

MR11

DIAGNOSTIC
MD-11-DZMRA-A

EP-DZMRA-A-DL-A

NOV 1976

COPYRIGHT © 1976

digital

FICHE 1 OF 1

MADE IN USA

The microfiche card contains a grid of frames. The first frame in the top-left corner contains a small diagram of a circuit or system. The subsequent frames contain various data, including lists of numbers, text, and small diagrams. The data appears to be organized in a structured manner, possibly representing a diagnostic procedure or a set of test results. The frames are arranged in a grid that covers most of the card's surface.

1. ABSTRACT

THE DZMRA DIAGNOSTIC PROGRAM IS WRITTEN TO BE USED AS AN AID TO HARDWARE DEBUGGING AND MAINTENANCE OF THE MR11-DB (64 WORD BULK STORAGE BOOTSTRAP LOADER). THESE PROGRAMS MAY ALSO BE USED AS A DATA RELIABILITY TEST.

THE AVAILABLE TESTS ARE

PRG0 - LOGIC TESTS

PRG1 - ROM DATA DUMP

PRG2 - SINGLE ROM ADDRESS READ DATA LOOP

2. REQUIREMENTS

2.1 EQUIPMENT

A. PDP 11 FAMILY CENTRAL PROCESSOR

B. MR11-DB (64 WORD BULK STORAGE BOOTSTRAP LOADER)

2.2 STORAGE

THIS PROGRAM USES CORE 0-4100(8)

3. LOADING AND STARTING PROCEDURE

LOAD PROGRAM INTO MEMORY USING ABS LOADER.

LOAD ADDRESS = 00200

SET SR = DESIRED STANDARD PDP-11 DIAGNOSTIC OPTIONS (SEE SECT 6.0)

NOTE: ALL SWITCHES AUTOMATICALLY SELECTS AND STARTS PROGRAM 0.

DEPRESS START THE PROGRAM WILL TYPE OUT INSTRUCTIONS. ALL USER RESPONSES ARE VIA THE KEYBOARD (CARRIAGE RETURN TERMINATES THE RESPONSE)

TO RESTART THE SELECTED PROGRAM LOAD ADDRESS = 000210 AND DEPRESS START

4.0 SWITCH SETTINGS

SW15 1 OR UP HALT ON ERROR

SW14 1 OR UP SCOPE LOOP

SW13 1 OR UP INHIBIT PRINTOUT

SW12 1 OR UP INHIBIT TRACE TRAPPING (NOT USED)

SW11 1 OR UP INHIBIT ITERATION

5. PROGRAM DESCRIPTIONS

5.1 PRG0 - LOGIC TESTS

THE LOGIC TESTS CONSIST OF 4 ROUTINES TO TEST THE MR11-DB LOGIC

5.1.1 ROUTINE DESCRIPTIONS

ROUTINE

TESTS

T1	ADDRESSABILITY OF MR11-DB
T2	DATA RELIABILITY
T3	THAT MR11-GB TIMES OUT WHEN REFERENCED BY A DATIP BUS CYCLE
T4	THAT DATA READ IS CORRECT

5.1.2 ERROR PRINTOUT

IF A ROUTINE FAILS AND THE INHIBIT PRINTOUT SWITCH IS NOT ENABLED (SR13) A PRINTOUT RESULTS. THE PC AT THE TIME OF FAILURE IS TYPED.

IF AN ERROR OCCURS IN T4 THE ROM DATA AND CORRECT DATA AND THE ADDRESS OF EACH IS TYPED OUT (THE ERROR TYPEOUT CANNOT BE DISABLED). THE FORMAT IS

ROM ADDRESS/ROM DATA
IMAGE ADDRESS*CORRECT DATA

5.2 PRG1 - ROM DATA DUMP

THIS PROGRAM TYPES OUT THE 64 WORDS OF ROM DATA AND HALTS.

5.3 PRG2 - SINGLE ROM ADDRESS READ DATA LOOP

THIS PROGRAM CONTINUOUSLY READS DATA FROM A TYPED IN ROM ADDRESS. TO CHANGE THE ADDRESS TYPE IN A NEW ADDRESS. (MUST BE EVEN)

%
.TITLE TEST DZMRA-A
.NLIST SEQ,MC
.LIST ME
.ABS

;MR11-DB (64 WORD BULK STORAGE BOOTSTRAP LOADER) DIAGNOSTIC
;LOAD ADDRESS=0200
;DEPRESS START
;RESTART ADDRESS=0210
;STACK POINTER IS AT 500

000030 000030
000032 002326
000032 000340

.=30
ERROR
340

100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135

E01

TEST DZMRA-A MACY11 27(732) 04-NOV-76 12:09 PAGE 4
 DZMRAA.CMB

000034	002236		SCOPEC	
000036	000000		0	
			;EQUATE	STATEMENTS
	104000		HLT=EMT	
	104400		SCOPE=TRAP	
	177560		TKCSR=177560	
	177562		TKDBR=177562	
	177564		TPCSR=177564	
	177566		TPDBR=177566	
	177776		PSW=177776	
	177570		SR=177570	
	000500		STKPTR=500	
	000060			;INITIAL STACK SETTING
	000062		TKINTA=60	
	000200		TKINTP=62	
			. =200	
000200	000167	000622	START1: JMP	PRMTRS
	000210		. =210	
000210	000167	000652	START3: JMP	RESTART
	001000		. =1000	

```
001000 001000
001002 000100
001004 004000
001006 000000
001010 000000
001012 000000
001014 000000
001016 000000
001020 001106
001022 001604
001024 001732
001026 012706 000500
001032 005067 000020
001036 005737 177570
001042 001411
001044 004567 000760
001050 002506
001052 004567 001070
001056 000000
001060 004567 000744
001064 002532

WORDS:      =1000
IMAGE:      64
DUMP:       4000
LAST:       0
CHAR:       0
TERM:       0
SRT:        0
ROMADD:     0
PRGTAB:     PRG0
            PRG1
            PRG2

PRMTRS: MOV  #STKPTR,%6      ;SET STACK PTR
        CLR  PRGNUM
        TST  @#SR
        BEQ  RESTART
        JSR  5,TYEM         ;TYPE MESSAGE 'PRG#='
        MB
        JSR  5,RECD         ;RECEIVE DATA AND PUT
        JSR  5,TYEM         ;IT HERE
        MB

001066 012767 173100 177722 RESTART: MOV  #173100,ROMADD ;GET FIRST ROM ADDRESS
001074 016700 177756        MOV  PRGNUM,%0      ;GET PROGRAM #
001100 006300                ASL   %0            ;SHIFT PROGRAM #
001102 000170 001020        JMP   @PRGTAB(0)     ;GO TO PROGRAM

;PROGRAM 0 LOGIC TESTS
PRG0:  MOV  #STKPTR,%6
        MOV  #RESTART,RETURN
;TEST1 TEST ABILITY TO REFERENCE ROM WITHOUT TIMING OUT

001106 012706 000500
001112 012767 001066 001172

T1:    MOV  ROMADD,%0      ;GET ROM ADDRESS
        MOV  WORDS,%1      ;GET ADDRESS COUNTER
        MOV  #ERROR1,4     ;SET UP TIME OUT VECTOR
T1A:   MOV  (0),%3         ;REFERENCE
        TST  (0)+          ;ROM
        ADD  -(0),DUMP
        CMP  (0),(0)
        BITB (0)+,(0)+
        RESET
        SUB  -(0),DUMP
        ADD  #2,%0         ;DELAY
        DEC  %1            ;INCREMENT POINTER
        BNE  T1A           ;DECREMENT ADDRESS COUNTER
        BR   T1B           ;BRANCH IF NOT FINISHED
        BR   T1B           ;GO TO SCOPE LOOP
ERROR1: CMP  (6)+,(6)+     ;REPOSITION STACK
        HLT
        BR   T1A          ;HERE IF ERROR
T1B:   SCOPE              ;LOOP ON ERROR

001120 016700 177672
001124 016701 177650
001130 012767 001172 176646
001136 011003
001140 005720
001142 064067 177636
001146 021010
001150 132020
001152 000005
001154 164067 177624
001160 062700 000002
001164 005301
001166 001363
001170 000403
001172 022626
001174 104000
001176 000757
001200 104400
```

;TEST2 TEST THAT ROM DATA CAN BE READ RELIABLY.

001202	016700	177610		T2:	MOV	ROMADD,%0	;GET ROM ADDRESS
001206	016701	177566			MOV	WORDS,%1	;GET ADDRESS COUNTER
001212	012767	000006	176564		MOV	#6,4	;INITIALIZE TIME OUT VECTOR
001220	005067	177560		T2A:	CLR	DUMP	;INITIALIZE DUMP
001224	011003				MOV	(0),%3	;GET DATA
001226	062067	177552			ADD	(0)+,DUMP	;ADD DATA TO DUMP
001232	166703	177546			SUB	DUMP,%3	;SUBTRACT DATA FROM DATA
001236	001402				BEQ	T2B	;BRANCH IF EQUAL
001240	104000			ERROR2:	HLT		;DATA ERROR
001242	000766				BR	T2A	;LOOP ON ERROR
001244	000005			T2B:	RESET		;DELAY
001246	044067	177532			BIC	-(0),DUMP	;CLEAR DUMP BITS
001252	001402				BEQ	T2C	;BRANCH IF EQUAL TO 0
001254	104000				HLT		;DATA ERROR
001256	000772				BR	T2B	;LOOP ON ERROR
001260	021010			T2C:	CMP	(0),(0)	;COMPARE DATA
001262	001402				BEQ	T2D	;BRANCH IF EQUAL
001264	104000				HLT		;DATA ERROR
001266	000774				BR	T2C	;LOOP ON ERROR
001270	122040			T2D:	CMPB	(0)+,-(0)	;COMPARE DATA (BYTE OPERATION)
001272	001402				BEQ	T2E	;BRANCH IF EQUAL
001274	104000				HLT		;DATA ERROR
001276	000774				BR	T2D	;LOOP ON ERROR
001300	005720			T2E:	TST	(0)+	;INCREMENT ADDRESS POINTER
001302	005301				DEC	%1	;DECREMENT ADDRESS COUNTER
001304	001345				BNE	T2A	;RETURN IF NOT DONE
001306	104400				SCOPE		

;TEST3 TEST THAT ROM TIMES OUT IF REFERENCED BY OTHER
;THAN DATA BUS CYCLE

001310	012706	000500		T3:	MOV	#STKPTR,%6	;SET STACK PTR
001314	016700	177476			MOV	ROMADD,%0	;GET ROM ADDRESS
001320	016701	177454			MOV	WORDS,%1	;GET ADDRESS COUNTER
001324	012767	001340	176452	T3AA:	MOV	#T3B,4	;SET UP TIME OUT VECTOR
001332	010010			T3A:	MOV	%0,(0)	;ATTEMPT TO ALTER DATA
001334	104000				HLT		;HERE IF DID NOT TIME OUT
001336	000775				BR	T3A	;LOOP ON ERROR
001340	012767	001356	176436	T3B:	MOV	#T3D,4	;SET UP TIME OUT VECTOR
001346	022626				CMP	(6)+,(6)+	;REPOSITION STACK
001350	005210			T3C:	INC	(0)	;ATTEMPT TO ALTER DATA
001352	104000				HLT		;HERE IF DID NOT TIME OUT
001354	000775				BR	T3C	;LOOP ON ERROR
001356	012767	001376	176420	T3D:	MOV	#T3F,4	;SET UP TIME OUT VECTOR
001364	022626				CMP	(6)+,(6)+	;REPOSITION STACK
001366	005077	177424		T3E:	CLR	ROMADD	;ATTEMPT TO ALTER DATA
001372	104000				HLT		;HERE IF DID NOT TIME OUT
001374	000774				BR	T3E	;LOOP ON ERROR
001376	005720			T3F:	TST	(0)+	;INCREMENT ADDRESS POINTER
001400	022626				CMP	(6)+,(6)+	;REPOSITION STACK
001402	005301				DEC	%1	;DECREMENT ADDRESS COUNTER
001404	001347				BNE	T3AA	;RETURN IF NOT DONE
001406	012737	000006	000004		MOV	#6,%4	;RESTORE TIME OUT TRAP

H01

TEST DZMRA-A MACY11 27(732) 04-NOV-76 12:09 PAGE 7
DZMRAA.CMB

001414 104400

SCOPE

;SCOPE LOOP

;THIS TEST COMPARES ROM AND IMAGE DATA
;AND TYPES OUT DIFFERENCES

001416	012706	000500	T4:	MOV	#STKPTR,%6	;SET STACK PTR	
001422	016701	177352		MOV	WORDS,%1	;GET # OF WORDS	
001426	016700	177364		MOV	ROMADD,%0	;GET ROM ADDRESS	
001432	016703	177344		MOV	IMAGE,%3	;GET IMAGE ADDRESS	
001436	021013		T4B:	CMP	(0),(3)	;COMPARE DATA	
001440	001004			BNE	T4D		
001442	005301		T4C:	DEC	%1	;ALL DATA BEEN COMPARED	
001444	001437			BEQ	T4E		
001446	022023			CMP	(0)+,(3)+	;INCREMENT ADDRESS POINTERS	
001450	000772			BR	T4B		
001452	010067	000712	T4D:	MOV	%0,D2BTYP	;TYPE	
001456	004767	000710		JSR	7,02A	;ROM ADDRESS	
001462	004567	000342		JSR	5,TYPEN	;TYPE	
001466	002616			M10		;SEPARATOR	
001470	011067	000674		MOV	(0),D2BTYP	;TYPE	
001474	004767	000672		JSR	7,02A	;ROM DATA	
001500	004567	000324		JSR	5,TYPEN	;TYPE	
001504	002532			M8		;CR/LF	
001506	010367	000656		MOV	%3,D2BTYP	;TYPE	
001512	004767	000654		JSR	7,02A	;IMAGE ADDRESS	
001516	004567	000306		JSR	5,TYPEN	;TYPE	
001522	002624			M12		;SEPARATOR	
001524	011367	000640		MOV	(3),D2BTYP	;TYPE	
001530	004767	000636		JSR	7,02A	;IMAGE DATA	
001534	004567	000270		JSR	5,TYPEN	;TYPE	
001540	002532			M8		;CR/LF	
001542	000737			BR	T4C	;GO TO T4C	
001544	104400		T4E:	SCOPE			
001546	012737	000207	177566	END:	MOV	#207,0#TPDBR	;RING THE BELL
001554	105737	177564		TSTB	0#TPCSR		
001560	100375			BPL	-4		
001562	013700	000042		MOV	0#42,%0	;RETURN TO DECTAPE MONITOR?	
001566	001404			BEQ	DONE1		
001570	004710			JSR	7,(0)	;RETURN!	
001572	000240			NOP			
001574	000240			NOP			
001576	000240			NOP			
001600	000167	177302	DONE1:	JMP	PRG0		

;THIS PROGRAM TYPES OUT ROM DATA

001604	012706	000500	PRG1:	MOV	#STKPTR,%6	;INITIALIZE STACK
001610	004567	000214		JSR	5,TYPEN	;TYPE MESSAGE
001614	002516			M7		; 'ROM DATA'
001616	016701	177156		MOV	WORDS,%1	;GET # OF WORDS
001622	016700	177170	PRG1A:	MOV	ROMADD,%0	;GET STARTING ADDRESS
001626	012702	000012		MOV	#12,%2	;GET ADDRESS INDICATOR
001632	105767	175726		TSTB	TPCSR	;WAIT FOR
001636	100375			BPL	-4	;TELEPRINTER FLAG

001640	010067	000524		PRG1B:	MOV	%0,D2BTYP	;GET ADDRESS
001644	004767	000522			JSR	7,02A	;AND TYPE IT
001650	004567	000154			JSR	5,TYPEM	;TYPE
001654	002532				MB		;CR/LF
001656	012067	000506		PRG1C:	MOV	(0)+,D2BTYP	;TYPE
001662	004767	000504			JSR	7,02A	;DATA
001666	105767	175672			TSTB	TPCSR	;WAIT FOR
001672	100375				BPL	.-4	;TELEPRINTER FLAG
001674	012767	000040	175664		MOV	#',TPDBR	;TYPE SPACE
001702	005301				DEC	%1	;ALL DATA TYPED
001704	001410				BEQ	PRG1D	;GO TO FINISH
001706	005302				DEC	%2	
001710	001362				BNE	PRG1C	;RETURN TO PRG1B
001712	012702	000012			MOV	#12,%2	;GET ADDRESS INDICATOR
001716	004567	000106			JSR	5,TYPEM	;TYPE
001722	002532				MB		;CR/LF
001724	000745				BR	PRG1B	;RETURN TO PRG1B
001726	000167	177074		PRG1D:	JMP	PRMTRS	;GO GET NEXT TEST
;THIS PROGRAM CYCLES A SINGLE ADDRESS (ADDRESS MUST BE EVEN) TO CHANGE							
;THE ADDRESS TYPE NEW ADDRESS ON THE TTY.							
001732	012706	000500		PRG2:	MOV	#STKPTR,%6	;INITIALIZE STACK POINTER
001736	012737	002024	000004		MOV	#PRG2C,%4	;LOAD TRAP ERROR VECTOR
001744	005067	176026			CLR	PSW	;CLEAR PROCESSOR STATUS
001750	012767	002002	176102		MOV	#PRG2A,TKINTA	;LOAD KEYBOARD INTERRUPT VECTOR
001756	012767	000340	176076		MOV	#340,TKINTP	;LOAD KEYBOARD PRIORITY
001764	012767	000100	175566		MOV	#100,TKCSR	;SET INTERRUPT ENABLE BIT
001772	016700	177020			MOV	ROMADD,%0	;GET ROM ADDRESS
001776	005710				TST	(0)	;READ ROM ADDRESS
002000	000776				BR	.-2	;LOOP
002002	004567	000140		PRG2A:	JSR	5,RECD	;GO GET ADDRESS &
002006	000000			PRG2B:	0		;PUT IT HERE
002010	016700	177772			MOV	PRG2B,%0	
002014	004567	000010			JSR	5,TYPEM	;TYPE
002020	002532				MB		;CR/LF
002022	000002				RTI		;EXIT KEYBOARD INTERRUPT SERVICE
002024	104000			PRG2C:	HLT		;ERROR! DID YOU TYPE AN ODD ADDRESS?
002026	000777				BR	.	;SIT HERE UNTIL CORRECT ADDRESS IS TYPED IN
002030	010026			TYPEM:	MOV	%0,(6)+	;SAVE REGISTER 0
002032	012500				MOV	(5)+,%0	;PLACE MESSAGE ADDRESS IN R0
002034	112067	176752			MOVB	(0)+,TERM	;GET TERMINATOR CHARACTER
002040	112067	176744		TYPEMA:	MOVB	(0)+,CHAR	;GET NEXT CHARACTER
002044	126767	176740	176740		CMPB	CHAR,TERM	;WAS NEXT CHARACTER THE TERM
002052	001005				BNE	TYPEMB	;CHARACTER
002054	014600				MOV	-(6),%0	;RESTORE R0
002056	105767	175502			TSTB	TPCSR	
002062	100375				BPL	.-4	
002064	000205				RTS	5	;AND EXIT
002066	126727	176716	000045	TYPEMB:	CMPB	CHAR,#'%	;WAS CHARACTER %
002074	001015				BNE	TYPEMC	
002076	105767	175462			TSTB	TPCSR	;TEST TELEPRINTER FLAG
002102	100375				BPL	.-4	;AND WAIT FOR DONE
002104	012767	000215	175454		MOV	#215,TPDBR	;LOAD TELEPRINTER WITH CAR. RET

```

002112 105767 175446 TSTB TPCSR ;TEST TELEPRINTER FLAG
002116 100375 BPL -4 ;AND WAIT FOR DONE
002120 012767 000212 175440 MOV #212,TPDBR ;LOAD TELEPRINTER WITH LINE FEED
002126 000744 BR TYPEMA ;GET NEXT CHARACTER
002130 105767 175430 TYPEMC: TSTB TPCSR ;TEST TELEPRINTER FLAG
002134 100375 BPL -4 ;AND WAIT FOR DONE
002136 016767 176646 175422 MOV CHAR,TPDBR ;LOAD TELEPRINTER BUFFER
002144 000735 BR TYPEMA ;AND GET NEXT CHARACTER
002146 005015 RECD: CLR (5) ;CLEAR OUT OLD DATA
002150 105767 175404 RECD: TSTB TKCSR ;TEST KEYBOARD FLAG
002154 100375 BPL -4 ;AND WAIT FOR CHARACTER
002156 116767 175400 176624 MOVB TKDBR,CHAR ;GET CHARACTER
002164 016767 176620 175374 MOV CHAR,TPDBR ;ECHO CHARACTER
002172 126727 176612 000215 CMPB CHAR,#215 ;WAS CHARACTER CARRIAGE RETURN
002200 001005 BNE RECD ;INCREMENT RETURN ADDRESS
002202 005725 TST (5)+
002204 105767 175354 TSTB TPCSR
002210 100375 BPL -4
002212 000205 RTS 5 ;AND EXIT
002214 042767 177770 176566 RECD: BIC #177770,CHAR ;STRIP AWAY ALL BUT 3 LSB
002222 006315 ASL (5) ;ROTATE
002224 006315 ASL (5) ;PREVIOUS
002226 006315 ASL (5) ;DATA
002230 056715 176554 BIS CHAR,(5) ;AND INSERT CHARACTER
002234 000745 BR RECD ;GET NEXT CHARACTER
;SCOPE OR/AND ITERATION LOOP FOR EACH TEST 100.TIMES

002236 032767 040000 175324 SCOPEC: BIT #40000,SR ;TEST SR FOR SCOPE
002244 001023 BNE SCOPEB ;YES SCOPE
002246 032767 004000 175314 BIT #4000,SR ;TEST FOR ITERATION
002254 001007 BNE SCOPEB ;INHIBIT ITERATION
002256 026767 000026 000022 CMP SCOPEF,ICOUNT ;ITERATION COMPLETE
002264 001403 BEQ SCOPEB ;ITERATION COMPLETE GO TO SCOPEB
002266 005267 000016 INC SCOPEF ;INCREMENT ITERATION COUNT
002272 000410 BR SCOPEB ;GO TO SCOPEB
002274 005067 000010 SCOPEG: CLR SCOPEF ;CLEAR ITERATION COUNT
002300 011667 000006 MOV #2%6,RETURN ;GET ADDRESS OF NEXT TEST
002304 000002 RTI ;EXIT
002306 000040 ICOUNT: 40 ;CONTAINS SUBTEST ITERATION COUNT
002310 000000 SCOPEF: 0
002312 001066 RETURN: RESTART
002314 005726 SCOPEB: TST(6)+ ;POP PC
002316 012667 175454 MOV (6)+,PSW ;RESTORE CONDITION CODES
002322 000177 177764 JMP #RETURN
002326 036727 175236 020000 ERROR: BIT SR,#20000 ;INHIBIT PRINTOUT?
002334 001401 BEQ .+4 ;BRANCH IF ERROR PRINT OUT
002336 000002 RTI ;RETURN TO TEST
002340 004567 177464 JSR %5,TYPEM ;TYPE ERROR MESSAGE
002344 002476 ERRORM ;PC=
002346 011667 000016 MOV (6),D2BTYP ;TYPE PROGRAM COUNTER
002352 004767 000014 JSR 7,02A
002356 005767 175206 TST SR ;HALT ON ERROR?
002362 100001 BPL .+4
002364 000000 HALT
002366 000002 RTI ;YES HALT
;RETURN TO TEST

```

002370	000000		D2BTYP: 0		
002372	016746	175166	02A: MOV	TPCSR, -(6)	;SAVE TPCSR
002376	010246		MOV	%2, -(6)	;SAVE R2
002400	010146		MOV	%1, -(6)	;SAVE R1
002402	010046		MOV	%0, -(6)	;SAVE R0
002404	016700	177760	MOV	D2BTYP, %0	;GET DATA TO BE TYPED
002410	012701	000006	MOV	#6, %1	;GET COUNTER
002414	005002		CLR	%2	;CLEAR WORKING REGISTER
002416	006100		ROL	%0	;MOV FIRST BIT (MSB) INTO
002420	006102		ROL	%2	;R2
002422	062702	000260	02AA: ADD	#260, %2	;FORM ASCII CODE
002426	105767	175132	TSTB	TPCSR	;TEST TELEPRINTER
002432	100375		BPL	. -4	;FLAG AND WAIT UNTIL DONE
002434	010267	175126	MOV	%2, TPDBR	;LOAD TELEPRINTER BUFFER
002440	005002		CLR	%2	;CLEAR WORKING REGISTER
002442	006100		ROL	%0	;ROTATE THE
002444	006102		ROL	%2	;NEXT
002446	006100		ROL	%0	;OCTAL CHARACTER
002450	006102		ROL	%2	;INTO
002452	006100		ROL	%0	;REGISTER
002454	006102		ROL	%2	;TWO
002456	005301		DEC	%1	;DECREMENT COUNTER
002460	001360		BNE	02AA	;GO TO 02AA IF NOT 0
002462	012600		MOV	(6)+, %0	;FINISHED. RESTORE REGISTERS
002464	012601		MOV	(6)+, %1	
002466	012602		MOV	(6)+, %2	
002470	012667	175070	MOV	(6)+, TPCSR	;AND TPCSR
002474	000207		RTS	7	;AND EXIT

				:ASCII MESSAGES	
002476	022500	050040	036503	ERRORM:	.ASCII 'a% PC= a'
002504	040040				
002506	022500	051120	021507	M6:	.ASCII 'a%PRG#=a'
002514	040075				
002516	022500	047522	020115	M7:	.ASCII 'a%ROM DATA%a'
002524	040504	040524	040045		
002532	022500	100		M8:	.ASCII 'a%a'
002535	100	051045	046517	M9:	.ASCII 'a%ROM ADDRESS/IMAGE ADDRESS ROM DATA*IMAGE DATA%a'
002542	040440	042104	042522		
002550	051523	044457	040515		
002556	042507	040440	042104		
002564	042522	051523	051040		
002572	046517	042040	052101		
002600	025101	046511	043501		
002606	020105	040504	040524		
002614	040045				
002616	027500	100		M10:	.ASCII 'a/a'
002621	100	040040		M11:	.ASCII 'a a'
002624	025100	100		M12:	.ASCII 'a*a'

003776	003776	.=3776
	000000	WORD
		;DATA CUT INTO THE MR11-DB
004000	010702	010702
004002	000451	000451
004004	177462	177462
004006	000005	000005
004010	010702	010702
004012	000445	000445
004014	177406	177406
004016	000005	000005
004020	010702	010702
004022	000417	000417
004024	177344	177344
004026	000005	000005
004030	004003	004003
004032	100000	100000
004034	024000	024000
004036	010702	010702
004040	000410	000410
004042	172524	172524
004044	060003	060003
004046	060011	060011
004050	000200	000200
004052	100000	100000
004054	010702	010702
004056	000423	000423
004060	176716	176716
004062	000005	000005
004064	010200	010200
004066	005720	005720
004070	012001	012001
004072	005311	005311
004074	005720	005720
004076	012041	012041
004100	031011	031011
004102	001776	001776
004104	005720	005720
004106	031041	031041
004110	001406	001406
004112	000112	000112
004114	173100	173100
004116	000340	000340
004120	010702	010702
004122	000401	000401
004124	177450	177450
004126	000005	000005
004130	010200	010200
004132	005720	005720

TEST DZMRA-A
DZMRAA.CMB

MACY11 27(732) 04-NOV-76 12:09 PAGE 14

004134	012001		012001
004136	012711	177000	012711,177000
004142	011041		011041
004144	032711	100200	032711,100200
004150	001775		001775
004152	100757		100757
004154	005007		005007
004156	000000		000000
004160	173110		173110
004162	000340		000340
004164	173220		173220
004166	000340		000340
004170	173154		173154
004172	000340		000340
004174	173120		173120
004176	000340		000340
	000001		.END

CHAR	001010	173#	373*	374	380	391	396*	397	398	404*	408
DONE1	001600	312	317*								
DUMP	001004	171#	206*	210*	225*	227*	228	233*			
D2BTYP	002370	289*	293*	297*	301*	330*	334*	434*	441#	446	
END	001546	308#									
ERROR	002326	146	429#								
ERRORM	002476	433	470#								
ERROR1	001172	203	215#								
ERROR2	001240	230#									
HLT =	104000	151#	216	230	235	239	243	258	263	268	366
ICOUNT	002306	416	423#								
IMAGE	001002	170#	282								
LAST	001006	172#									
M10	002616	292	486#								
M11	002621	487#									
M12	002624	300	488#								
M6	002506	185	472#								
M7	002516	323	474#								
M8	002532	189	296	304	333	345	364	476#			
M9	002535	477#									
O2A	002372	290	294	296	302	331	335	435	442#		
O2AA	002422	451#	463								
PRGNUM	001056	181*	187#	192							
PRGTAB	001020	177#	194								
PRG0	001106	177	197#	317							
PRG1	001604	178	321#								
PRG1A	001622	325#									
PRG1B	001640	330#	346								
PRG1C	001656	334#	342								
PRG1D	001726	340	347#								
PRG2	001732	179	351#								
PRG2A	002002	354	360#								
PRG2B	002006	361#	362								
PRG2C	002024	352	366#								
PRMTRS	001026	163	180#	347							
PSW =	177776	157#	353*	427*							
RECD	002146	186	360	393#							
RECD A	002150	394#	409								
RECD B	002214	399	404#								
RESTAR	001066	165	183	191#	198	425					
RETURN	002312	198*	421*	425#	428						
ROMADD	001016	176#	191*	201	222	254	267*	281	325	357	
SCOPE =	104400	152#	218	248	275	306					
SCOPEB	002314	413	419	426#							
SCOPEC	002236	148	412#								
SCOPEF	002310	416	418*	420*	424#						
SCOPEG	002274	415	417	420#							
SR =	177570	158#	182	412	414	429	436				
SRT	001014	175#									
START1	000200	163#									
START3	000210	165#									
STKPTR=	000500	159#	180	197	253	279	321	351			
TERM	001012	174#	372*	374							
TKCSR =	177560	153#	356*	394							
TKDBR =	177562	154#	396								
TKINTA=	000060	160#	354*								

TEST DZMRA-R
DZMRAA.CMB

MACY11 27(732) 04-NOV-76 12:09 PAGE 17
CROSS REFERENCE TABLE -- USER SYMBOLS

TKINTP=	000062	161#	355*											
TPCSR =	177564	155#	309	327	336	377	382	385	389	401	442	452	467*	
TPDBR =	177566	156#	308*	338*	384*	387*	391*	397*	454*					
TYPEM	002030	184	188	291	295	299	303	322	332	344	363	370#	432	
TYPEMA	002040	373#	388	392										
TYPEMB	002066	375	380#											
TYPEMC	002130	381	389#											
T1	001120	201#												
T1A	001136	204#	213	217										
T1B	001200	214	218#											
T2	001202	222#												
T2A	001220	225#	231	247										
T2B	001244	229	232#	236										
T2C	001260	234	237#	240										
T2D	001270	238	241#	244										
T2E	001300	242	245#											
T3	001310	253#												
T3A	001332	257#	259											
T3AA	001324	256#	273											
T3B	001340	256	260#											
T3C	001350	262#	264											
T3D	001356	260	265#											
T3E	001366	267#	269											
T3F	001376	265	270#											
T4	001416	279#												
T4B	001436	283#	288											
T4C	001442	285#	305											
T4D	001452	284	289#											
T4E	001544	286	306#											
WORDS	001000	169#	202	223	255	280	324							
.	= 004200	145#	162#	164#	166#	168#	310	328	337	359	367	378	383	386
		390	395	402	430	437	453	489#						

TEST DZMRA-A
DZMRAA.CMBMACY11 27(732) 04-NOV-76 12:09 PAGE 19
CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ADD	206	211	227	451												
ASL	193	405	406	407												
BEQ	183	229	234	238	242	286	312	340	417	430						
BIC	233	404														
BIS	408															
BIT	412	414	429													
BITB	208															
BNE	213	247	273	284	342	375	381	399	413	415	463					
BPL	310	328	337	378	383	386	390	395	402	437	453					
BR	214	217	231	236	240	244	259	264	269	288	305	346	359	367	388	
	392	409	419													
CLR	181	225	267	353	393	420	448	455								
CMP	207	215	237	261	266	271	283	287	416							
CMPB	241	374	380	398												
DEC	212	246	272	285	339	341	462									
ENT	151															
HALT	145	438														
INC	262	418														
JMP	163	165	194	317	347	428										
JSR	184	186	188	290	291	294	295	298	299	302	303	313	322	331	332	
	335	344	360	363	432	435										
MOV	180	191	192	197	198	201	202	203	204	222	223	224	226	253	254	
	255	256	257	260	265	274	279	280	281	282	289	293	297	301	308	
	311	321	324	325	326	330	334	338	343	351	352	354	355	356	357	
	362	370	371	376	384	387	391	397	421	427	434	442	443	444	445	
	446	447	454	464	465	466	467									
MOVB	372	373	396													
NOP	314	315	316													
RESET	209	232														
ROL	449	450	456	457	458	459	460	461								
RTI	365	422	431	439												
RTS	379	403	468													
SUB	210	228														
TRAP	152															
TST	182	205	245	270	358	400	426	436								
TSTB	309	327	336	377	382	385	389	394	401	452						
.ABS	138															
.ASCII	470	472	474	476	477	486	487	488								
.END	563															
.LIST	137	145														
.NLIST	136	145														
.REM	1															
.REPT	145															
.TITLE	135															
.WORD	490															

ERRORS DETECTED: 0
 DEFAULT GLOBALS GENERATED: 0

*.NOW.SEQ/SOL/CRF/NL:TOC/PAGNUM=DZMRAA.CMB

RUN-TIME: 13.7 SECONDS

RUN-TIME RATIO: 26/5=4.6

CORE USED: 6K (11 PAGES)

F02

TEST DZMRA-A
DZMRAA.CMB

NACV11 27(732) 04-NOV-76 12:09 PAGE 20
CROSS REFERENCE TABLE -- PERMANENT SYMBOLS