAILABLE.

9.

PROGRAM DESCRIPTION

THIS TEST IS COMPOSED OF TWO SECTIONS, EACH OF WHICH CONTAINS TWO BASIC TESTS. THE FIRST OF THE TWO BASIC TESTS PERFORMED MOVES A ZERO THROUGH EACH BIT POSITION OF THE LOCATION UNDER TEST. EACH LOCATION IN THE TEST ZONE IS TESTED IN THIS MANNER. THE SECOND BASIC TEST MOVES A 1 THROUGH EACH BIT POSITION OF THE LOCATION UNDER TEST. THESE TWO BASIC WRITE READ TESTS COMPRISE PART1. THE TWO BASIC TESTS OF PART2 ARE SIMILAR EXCEPT THAT THE TEST OBJECT IS TO VERIFY REPEATED READ REGENERATE CYCLES DO NOT NOISE DISTURB ANY OTHER LOCATION IN THE TEST ZONE. HENCE IMMEDIATELY AFTER A ZERO (OR A 1) IS MOVED INTO A NEW BIT POSITION OF THE LOCATION UNDER TEST A X10 ITERATION READ LOOP IS EXECUTED, FOLLOWED BY A CHECK OF THE REMAINING TEST ZONE LOCATIONS TO VERIFY NO DISTURBANCE HAS OCCURRED. THE MOVING 0 DISTURBANCE TEST IS ALSO PERFORMED FIRST IN PART2, FOLLOWED BY THE MOVING 1 DISTURBANCE TEST.

AFTER THE PROGRAM HAS BEEN STARTED AND IS RUNNING (SUCCESSFULLY) IN PART 1, THE OPERATOR SHOULD SET THE PART 2 SWITCH (SW11) TO PERFORM TO THE DISTURBANCE TESTS. PROGRAM CONTROL IS SUCH THAT THE ORDER OF TESTING STARTS WITH TESTS TO DEMONSTRATE THE ABILITY OF MEMORY TO WRITE AND HEAD THE MOVING BIT CONFIGURATIONS BEFORE PERFORMING THE MORE DIAGNOSTICALLY COMPLEX DISTURBANCE TESTS. THE PROGRAM WILL RING THE TELETYPE BELL ONCE EVERY MINUTE TO INDICATE THAT IT IS RUNNING.

10.

LISTING

.NLIST SEQ

.TITLE MEMORY MOVING ONES AND ZEROES TEST MAINDEC-11-DZMME-A
;COPYRIGHT 1970,1972 DIGITAL EQUIPMENT CORP., MAYNARD, MASS.
NOP=240

000240

.ENABL ABS
.=0

000000

!TRAP CATCHER 0-176

000004
000084 000362
000200 000200
000200 000460
000202 000406
000204 000000
000206 000000
000210 177564
000212 177566
000214 177570
000216 000000

.#4
TLG
.#200
BR START1
BR START
LOLMTI 0
HILMTI 0
TCSR1 177564
TDBR1 177566
SWREG1 177570
SWITCH1 0

000220	005067	177772		STARTI	CLR	SWITCH	
000224	012767	177752	001206		MOV	#=26,BELLCT	
000232	012767	002560	177744		MOV	#ENDBUF,LOLMT	
000240	012767	017470	177740		MOV	#17470,HILMT	
000246	012706	002556			MOV	#BUFFER,X6	
000252	012702	002562			MOV	#MSG1,X2	
000256	004767	001456			JSR	X7, TOP	
000262	000000				HALT		
000264	005777	177724			TST	@SWREG ;LOOK FOR LOLMT	
000270	001407				BEQ	HISET ;DEFERRED	
000272	027767	177716	177704		CMP	@SWREG,LOLMT ;CKN LOLMT INPUT>TERMNL	
000300	103403				BLO	HISET ;LOLMT<TERMNL USE LMT DEFINED	
000302	017767	177706	177674		MOV	@SWREG,LOLMT ;LOLMT>TERMNL STORE INPUT	
000310	000000			HISETI	HALT	;WAIT FOR CONTINUE	
000312	005777	177676			TST	@SWREG ;LOOK FOR HILMT	
000316	001407				BEQ	CONSET ;DEFERRED	
000320	027767	177670	177656		CMP	@SWREG,LOLMT	
000326	103403				BLO	CONSET	
000330	017767	177660	177650		MOV	@SWREG,HILMT ;STORE INPUT	
000336	000000			CONSETI	HALT	;SET UP CONTROL SWITCHES	
000340	000441				BR	ALLONE	
000342	005067	177650		START1I	CLR	SWITCH	
000346	012706	002556			MOV	#BUFFER,X6	
000352	010603				MOV	X6,X3	;TEST POINT IN LOWER BANK
000354	005723			SEEI	TST	(X3)+	;TEST
000356	000240				NOP		;PRECAUTIONARY DELAY
000360	000775				BR		;NO TRAP, CONTINUE
000362	162703	000004		TLGI	SUB	#4,X3	;TRAPPED
000366	005737	000042			TST	##42	
000372	001407				BEQ	S1	
000374	023727	000042	000702		CMP	##42,#ENDA0R	
000402	001405				BEQ	S2	
000404	162703	002734			SUB	#1500,,X3	
000410	000402				BR	S2	
000412	162703	000300		S1I	SUB	#300,X3	
000416	010367	177564		S2I	MOV	X3,HILMT	
000422	012767	002560	177554		MOV	#ENDBUF,LOLMT	
000430	012767	177752	001002		MOV	#=26,BELLCT	
000436	000402				BR	ALLONE	
000440	005000			ZIROUTI	CLR	X0	
000442	000402				BR	,+6	
000444	012700	177777		ALLONEI	MOV	#=1,X0 ;SET UP ONES IN R0	
000450	005067	000760			CLR	FAIL0	
000454	005067	000756			CLR	FAIL1	
000460	016701	177520			MOV	LOLMT,X1 ;GET LOLMT	
000464	010021				MOV	X0,(1)+ ;LOAD MEMORY WITH ONES	
000466	020167	177514			CMP	X1,HILMT ;CKN MEMORY LOADED	
000472	103774				BLO	,=6	
000474	005700				TST	X0 ;SWITCH FOR MOVING ONE TEST	
000476	001431				BEQ	CIPHER ;GOTO MOVING ONE TEST	
000500	012700	177776			MOV	#=2,X0 ;SET UP ONE ZERO IN R0	
000504	016701	177474			MOV	LOLMT,X1 ;GET LOLMT	
000510	010011			WRTONEI	MOV	X0,X1 ;WRITE ONE ZERO	

000512 020011
000514 001405
000516 012767 177777 000710
000524 004767 000440
000530 000261
000532 006100
000534 022700 177777
000540 001401
000542 000762

OKAY01

CMP X0,0X1 IREAD ONE ZERO
BEQ OKAY0
MOV #-1,FAIL0 ISET FAIL0 NEGATIVE
JSR X7,ERROR1 ITO ERROR1
SEC
RDL X0 IADVANCE ZERO TO NXMSD
CMP #-1,X0 ICKN FOR ALL ONES
BEQ NEXONE IIF SO MOVE ON TO NX LOC
BR WRTONE ICONTINUE WRT=RD-ROT SEQUENCE

000544	012700	177776		NEXONE:	MOV	#-2,X0	
000550	021121				CMP	@X1,(1)+	;JUST INCREMENTING
000552	020167	177430			CMP	X1,HILMT	;CKN HILMT
000556	103754				BLO	WRTONE	
000560	000727				BR	ZIPOUT	
000562	012700	000001		CIPHER:	MOV	#1,X0	;PUT ONE IN R0
000566	016701	177412			MOV	L0LMT,X1	;L0LMT INTO R1
000572	010011			WRTZIP:	MOV	X0,@X1	;WRITING ONE ON FIELD OF ZERO
000574	020011				CMP	X0,@X1	;READ
000576	001405				BEQ	OKAY1	
000600	012767	177777	000630		MOV	#-1,FAIL1	;SET FAIL1 NEGATIVE
000606	004767	000364			JSR	X7,ERROR2	;TO ERROR2
000612	000241			OKAY1:	CLC		
000614	006100				ROL	X0	;ADVANCE 1 TO NX MSD
000616	001401				BEQ	NEXZIP	;MOV ON NX LOC
000620	000764				BR	WRTZIP	;CONTINUE WRT-RD-ROT SEQUENCE
000622	012700	000001		NEXZIP:	MOV	#1,X0	
000626	021121				CMP	@X1,(1)+	;JUST INCREMENTING
000630	020167	177352			CMP	X1,HILMT	;CKN HILMT AND INCR
000634	103756				BLO	WRTZIP	
000636	005267	000576		ITERCT:	INC	BELLCY	;INCR PASS ITERATION COUNTER
000642	001023				CONTCK		
000644	012767	177752	000566		MOV	#-26,BELLCY	
000652	105777	177332			TSTB	@TCSR	
000656	100375				BPL	,=4	
000660	012777	000207	177324		MOV	#207,@TDBR	
000666	013702	000042			MOV	@#42,X2	
000672	001407				BEQ	CONTCK	
000674	005167	177316			COM	SWITCH	
000700	001024				BNE	DSTURB	
000702	004712			ENDADRI:	JSR	X7,(2)	
000704	000240				NOP		
000706	000240				NOP		
000710	000240				NOP		
000712	032777	010000	177274	CONTCK:	BIT	#10000,@SWREG	
000720	001401				BEQ	SW11CK	
000722	000000				HALT		
000724	032777	004000	177262	SW11CK:	BIT	#4000,@SWREG	
000732	001007				BNE	DSTURB	;SW11 UP GOTO PART 2
000734	005767	177256			TST	SWITCH	
000740	001004				BNE	DSTURB	
000742	000167	177476			JMP	ALLONE	;NO SW11 REPEAT PART 1
000746	005000			ZIPPER:	CLR	X0	;GET ZERO
000750	000402				BR	,+6	;GO PROPAGATE
000752	012700	177777		DSTURB:	MOV	#-1,X0	;SETUP 1/S IN R0
000756	016701	177222			MOV	L0LMT,X1	;GET L0LMT
000762	010021				MOV	X0,(1)+	;LOAD MEMORY WITH ONES
000764	020167	177216			CMP	X1,HILMT	;CKN MEMORY LOADED
000770	103774				BLO	,=6	
000772	005700				TST	X0	;SWITCH FOR MOVING ONE TEST
000774	001440				BEQ	ZIPCK	
000776	012700	177776			MOV	#-2,X0	;SET ONE ZERO IN R0

001002 016701 177176

MOV

L0LMT,X1

JGET L0LMT

001006	012767	000012	000426	SETLPI	MOV	#12,NOISY);SET UP WRITE-READ LOOP
001014	010011			DISTLPI	MOV	X0,@X1);WRITE
001016	020011				CMP	X0,@X1);READ
001020	005367	000416			DEC	NOISY	
001024	001373				BNE	DISTLPI);CONTINUE LOOP
001026	000261				SEC		
001030	006100				ROL	X0);ADVANCE ZERO
001032	022700	177777			CMP	#-1,X0);CKN FULL CIRCLE
001036	001401				BEQ	NEXLOC);MOVE ONE TO NX LOC
001040	000762				BR	SETLP);CONTINUE SEQUENCE
001042	012700	177776		NEXLOCI	MOV	#-2,X0	
001046	021121				CMP	@X1,(1)+);JUST INCREMENTING
001050	020167	177132			CMP	X1,HILMT);CKN HILMT
001054	103402				BLO	COOLCK);NO HILMT YET CK NX LOC FOR DISTURB
001056	000167	177554			JMP	ITERCT	
001062	022711	177777		COOLCKI	CMP	#-1,@X1);CKN FOR ALL 1'S
001066	001402				BEQ	,+6	
001070	004767	000116			JSR	X7,ERROR4	
001074	000744				BR	SETLP);BACK TO DISTURB LOOP
001076	012700	000001		ZIPCKI	MOV	#1,X0);ONE TO R0
001102	016701	177076			MOV	L0LMT,X1);GET L0LMT
001106	012767	000012	000326	FIXLPI	MOV	#12,NOISY);SET WRITE-READ LOOP
001114	010011			NOYZLPI	MOV	X0,@X1);WRITE
001116	020011				CMP	X0,@X1);READ
001120	005367	000316			DEC	NOISY	
001124	001373				BNE	NOYZLP);CONTINUE LOOP
001126	000241				CLC		
001130	006100				ROL	X0);ADVANCE 1
001132	001401				BEQ	GETNX);MOVE ON TO NX LOC
001134	000764				BR	FIXLP);CONTINUE SEQUENCE
001136	012700	000001		GETNXI	MOV	#1,X0	
001142	021121				CMP	@X1,(1)+);JUST INCREMENTING
001144	020167	177036			CMP	X1,HILMT);CKN HILMT
001150	103402				BLO	SYMCK);NO HILMT YET CK NX LOC FOR DISTURB
001152	000167	177460			JMP	ITERCT);GO INCREMENT PASS COUNTER
001156	005711			SYMCKI	TST	@X1);CKN FOR ZEROS
001160	001402				BEQ	,+6	
001162	004767	000016			JSR	X7,ERROR3	
001166	000747				BR	FIXLP);BACK TO DISTURB LOOP
001170	012702	002274		ERROR1I	MOV	#MSG2,X2);GET ERROR1 MSG
001174	000410				BR	ERRMSG);PRINTS MOVING0 WJR ERROR
001176	012702	002365		ERROR2I	MOV	#MSG4,X2);GET ERROR2 MSG
001202	000405				BR	ERRMSG);PRINTS MOVING1 WJR ERROR
001204	012702	002330		ERROR3I	MOV	#MSG3,X2);GET ERROR3 MSG
001210	000402				BR	ERRMSG);PRINTS MOVING0 DISTURBANCE ERROR
001212	012702	002421		ERROR4I	MOV	#MSG5,X2);PRINTS MOVING1 DISTURBANCE ERROR
001216	032777	020000	176770	ERRMSGI	BIT	#20000,@SWREG);SET UP MASK IN PWSWK
001224	001030				BNE	SWCHK);UP, GO CK ERROR CONTROLS
001226	004767	000506			JSR	X7,TOP);OK, GO PRINT ERROR MSG
001232	010102			ERRPTI	MOV	X1,X2);GET ERROR ADDRESS
001234	004767	000204			JSR	X7,PRTAB);GO TO PRINT ROUTINE

001240	012732	002077		SPACE:	MOV	#SPOCK,X2	
001244	004767	000470			JSR	X7, TOP	;PRINT 6 SPACES
001250	011102				MOV	@X1,X2	;GET ERROR DATA
001252	004767	000166			JSR	X7, PRTAB	;GO TO PRINT ROUTINE
001256	105777	176726			TSTB	@TCSR	ICK TTY AVAIL
001262	100375				BPL	,=4	
001264	012777	000215	176720		MOV	#215,@TDBR	;SEND CR
001272	105777	176712			TSTB	@TCSR	ICK TTY AVAIL
001276	100375				BPL	,=4	
001300	012777	000212	176704		MOV	#212,@TDBR	;SEND LF
001306	032777	140000	176700	SWCHK:	BIT	#140000,@SWREG	
001314	001005				BNE	CKS45	;REQUEST UP
001316	005067	000112			CLR	FAIL0	
001322	005067	000110			CLR	FAIL1	
001326	000207				RTS	X7	;NO REQUEST-RESUME TEST
001330	032777	100000	176656	CKS45:	BIT	#100000,@SWREG	
001336	001405				BEG	CKFAIL	ICK ERROR12 IF NO HALT
001340	012702	002107			MOV	#MSG6,X2	;SET MSG2 ADDRESS
001344	004767	000370			JSR	X7, TOP	ITO TTY OUTPUT
001350	000000				HALT		WAIT FOR CONTINUE
001352	005767	000056		CKFAIL:	TST	FAIL0	ICK ERROR1
001356	100404				BMI	SCOPE	YUP
001360	005767	000052			TST	FAIL1	ICK ERROR2
001364	100401				BMI	SCOPE	YUP
001366	000207				RTS	X7	;NOT ERROR 1 OR 2 RESUME TEST
001370	032777	040000	176616	SCOPE:	BIT	#40000,@SWREG	
001376	001005				BNE	LOOPON	;SCOPE SWITCH UP
001400	005067	000030			CLR	FAIL0	
001404	005067	000026			CLR	FAIL1	
001410	000207				RTS	X7	
001412	010011			LOOPON:	MOV	X0,@X1	WRITE
001414	020011				CMP	X0,@X1	READ
001416	001764				BEG	SCOPE	
001420	032777	020000	176566		BIT	#20000,@SWREG	
001426	001360				BNE	SCOPE	
001430	000167	177576			JMP	ERRPT	ICK PRINT
001434	000000			FAIL0:	0		
001436	000000			FAIL1:	0		
001440	000000			BELLCT:	0		
001442	000000			NOISY:	0		
001444	005067	000250		PRTAB:	CLR	BINCT	
001450	005067	000242			CLR	WGICT	
001454	012704	001726			MOV	#LIST,X4	;GET LIST ADDRESS
001460	142777	000177	176522		BICB	#177,@TCSR	;CLR INT FLAG
001466	012767	000005	000230		MOV	#5,ASCNT	
001474	012767	000007	000210		MOV	#7,SEVEN	
001502	012767	000001	000204		MOV	#1,DECML	

001510	105777	176474	WAIT1	TSTB	@TCSR
001514	100375			BPL	WAIT1
001516	005702			TST	X2
001520	100404			BMI	MINUS ;NEG SIGN PRINT 1
001522	012777	000260	176462	MOV	#260,@TDBR ;POS SIGN PRINT 0
001530	000403			BR	STARTX
001532	012777	000261	176452	MINUS	MOV #261,@TDBR
001540	016703	000146		STARTX	MOV SEVEN,X3 ;PUT MASK IN R3
001544	010267	000152		MOV	X2,TOODLE ;GET READY TO DOODLE NUMBER IN TOODLE
001550	005167	000146		COM	TOODLE ;THIS COMPENSATES FOR COMPLEMENT DURING BIC
001554	046703	000142		BIC	TOODLE,X3 ;ANDDED RESULTS ARE IN R3
001560	001410			BEQ	WRTOC ;ZERO, WRITE 0 IN LIST
001562	066767	000126	000126	MKNUM	ADD DECML,WGTCT ;COUNT UP TO
001570	005267	000124		INC	BINCT ;AND RECORD
001574	026703	000116		CMP	WGTCT,X3 ;SAME BINARY WEIGHT
001600	001370			BNE	MKNUM ;KEEP COUNT
001602	062767	000260	000110	WRTOC	ADD #260,BINCT ;ADD ASCII PREFIX
001610	016724	000104		MOV	BINCT,(4)+ ;WRITE ASCII CHAR IN LIST
001614	066767	000072	000072	ADD	SEVEN,DECML ;EXPAND BINARY WEIGHT
001622	005067	000070		CLR	WGTCT
001626	005067	000066		CLR	BINCT
001632	005367	000066		DEC	ASCNT
001636	001410			BEQ	XLIST ;5 CHAR IN LIST
001640	012703	000003		MOV	#3,X3 ;SET X3 FOR ADD LOOP
001644	066767	000042	000040	MOADD	ADD SEVEN,SEVEN ;MAKING SEVENTY BY SEVEN
001652	005303			DEC	X3
001654	001373			BNE	MOADD
001656	000730			BR	STARTX ;INX SEVEN SET GET NX OCTAL
001660	012767	000005	000036	XLIST	MOV #5,ASCNT ;SEND 5 CHAR TO TTY
001666	105777	176316		WAIT2	TSTB @TCSR
001672	100375			BPL	WAIT2
001674	014477	176312		MOV	-(4),@TDBR
001700	005367	000020		DEC	ASCNT
001704	001401			BEQ	HDFHM
001706	000767			BR	WAIT2
001710	000207			HDFHM	RTS ;HEAD FOR HOME
001712	000000			SEVEN	0
001714	000000			DECML	0
001716	000000			WGTCT	0
001720	000000			BINCT	0
001722	000000			TOODLE	0
001724	000000			ASCNT	0
001726	000000			LIST	0
001730	000000				0
001732	000000				0
001734	000000				0
001736	000000				0

```

001740 142777 000177 176242 TOP1 BICB #177,@TCSR ICLR INT FLAG
001746 105777 176236 TSTB @TCSR
001752 100375 BPL ,=4
001754 112777 000215 176230 MOVB #215,@TDBR ISEND CARRIAGE RETURN
001762 105777 176222 TSTB @TCSR
001766 100375 BPL ,=4
001770 112777 000212 176214 MOVB #212,@TDBR ISEND LINE FEED
001776 112267 000074 MOVB (2)+,EOMK IMOVE IN EOM MARKER
002002 121267 000070 TOP1 CMPB @X2,EOMK ICOMPARE FOR EOM
002006 001411 BEQ ATRATE
002010 121227 000100 CMPB @X2,#'0
002014 001406 BEQ ATRATE
002016 105777 176166 TSTB @TCSR ICK TTY
002022 100375 BPL ,=4 IWAIT FOR DONE
002024 112277 176162 MOVB (2)+,@TDBR IMOVE CHARACTER
002030 000764 BR TOP1 IBRANCH BACK
002032 105777 176152 ATRATE: TSTB @TCSR
002036 100375 BPL ,=4
002040 112777 000215 176144 MOVB #215,@TDBR ISEND CARRIAGE RETURN
002046 105777 176136 TSTB @TCSR
002052 100375 BPL ,=4
002054 112777 000212 176130 MOVB #212,@TDBR ISEND LINEFEED
002062 121267 000010 CMPB @X2,EOMK ICRN END MESSAGE MARK
I NOT EOM
002066 001402 BEQ ,+6 ISKP NX2 IF EOMK
002070 005202 INC X2 IINCRMTN R2
002072 000743 BR TOP1 INO EOM, SO LOOP
002074 000207 RTS X7
002076 000 EOMK: ,BYTE 0
002077 057 020040 020040 SPOCK: ,ASCII 1/ 1/
002104 020040 057
002107 054 047524 051440 MSG6: ,ASCII,,TO SCOPE-CLEAR HALT SW-SET SCOPE SW, AND CONTINUE.0)
002114 047503 042520 041455
002122 042514 051101 044040
002130 046101 020124 053523
002136 051455 052105 051440
002144 047503 042520 051440
002152 027127 040440 042116
002160 041440 047117 044524
002166 052516 027105 100
002173 116 020117 041523 ,ASCII:INO SCOPE ON DISTURBANCE ERROR.0)
002200 050117 020105 047117
002206 042040 051511 052524
002214 041122 047101 042503
002222 042440 051122 051117
002230 040056
002232 052522 020116 047527 ,ASCII:IRUN WORST NOISE ON DISTURB ERROR.,)
002240 051522 020124 047516
002246 051511 020105 047117
002254 042040 051511 052524
002262 041122 042440 051122
002270 051117 026056
002274 046454 053117 047111 MSG2: ,ASCII,,MOVING 0 READ-WRITE ERROR.,)
002302 020107 020060 042522

```


002310	042101	053455	044522
002316	042524	042440	051122
002324	051117	026056	
002330	046454	053117	047111
002336	020107	020060	044504
002344	052123	051125	040502
002352	041516	020105	051105
002360	047522	027122	054
002365	054	047515	044526
002372	043516	030440	051040
002400	040505	026504	051127
002406	052111	020105	051105
002414	047522	027122	054

MSG3: ,ASCII,MOVING 0 DISTURBANCE ERROR,,)

MSG4: ,ASCII,MOVING 1 READ=WRITE ERROR,,)

002421	054	047515	044526
002426	043516	030440	042040
002434	051511	052524	041122
002442	047101	042503	042440
002450	051122	051117	026056
002456	000000		
	002556		
002556	000000		
002560	000000		

MSG5: ,ASCII,MOVING 1 DISTURBANCE ERROR,,)

TERMNL: 0 ,EVEN
; TERMNL+100

BUFFER: 0

ENDBUF: 0

002562	051457	052105	046440	MSG11	,ASCII	1/SET MEMORY ADDRESS LIMITS VIA SWITCH REGISTER0;
002570	046525	051117	020131			
002576	042121	051104	051505			
002604	020123	044514	044515			
002612	051524	053040	040511			
002620	051440	044527	041524			
002626	020110	042522	044507			
002634	052123	051105	100			
002641	123	052105	046040	,ASCII		1/SET LOWER LIMIT IN SW-REG AND PRESS CONTINUE0;
002646	053517	051105	046040			
002654	046511	052111	044440			
002662	020116	053523	051055			
002670	043505	040440	042116			
002676	050040	042522	051523			
002704	041440	047117	044524			
002712	052516	040105				
002716	042523	020124	050125	,ASCII		1/SET UPPER LIMIT IN SW-REG AND PRESS CONTINUE0;
002724	042520	020122	044514			
002732	044515	020124	047111			
002740	051440	026527	042522			
002746	020107	047101	020104			
002754	051120	051505	020123			
002762	047503	052116	047111			
002770	042525	100				
002773	125	042116	043105	,ASCII		1/UNDEFINED LIMITS WILL BE SET TO 2560-174700;
003000	047111	042105	046040			
003006	046511	052111	020123			
003014	044527	046114	041040			
003022	020105	042523	020124			
003030	047524	031040	033065			
003036	026460	033461	033464			
003044	040060					
003046	044514	044515	051524	,ASCII		1/LIMITS BELOW THIS RANGE WILL BE DEFAULTED0;
003054	041040	046105	053517			
003062	052040	044510	020123			
003070	040522	043516	020105			
003076	044527	046114	041040			
003104	020105	042504	040506			
003112	046125	042524	040104			
003120	053523	030461	030075	,ASCII		1/SW11=0 DO PART 1 "WRITE-READ MOVING 1'S AND 0'S0;
003126	042040	020117	040520			
003134	052122	030440	021040			
003142	051127	052111	026505			
003150	042522	042101	046440			
003156	053117	047111	020107			
003164	023461	020123	047101			
003172	020104	023460	040123			
003200	053523	030461	030475	,ASCII		1/SW11=1 DO PART 2 "MEMORY DISTURBANCE"/;
003206	042040	020117	040520			
003214	052122	031040	021040			
003222	042515	047515	054522			
003230	042040	051511	052524			
003236	041122	047101	042503			

003244 027442

000001

.EVEN
.END

ALLONE	000444	ASCNT	001724	ATRATE	002032	BELLCT	001440
BINCT	001720	BUFFER	002556	CIPHER	000562	CKFAIL	001352
CKS45	001330	CONSET	000336	CONTK	000712	COOLCK	001062
DECML	001714	DISTLP	001014	DSTURB	000752	ENDADR	000702
ENDBUF	002560	EOMK	002076	ERRMSG	001216	ERROR1	001170
ERROR2	001170	ERROR3	001204	ERROR4	001212	ERRPT	001232
FAIL0	001434	FAIL1	001436	FIXLP	001106	GETNX	001136
HDFHM	001710	HILMT	000206	HISET	000310	ITERCT	000636
LIST	001720	LOLMT	000204	LOOPON	001412	MINUS	001532
MKNUM	001562	MOADD	001644	MSG1	002562	MSG2	002274
MSG3	002330	MSG4	002366	MSG5	002421	MSG6	002107
NEXLOC	001042	NEXONE	000544	NEXZIP	000622	NOISY	001442
NQP	000240	NOYZLP	001114	OKAY0	000530	OKAY1	000612
PRTAB	001444	SCOPE	001370	SEE	000354	SETLP	001006
SEVEN	001712	SPACE	001240	SPOCK	002077	START	000220
STARTX	001540	START1	000342	SWCHK	001306	SWITCH	000216
SHREG	000210	SH1CK	000724	SYMCK	001156	TCSR	000210
TDBR	000212	TERMINL	002456	TLG	000362	TOODLE	001722
TOP	001740	TOP1	002002	WAIT1	001510	WAIT2	001666
WGTCT	001710	WRTOC	001602	WRTONE	000510	WRTZIP	000572
XLIST	001660	ZIPCK	001076	ZIPOUT	000440	ZIPPER	000746
S1	000412	S2	000416	.	003246		

ERRORS DETECTED: 0

MEMORY MOVING ONES AND ZEROES TEST
DEMMEA,P11

MAINDEC-11-DEMME-A

MACY11,616 16-MAY-72 17159 PAGE 15

*DEMMEA,DEMMEA*DEMMEA/SOL
RUN-TIME: 2 4 0 SECONDS
CORE USED: 3K