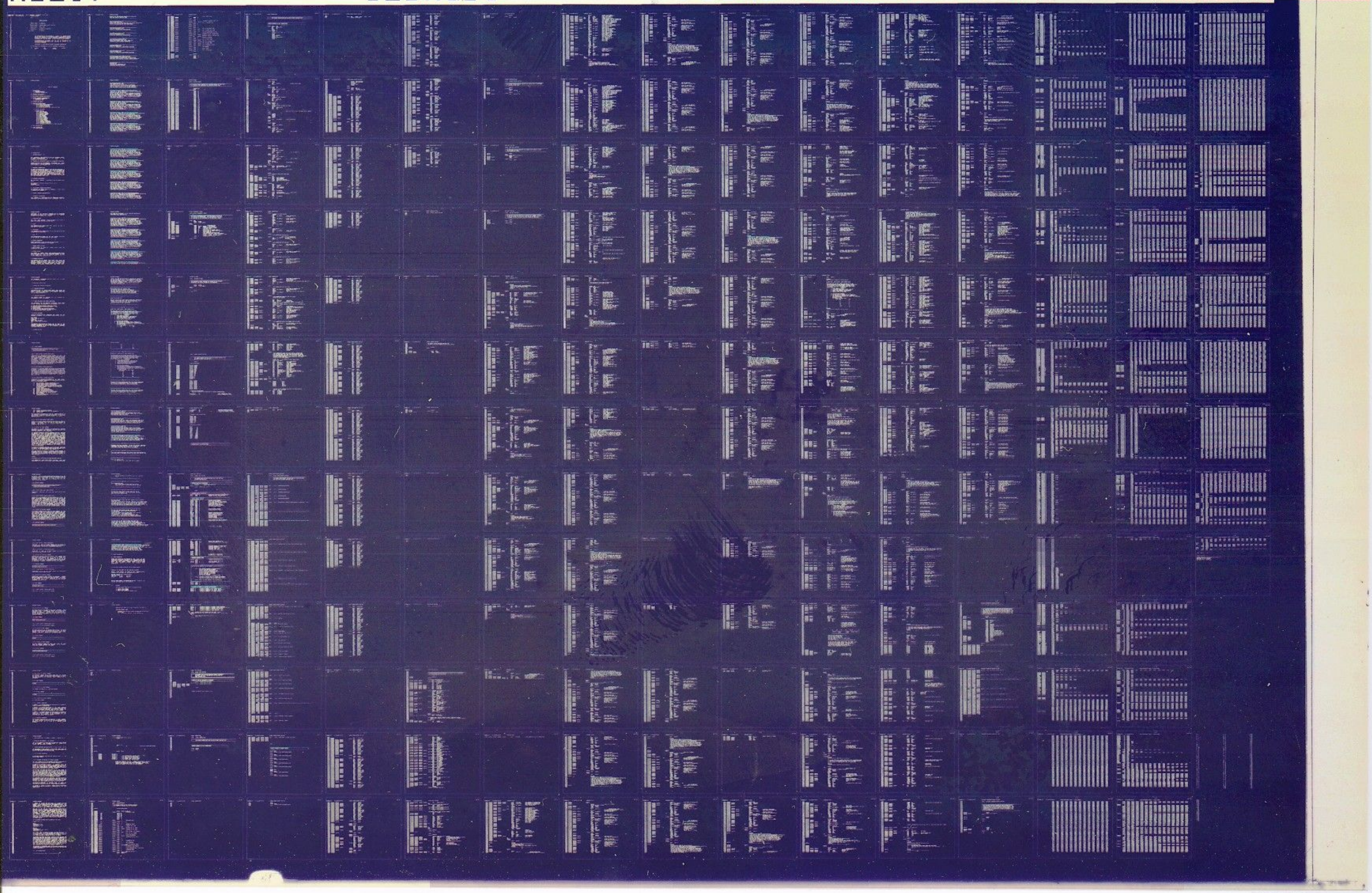
DMP-11,DMR-11, M8207 STATIC DIAG#2 AH-E229D-MC JUL 1982
M8207 CZDMQDO FICHE 1 OF 1



```
                .REM @

                        IDENTIFICATION
                        --------------

PRODUCT CODE:   AC-E228D-MC
PRODUCT NAME:   CZDMQDO M8207 STATIC DIAG #2
PRODUCT DATE:   APRIL 1982
MAINTAINER:     DIAGNOSTICS MERRIMACK
AUTHOR:         ED BADGER
```

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 PROGRAM ABSTRACT

THIS DIAGNOSTIC WAS DESIGNED TO TEST OUT THE M8200, M8204, OR M8207 MICROPROCESSOR. IT IS THE SECOND OF TWO DIAGNOSTICS FOR THESE OPTIONS.

THE PROGRAM WAS IMPLEMENTED USING THE DIAGNOSTIC SUPERVISOR.

THROUGH DIALOGUE WITH THE OPERATOR, THE PROGRAM WILL ALLOW MODIFICATION OF DEVICE PARAMETERS, SUCH AS UNIBUS ADDRESS, VECTOR ADDRESS, AND PROCESSOR TYPE.

1.2 HARDWARE INTRODUCTION

THE M820X MICROPROCESSOR USES AN EIGHT BIT DATA PATH WITH A SIXTEEN BIT INSTRUCTION MEMORY. THE INSTRUCTION MEMORY AND DATA MEMORY ARE TWO SEPARATE MEMORIES. THE MICROPROCESSOR IS DESIGNED FOR MOVING DATA AT HIGH RATES TO WORK AS A HIGH SPEED LINK BETWEEN PROCESSORS WHEN USED WITH A LINE UNIT. THE M8200 AND M8207 HAVE PROM INSTRUCTION MEMORIES. THE M8204 HAS WRITEABLE CONTROL STORE. THE MEMORY SIZES BETWEEN ALL THREE PROCESSORS VARY ALSO.

2.0 HARDWARE REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE M8207 LOGIC TESTS:

PDP-11/04,05.10.20,30,34,35,40,45,50,60, OR 70
16K MEMORY
CONSOLE TERMINAL

3.0 PRELIMINARY PROGRAM REQUIREMENTS

THE PROCESSOR AND MEMORY SHOULD BE THOROUGHLY TESTED PRIOR TO RUNNING THIS DIAGNOSTIC.

4.0 GENERAL PROGRAM CONSIDERATIONS

4.1 DIAGNOSTIC SUPERVISOR

THIS PROGRAM IS COMPATIBLE WITH THE STANDALONE DIAGNOSTIC SUPERVISOR, AND MUST BE LOADED TO BE CO-RESIDENT WITH THE

SUPERVISOR, OR BE PREVIOUSLY COMBINED WITH THE SUPERVISOR AND LOADED AS A SINGLE FILE. IN EITHER CASE, THE COMBINED PROGRAM WILL NOT EXCEED 16K OF MEMORY.

4.2 EXECUTION TIME

THE TOTAL TIME REQUIRED TO RUN THE M8207 STATIC TESTS IS ABOUT 120 SECONDS PER PASS FOR EACH UNIT.

4.3 XXDP+

THIS PROGRAM MAY BE LOADED UNDER XXDP+, AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

4.4 ACT/SLIDE

THIS PROGRAM MAY BE LOADED UNDER ACT OR SLIDE AND MAY BE RUN IN DUMP MODE OR CHAIN MODE.

4.5 APT

THIS PROGRAM MAY BE LOADED BY THE APT SYSTEM (INCLUDING APT-RD) AND RUN IN PROGRAM MODE OR SCRIPT MODE.

4.6 MEMORY MANAGEMENT

MEMORY MANAGEMENT IS NOT UTILIZED IN THIS PROGRAM. IF IT IS INSTALLED, IT IS DISABLED BY THE PROGRAM.

4.7 MEMORY PARITY OPTION

IF PARITY MEMORY IS INSTALLED, MEMORY PARITY TRAPS ARE DISABLED BY THE PROGRAM.

4.8 ERROR LOGGING

THE NUMBER OF ERRORS WHICH HAVE OCCURRED ON EACH DEVICE UNDER TEST SINCE THE LAST START OR RESTART COMMAND IS KEPT IN AN ERROR LOG. THIS LOG MAY BE PRINTED BY USING THE "PRINT" COMMAND (SEE SECTION 6.3.8).

5.0 PROGRAM LOAD MEDIA

THIS PROGRAM CAN BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER OR FROM ACT, SLIDE, OR APT SYSTEMS, OR FROM ANY MEDIA SUPPORTED BY XXDP+. WHEN USING THE PAPER TAPE ABSOLUTE LOADER, THE PROGRAM SHOULD BE LOADED FIRST, FOLLOWED BY THE DIAGNOSTIC SUPERVISOR. WHEN USING XXDP+,

THE DIAGNOSTIC SUPERVISOR SHOULD BE LOADED FIRST, FOLLOWED BY THE DIAGNOSTIC PROGRAM.

6.0 OPERATING INSTRUCTIONS

6.1 LOADING AND STARTING PROCEDURES

6.1.1 LOADING PROCEDURES

THIS PROGRAM MAY BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER. IT MAY ALSO BE LOADED FROM ANY XXDP+ LOAD MEDIA. WHEN LOADED UNDER XXDP+, THE DIAGNOSTIC SUPERVISOR WILL BE LOADED AUTOMATICALLY.

6.1.2 STARTING PROCEDURES

THE PROGRAM STARTS AT LOCATION 200. USE STANDARD DEC PROCEDURES TO START THE PROGRAM.

6.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION

THE DIAGNOSTIC CAN BE EXECUTED STANDALONE UNDER XXDP+ WITHOUT READING THE REMAINDER OF THIS DOCUMENT, AS FOLLOWS:

A) LOAD AND START DIAGNOSTIC USING RUN COMMAND
B) RECEIVE DIAGNOSTIC SUPERVISOR PROMPT (DR>)
C) ENTER STA<CR>
D) ANSWER HARDWARE AND SOFTWARE QUESTIONS
E) GET END OF PASS MESSAGES OR ERROR MESSAGES
F) TO END EXECUTION, ENTER CONTROL/C

6.2 INITIAL DIALOGUE

AFTER THE PROGRAM AND THE SUPERVISOR ARE LOADED AND THE PROGRAM IS STARTED , THE FOLLOWING IDENTIFICATION IS TYPED:

```
DRS LOADED
DIAG.  RUN-TIME SERVICES
CZDMQ-D-0
M8207 DIAG.  #2 OF 2
UNIT IS M8200,M8204,OR M8207
DR>
```

THE OPERATOR THEN PROCEEDS BY TYPING ONE OR MORE OF THE COMMANDS DESCRIBED IN THE FOLLOWING SECTION 6.3.(FOR MORE DETAILED INFORMATION, REFER TO THE DIAGNOSTIC SUPERVISOR FUNCTIONAL SPECIFICATION).

6.3 PROGRAM OPTIONS

6.3.1 START COMMAND

```
*****************************************************************
STA(RT)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
   <FLAG-LIST>/EOP:<INCR>
*****************************************************************
```

6.3.1.1 TESTS SWITCH (/TESTS:<TEST-LIST>)

<TEST-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (1:2 ETC.) OR RANGES OF DECIMAL NUMBERS (1-5:8-10 ETC.) THAT SPECIFY THE TESTS TO BE EXECUTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS RANGE FROM 1 TO THE LARGEST TEST NUMBER IN THE DIAGNOSTIC. THEY MAY BE SPECIFIED IN ANY ORDER. TESTS WILL BE EXECUTED IN NUMERICAL ORDER REGARDLESS OF THE ORDER OF SPECIFICATION. THE DEFAULT IS TO EXECUTE ALL TESTS. ON THIS AND ALL SWITCHES, THE ANGLE BRACKETS <> ARE PUNCTUATION USED IN THE DEFINITION ONLY, AND ARE NOT TO BE TYPED BY THE OPERATOR. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.2 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS A DECIMAL NUMBER INDICATING THE DESIRED NUMBER OF PASSES. A PASS IS DEFINED AS THE EXECUTION OF THE FULL DIAGNOSTIC (ALL SELECTED TESTS) AGAINST ALL UNITS SUBMITTED. THE DEFAULT IS NON-ENDING EXECUTION. IN THIS CASE EXIT FROM THE PROGRAM IS ACCOMPLISHED EITHER BY TYPING A CONTROL/C OR BY OCCURANCE OF AN ERROR WITH THE HALT ON ERROR FLAG BEING SET. THE EXIT IS A RETURN TO COMMAND MODE. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.3 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS A SEQUENCE OF ELEMENTS OF THE FORM <FLAG>, <FLAG=1>, OR <FLAG=0>, SEPARATED BY COLONS, WHERE <FLAG> HAS ONE OF THE FOLLOWING VALUES:

HOE	HALT ON ERROR, CAUSING COMMAND MODE TO BE ENTERED WHEN AN ERROR IS ENCOUNTERED
LOE	LOOP ON ERROR, CAUSING THE DIAGNOSTIC TO LOOP CONTINUOUSLY WITHIN THE SMALLEST DEFINED BLOCK OF CODING (SEGMENT, SUBTEST, OR TEST) CONTAINING THE ERROR
IER	INHIBIT ERROR REPORTING
IBE	INHIBIT BASIC ERROR REPORTS
IXE	INHIBIT EXTENDED ERROR REPORTS
PRI	DIRECT ALL MESSAGES TO A LINE PRINTER
PNT	PRINT NUMBER OF TEST BEING EXECUTED
BOE	BELL ON ERROR
UAM	RUN IN UNATTENDED MODE, BYPASSING MANUAL INTERVENTION TESTS

```
   ISR   INHIBIT STATISTICAL REPORTS
   IDU   INHIBIT DROPPING OF UNITS BY DIAGNOSTIC
   LOT   LOOP ON TEST
```

THE FLAGS NAMED OR EQUATED TO 1 ARE SET, THOSE EQUATED TO 0 ARE CLEARED. A FLAG NOT SPECIFIED IS CLEARED. IF THE FLAGS SWITCH IS NOT GIVEN ALL FLAGS ARE CLEARED. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.4 END OF PASS SWITCH (/EOP:<INCR>)

<INCR> IS A DECIMAL NUMBER INDICATING HOW OFTEN (IN TERMS OF PASSES) IT IS DESIRED THAT THE END OF PASS MESSAGE BE PRINTED. THE DEFAULT IS AT THE END OF EVERY PASS. SEE EXAMPLE AT END OF 6.3.1.5.

6.3.1.5 EFFECT OF START COMMAND

THE EFFECT OF THE START COMMAND IS TO INITIATE THE HARDWARE PARAMETER DIALOGUE, THE SOFTWARE PARAMETER DIALOGUE, AND THEN THE DIAGNOSTIC TESTS THEMSELVES.

THE HARDWARE PARAMETER DIALOGUE COMMENCES WITH THE QUESTION "# UNITS?" TO WHICH THE OPERATOR REPLIES WITH A DECIMAL NUMBER N FROM 1 TO 16. THE TERM 'UNIT" REFERS TO THE DEVICE TO WHICH THIS SERIES OF DIAGNOSTICS IS DEDICATED. FOLLOWING THIS ARE THE QUESTIONS WHEREBY THE P-TABLES THEMSELVES WILL BE BUILT. EACH P-TABLE IS A CORE-RESIDENT TABLE CONTAINING ALL THE HARDWARE INFORMATION FOR ONE UNIT. THE OPERATOR MUST SUPPLY N (NUMBER OF UNITS) VALUES FOR EACH QUESTION. HE MAY DO THIS BY GIVING ONE ANSWER TO EACH QUESTION (IN WHICH CASE THE SERIES OF QUESTIONS WILL BE POSED N TIMES) OR BY GIVING N VALUES, SEPARATED BY COMMAS, TO EACH QUESTION (SERIES WILL BE POSED ONCE). EACH QUESTION IS FOLLOWED BY THE RESPONSE RADIX (D FOR DECIMAL, B FOR BINARY, O FOR OCTAL, L FOR YES/NO) IN PARENTHESES AND THE DEFAULT VALUE AFTER THE PARENTHESES.

FOLLOWING THE HARDWARE QUESTIONS ARE THE SOFTWARE QUESTIONS TO BUILD THE SOFTWARE TABLES, WHICH DEFINE THE MODE (QUICK VERIFY ETC.) THAT THE DIAGNOSTIC WILL EXECUTE IN.

WHEN THE QUESTION "# UNITS?" IS ANSWERED, MEMORY STORAGE IS ALLOCATED FOR THE P-TABLES, AND IF THERE IS NOT ENOUGH TO ACCOMMODATE THEM THE MESSAGE "TOO MANY UNITS" IS ISSUED. IN THIS CASE THE DIAGNOSTIC MUST BE EXECUTED MORE THAN ONCE TO TEST ALL UNITS.

EXAMPLE:

```
STA/TESTS:1:2-4:6:8-10/PASS:3/FLAGS:IER:HOE=1:UAM:LOE
```

THIS COMMAND WILL CAUSE THREE PASSES TO BE MADE, EACH PASS CONSISTING OF TESTS 1,2,3,4,6,8,9, AND 10 EXECUTED AGAINST

ALL UNITS. THERE IS NO DIFFERENCE BETWEEN SAYING <FLAG> AND SAYING <FLAG=1>. THE NOTATION <FLAG=0> IS MEANINGFUL ONLY ON A COMMAND OTHER THAN START TO CLEAR A FLAG THAT WAS PREVIOUSLY SET. NOTE THAT ON ALL COMMANDS ONLY THE FIRST THREE LETTERS ARE SCANNED.

6.3.2 RESTART COMMAND

```
**********************************************************************
RES(TART)/TESTS:<TEST-LIST>/PASS:<PASS-CNT>/FLAGS:
   <FLAG-LIST>/UNITS:<UNIT-LIST>
**********************************************************************
```

6.3.2.1 TESTS, PASS, AND FLAGS SWITCHES

<TEST-LIST>, <PASS-CNT>, AND <FLAG-LIST> ARE AS IN THE START COMMAND.

6.3.2.2 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS A SEQUENCE OF DECIMAL NUMBERS (0,1 ETC.) OR RANGES OF DECIMAL NUMBERS (0-5, 8-10 ETC.) THAT SPECIFY THE UNITS TO BE TESTED. THE NUMBERS ARE SEPARATED BY COLONS. THE NUMBERS MAY RANGE FROM 0 THRU N-1 (N IS THE NUMBER OF UNITS SPECIFIED IN THE PREVIOUS START COMMAND). THE NUMBER INDICATES THE POSITION OF THE P-TABLE AS THE DATA WAS ENTERED DURING THE HARDWARE DIAGLOGUE. THE UNITS WHICH ARE SELECTED MUST NOT HAVE BEEN DROPPED BY THE DROP COMMAND. SEE THE DISCUSSION OF ADD AND DROP COMMANDS BELOW. DEFAULT IS TO TEST ALL UNITS WHICH HAVE NOT BEEN DROPPED BY A DROP COMMAND.

6.3.2.3 EFFECT OF RESTART COMMAND

THE RESTART COMMAND DIFFERS FROM THE START COMMAND IN THAT THE P-TABLES FROM THE PREVIOUS START COMMAND (THERE MUST HAVE BEEN ONE) ARE USED, INSTEAD OF NEW ONES BEING BUILT. THE UNITS SWITCH GIVES THE ABILITY TO SELECT A SUBSET OF THESE. THE SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED (OPERATOR WILL BE ASKED). THE COMMAND CAN BE USED AFTER COMMAND MODE HAS BEEN REENTERED IN ANY OF THE THREE NORMAL WAYS: A) THE REQUESTED NUMBER OF PASSES HAVE BEEN MADE B) AN ERROR WAS ENCOUNTERED WITH THE HALT ON ERROR FLAG SET C) A CONTROL/C WAS ENTERED BY THE OPERATOR.

6.3.3 CONTINUE COMMAND

```
**********************************************************************
CON(TINUE)/PASS:<PASS-CNT/FLAGS:<FLAG-LIST>
**********************************************************************
```

425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480

6.3.3.1 PASS SWITCH (/PASS:<PASS-CNT>)

<PASS-CNT> IS SAME AS IN START COMMAND, BUT THE DEFAULT IS THE UNSATISFIED PASS-CNT FROM THE PREVIOUS START OR RESTART. IF NONE REMAINS, THE DEFAULT IS NON-ENDING EXECUTION.

6.3.3.2 FLAG SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS SAME AS IN START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

6.3.3.3 EFFECT OF CONTINUE COMMAND

CONTINUE MUST FOLLOW A START OR RESTART, AND COMMAND MODE MUST HAVE BEEN ENTERED DUE TO A HALT ON ERROR OR A CONTROL/C. THE EFFECT OF THE COMMAND IS TO GO TO THE BEGINNING OF THE TEST THAT WAS BEING EXECUTED WHEN THE HALT OR CONTROL/C TOOK PLACE. SOFTWARE DIALOGUE MAY OPTIONALLY BE REEXECUTED. HARDWARE PARAMETERS MAY NOT BE CHANGED.

6.3.4 PROCEED COMMAND

****
PRO(CEED)/FLAGS:<FLAG-LIST>
****

6.3.4.1 FLAGS SWITCH (/FLAGS:<FLAG-LIST>)

<FLAG-LIST> IS AS IN THE START COMMAND, BUT UNSPECIFIED FLAGS RETAIN THEIR CURRENT VALUE.

6.3.4.2 EFFECT OF PROCEED COMMAND

PROCEED MUST FOLLOW A START, RESTART, OR CONTINUE. COMMAND MODE MUST HAVE BEEN ENTERED VIA A HALT ON ERROR. THE EFFECT OF THE COMMAND IS TO BEGIN EXECUTION AT THE LOCATION FOLLOWING THE ERROR CALL. NEITHER HARDWARE NOR SOFTWARE PARAMETERS MAY BE ALTERED.

6.3.5 ADD COMMAND

****
ADD/UNITS:<UNIT-LIST>
****

6.3.5.1 UNITS SWITCH (/UNITS:<UNIT-LIST>

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.5.2 EFFECT OF ADD COMMAND

THE UNITS SPECIFIED ARE ADDED TO THE TEST SEQUENCE. EACH UNIT MUST HAVE A P-TABLE IN MEMORY DUE TO AN EARLIER HARDWARE DIALOGUE. THIS COMMAND MUST BE FOLLOWED BY A RESTART OR CONTINUE. THE UNITS SWITCH MUST BE SPECIFIED. THE ADD COMMAND IS MEANINGFUL ONLY FOR UNITS THAT WERE PREVIOUSLY DROPPED.

6.3.6 DROP COMMAND

****
DRO(P)/UNITS:<UNIT-LIST>
****

6.3.6.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.6.2 EFFECT OF DROP COMMAND

THE UNITS SPECIFIED WILL BE DROPPED FROM TESTING. THE UNITS WILL BE RESELECTED ONLY BY THE EXECUTION OF AN ADD OR START COMMAND. THE UNITS SWITCH MUST BE ENTERED. THIS COMMAND MUST BE FOLLOWED BY A RESTART OR A CONTINUE COMMAND.

6.3.7 PRINT COMMAND

****
PRI(NT)
****

6.3.7.1 EFFECT OF PRINT COMMAND

THE TOTAL NUMBER OF ERRORS FOR EACH UNIT SINCE THE LAST START OR RESTART COMMAND ARE PRINTED. THE ISR (INHIBIT STATISTICAL REPORTING) FLAG IS CLEARED.

6.3.8 DISPLAY COMMAND

****
DIS(PLAY)/UNITS:<UNIT-LIST>
****

6.3.8.1 UNITS SWITCH (/UNITS:<UNIT-LIST>)

<UNIT-LIST> IS AS IN THE RESTART COMMAND.

6.3.8.2 EFFECT OF DISPLAY COMMAND

THE HARDWARE P-TABLES FOR ALL UNITS UNDER TEST ARE PRINTED OUT IN THE FORMAT IN WHICH THEY WERE ENTERED. ANY UNITS THAT WERE DROPPED BY THE OPERATOR "DROP" COMMAND ARE SO DESIGNATED.

6.3.9 FLAGS COMMAND

****
FLA(GS)
****

6.3.9.1 EFFECT OF FLAGS COMMAND

THE CURRENT SETTINGS OF ALL FLAGS ARE PRINTED.

6.3.10 ZFLAGS COMMAND

****
ZFL(AGS)
****

6.3.10.1 EFFECT OF ZFLAGS COMMAND

ALL FLAGS ARE CLEARED.

6.3.11 CONTROL CHARACTERS

A CONTROL C (C) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES A RETURN TO COMMAND MODE.

A CONTROL Z (Z) ENTERED DURING ONE OF THE THREE OPERATOR DIALOGUES- INITIAL DIALOGUE (SEE 6.2),HARDWARE DIALOGUE (SEE 6.3.1.5), OR SOFTWARE DIALOGUE (SEE 6.3.1.5) CAUSES THE DEFAULTS TO BE TAKEN FOR THE REMAINDER OF THAT DIALOGUE.

A CONTROL O (O) ENTERED DURING THE EXECUTION OF A DIAGNOSTIC CAUSES ALL TELETYPE OUTPUT TO BE SURPRESSED FOR THE REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER O IS TYPED, WHICH RESTORES NORMAL TELETYPE OUTPUT.

6.3.12 HARDWARE PARAMETERS

THE FOLLOWING QUESTIONS WILL BE ASKED ON A START COMMAND. THE VALUE LOCATED TO THE LEFT OF THE QUESTION MARK IS THE DEFAULT VALUE THAT WILL BE TAKEN ON A CARRIAGE RETURN RESPONSE.

1. WHICH MICRO-CPU? (0= M8200, 4= M8204, 7= M8207) (0) 7?

2. MICRO-CPU CSR ADDRESS: (0) 160170?

THIS IS THE ADDRESS AT WHICH THE CSR REGISTERS (SEL0) RESIDE ON THE UNIBUS. THE ALLOWABLE RANGE IS 160000-177776 (OCTAL), AND THE DEFAULT IS 160170.

3. MICRO-PROCESSOR RUN SWITCH-TYPE 1 IF ON, IF OFF: (0) 0?

THE RUN SWITCH IS E28, SWITCH 7 ON THE M8207. MORE TESTS CAN BE PERFORMED IF THE RUN SWITCH IS OFF. YOU MAY GENERATE AN ERROR IF YOU ANSWER THIS QUESTION WRONG.

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETER QUESTIONS ARE ASKED BY PART 2 OF THE STATIC LOGIC TESTS.

6.3.14 EXTENDED DISCUSSION OF P-TABLE DIALOGUE

THE FULL CAPABILITY OF THE HARDWARE DIALOGUE IS REVEALED BY THE FOLLOWING DISCUSSION OF WHAT HAPPENS INTERNALLY.

AS SOON AS THE QUESTION "# UNITS?" IS ANSWERED (WITH THE NUMBER N, SAY) SPACE IN CORE IS ALLOCATED FOR N P-TABLES. ALL OF THE P-TABLES ARE OF THE SAME FORMAT, AND THERE IS A ONE-TO ONE CORRESPONDENCE BETWEEN THE HARDWARE PARAMETER QUESTIONS AND THE SLOTS IN THE P-TABLE FORMAT.

ON THE FIRST TRIP THRU THE QUESTIONS, ALL OF THE SLOTS IN ALL OF THE P-TABLES ARE FILLED. IF THE OPERATOR TYPES IN LESS THAN N EXPLICIT VALUES IN RESPONSE TO A PARTICULAR QUESTION, THESE VALUES ARE PLACED IN THE P-TABLES (ONE VALUE GOING INTO THE PROPER SLOT OF EACH P-TABLE BEGINNING WITH THE FIRST P-TABLE) UNTIL THE STRING OF VALUES IS EXHAUSTED. THE LAST VALUE IN THE STRING BECOMES THE NEW DEFAULT AND IS USED TO FILL THAT SLOT IN THE REMAINING P-TABLES.

ON SUBSEQUENT TRIPS THRU THE QUESTIONS, THE SAME PROCESS IS CARRIED OUT, EXCEPT THAT THE EARLIEST P-TABLE NOT TO HAVE RECEIVED AN EXPLICIT VALUE IN ANY OF ITS SLOTS NOW ASSUMES THE ROLE THAT TABLE NUMBER ONE PLAYED IN THE FIRST TRIP.

THE SERIES OF QUESTIONS IS REISSUED UNTIL AT LEAST ONE QUESTION HAS RECEIVED N EXPLICIT VALUES FROM THE OPERATOR.

IN GIVING A STRING OF VALUES, COMMAS WITHOUT INTERVENING VALUES MAY BE USED TO INDICATE A REPETITION OF THE LAST NAMED VALUE.

A STRING OF VALUES MAY BE GIVEN AS A RANGE (6-10 FOR EXAMPLE). IF THE VALUES REPRESENT PURE NUMERICAL DATA, THIS SAMPLE RANGE TRANSLATES TO THE STRING 6,7,8,9,10 (AN INCREMENT OF 1). IF THE VALUES ARE ADDRESSES, THE SAMPLE RANGE TRANSLATES TO THE STRING 6,8,10 (AN INCREMENT OF 2).

NOW LET US SEE HOW WE COULD USE THESE CAPABILITIES TO CONSTRUCT A SET OF P-TABLES. ASSUME THAT WE HAVE 16 UNITS, AND THAT THERE ARE THREE HARDWARE PARAMETERS FOR EACH (THREE SLOTS IN THE P-TABLE, THREE HARDWARE QUESTIONS IN THE DIALOGUE). LET THE DESIRED VALUE FOR THE FIRST PARAMETER BE THE NUMBER 75 FOR ALL 16 TABLES. LET THE DESIRED VALUE FOR THE SECOND PARAMETER BE EQUAL TO THE UNIT NUMBER (0,1,2,...,15) EXCEPT FOR UNIT 12, WHICH SHOULD RECEIVE THE VALUE 11. LET THE DESIRED VALUE FOR THE THIRD PARAMETER BE THE NUMBER 76 FOR THE FIRST 7 UNITS AND THE NUMBER 77 FOR THE LAST 9 UNITS.

THE FOLLOWING DIALOGUE WOULD ACCOMPLISH THIS GOAL:

```
# UNITS (D) ? 16

UNIT 1
<QUESTION 1> ? 75
<QUESTION 2> ? 0-6
<QUESTION 3> ? 76

UNIT 21
<QUESTION 1> ?
<QUESTION 2> ? 7-11,,13-15
<QUESTION 3> ? 77
```

THE FIRST TIME THE SERIES IS ASKED, SLOT ONE RECEIVES A 75 IN ALL 16 TABLES. SLOT TWO RECEIVES THE VALUES 0,1,2,...,6 IN TABLES 0 THRU 6 AND A CONSTANT 6 IN TABLES 7 THRU 15. SLOT THREE RECEIVES A CONSTANT 76 IN ALL 16 TABLES.

THE SECOND TIME THRU THE SERIES, TABLES 16 THRU THE END ARE GOING TO BE AFFECTED (NOTE THAT THIS PIECE OF INFORMATION IS PRINTED OUT FOR THE THE OPERATOR IN THE FORM 'UNIT XX' AT THE BEGINNING OF EACH SERIES). QUESTION 1 IS RESPONDED TO BY A <CR>, SO SLOT ONE STAYS AT CONSTANT 75 IN TABLES 7 THRU 15, SINCE NO NEW EXLICIT VALUES ARE TYPED IN. SLOT TWO GETS THE VALUES 7,8,9,10,11 IN TABLES 7 THRU 11, AND GETS A 11 IN SLOT 12, AND GETS THE VALUES 13,14,15 IN TABLES 13 THRU 15. SLOT THREE GETS THE VALUE 77 IN TABLES 7 THRU 15.

THE DIALOGUE IS TERMINATED WHEN THE SOFTWARE RECOGNIZES THAT 16 EXPLICIT VALUES HAVE BEEN GIVEN FOR AT LEAST ONE QUESTION (NAMELY QUESTION 2).

7.0 TEST DESCRIPTIONS

********** TEST 1 **********

CZDMQD.P11 12-JAN-82 09:50 PROGRAM DOCUMENT

```
705                     *VERIFY THAT REFERENCING UNIBUS DEVICE REGISTERS
706                     *DOES NOT CAUSE A TIME OUT TRAP
707                     ************************************************
708
709
710                     ********************** TEST 2 **********************
711                     *TEST OF BR RIGHT SHIFT
712                     *VERIFY THAT A DEST OF BR RSH (011) OF A MICRO-INSTRUCTION
713                     *SHIFTS THE RESULTING BR  DATA RIGHT ONCE.
714                     ************************************************
715
716
717                     ********************** TEST 3 **********************
718                     *IOP CRAM WRITE/READ TEST
719                     *FLOAT A 1 THROUGH EACH CRAM LOCATION
720                     ************************************************
721
722
723                     ********************** TEST 4 **********************
724                     *IOP CRAM WRITE/READ TEST
725                     *FLOAT A 0 THROUGH EACH CRAM LOCATION
726                     ************************************************
727
728
729                     ********************** TEST 5 **********************
730                     *IOP CRAM DUAL ADDRESSING TEST
731                     *WRITE EACH ADDRESS INTO ITSELF, READ EACH
732                     *ADDRESS TO VERIFY CORRECT ADDRESSING
733                     ************************************************
734
735
736                     ********************** TEST 6 **********************
737                     *IOP MAIN MEMORY TEST
738                     *FLOAT A 1 THROUGH ALL MAIN MEMORY LOCATIONS
739                     ************************************************
740
741
742                     ********************** TEST 7 **********************
743                     *IOP MAIN MEMORY TEST
744                     *FLOAT A 0 THROUGH ALL MAIN MEMORY LOCATIONS
745                     ************************************************
746
747
748                     ********************** TEST 8 **********************
749                     *IOP MAIN MEMORY DUAL ADDRESSING TEST
750                     *LOAD EACH MEMORY LOCATION WITH ITS OWN ADDRESS
751                     *READ BACK EACH LOCATION TO VERIFY CORRECT ADDRESSING
752                     ************************************************
753
754
755                     ********************** TEST 9 **********************
756                     *IOP MAR TEST
757                     *PERFORM DUAL ADDRESSING TEST
758                     *USING MAR AUTO-INC FEATURE
759                     ************************************************
760
```

```
761
762                     ********** TEST 10 **********
763                     *IOP (CRAM) ODT BITS TEST
764                     *LOAD MAR WITH A 0 INC MAR UNTIL IT OVERFLOWS
765                     *VERIFY THAT IBUS* 10 BITS IS SET ONLY WHEN MAR BIT 8 IS A ONE
766                     *AND THAT IBUS* 10 BIT6 IS SET ON MAR OVERFLOW
767                     ************************************************
768
769
770                     ********** TEST 11 **********
771                     *CRAM TEST OF JUMP(I) NEVER MICRO-PROCESSOR INSTRUCTION.
772                     *PERFORM THE JUMP INSTRUCTION
773                     *VERIFY THE JUMP DID NOT OCCUR BY CLOCKING THE INSTRUCTION
774                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
775                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
776                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
777                     *THE CRAM PC IS CORRECT.  IF THE CRAM PC IN NOT RIGHT,
778                     *THEN PORT4 CONTAINS A 37
779                     ************************************************
780
781
782                     ********** TEST 12 **********
783                     *CRAM TEST OF JUMP(I) ALWAYS MICRO-PROCESSOR INSTRUCTION.
784                     *PERFORM THE JUMP INSTRUCTION
785                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
786                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
787                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
788                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
789                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
790                     *THEN PORT4 WILL CONTAIN A 37
791                     ************************************************
792
793
794                     ********** TEST 13 **********
795                     *CRAM TEST OF JUMP(I) ON C BIT SET MICRO-PROCESSOR INSTRUCTION.
796                     *SET THE C BIT, PERFORM THE JUMP INSTRUCTION.
797                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
798                     *IN THE LOCATION IT IS AT.  THIS INSTRUCITON LOADS THE
799                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
800                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
801                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
802                     *THEN PORT4 WILL CONTAIN A 37.
803                     ************************************************
804
805
806                     ********** TEST 14 **********
807                     *CRAM TEST OF JUMP(I) ON Z BIT SET MICRO-PROCESSOR INSTRUCTION.
808                     *SET THE Z BIT, PERFORM THE JUMP INSTRUCTION.
809                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
810                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
811                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
812                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
813                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
814                     *THEN PORT4 WILL CONTAIN A 37
815                     ************************************************
816
```

```
817
818                     ********** TEST 15 **********
819                     *CRAM TEST OF JUMP(I) ON BRO SET MICRO-PROCESSOR INSTRUCTION.
820                     *SET THE BRO BIT, PERFORM THE JUMP INSTRUCTION.
821                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
822                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
823                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
824                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
825                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
826                     *THEN THE PORT4 WILL CONTAIN A 37
827                     ************************************************
828
829
830                     ********** TEST 16 **********
831                     *CRAM TEST OF JUMP(I) ON BR1 SET MICRO-PROCESSOR INSTRUCTION.
832                     *SET THE BR1 BIT, PERFORM THE JUMP INSTRUCTION.
833                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
834                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
835                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
836                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
837                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
838                     *THEN PORT4 WILL CONTAIN A 37
839                     ************************************************
840
841
842                     ********** TEST 17 **********
843                     *CRAM TEST OF JUMP(I) ON BR4 SET MICRO-PROCESSOR INSTRUCTION.
844                     *SET THE BR4 BIT, PERFORM THE JUMP INSTRUCTION.
845                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
846                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
847                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
848                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
849                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
850                     *THEN PORT4 WILL CONTAIN A 37
851                     ************************************************
852
853
854                     ********** TEST 18 **********
855                     *CRAM TEST OF JUMP(I) ON BR7 SET MICRO-PROCESSOR INSTRUCTION.
856                     *SET THE BR7 BIT, PERFORM THE JUMP INSTRUCTION
857                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
858                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
859                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
860                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
861                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
862                     *THEN PORT4 WILL CONTAIN A 37
863                     ************************************************
864
865
866                     ********** TEST 19 **********
867                     *CRAM TEST OF JUMP(I) ON C BIT CLEAR MICRO-PROCESSOR INSTRUCTION.
868                     *SET THE C BIT, PERFORM THE JUMP INSTRUCTION.
869                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
870                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
871                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
872                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
```

```
873                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
874                     *THEN PORT4 WILL CONTAIN A 37
875                     ************************************************************
876
877
878                     ********************* TEST 20 ***************************
879                     *CRAM TEST OF JUMP(I) ON Z BIT CLEAR MICRO-PROCESSOR INSTRUCTION.
880                     *CLEAR THE Z BIT, PERFORM THE JUMP INSTRUCTION.
881                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
882                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
883                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
884                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
885                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
886                     *THEN PORT4 WILL CONTAIN A 37
887                     ************************************************************
888
889
890                     ********************* TEST 21 ***************************
891                     *CRAM TEST OF JUMP(I) ON BRO CLEAR MICRO-PROCESSOR INSTRUCTION.
892                     *CLEAR THE BRO BIT, PERFORM THE JUMP INSTRUCTION.
893                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
894                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
895                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  A THIS POINT
896                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
897                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
898                     *THEN PORT4 WILL CONTAIN A 37
899                     ************************************************************
900
901
902                     ********************* TEST 22 ***************************
903                     *CRAM TEST OF JUMP(I) ON BR1 CLEAR MICRO-PROCESSOR INSTRUCTION.
904                     *CLEAR THE BR1 BIT, PERFORM THE JUMP INSTRUCTION.
905                     *VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
906                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
907                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
908                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
909                     *THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
910                     *THEN PORT4 WILL CONTAIN A 37
911                     ************************************************************
912
913
914                     ********************* TEST 23 ***************************
915                     *CRAM TEST OF JUMP(I) ON BR4 CLEAR MICRO-PROCESSOR INSTRUCTION.
916                     *CLEAR THE BR4 BIT, PERFORM THE JUMP INSTRUCTION.
917                     *VERIFY THE JUMP DID NOT OCCUR BY CLOCKING THE INSTRUCTION
918                     *IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
919                     *BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
920                     *THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
921                     *THE CRAM PC IS CORRECT, IF THE CRAM PC IS NOT RIGHT
922                     *THEN PORT4 CONTAINS A 37
923                     ************************************************************
924
925
926                     ********************* TEST 24 ***************************
927                     *CRAM TEST OF JUMP(I) ON BR7 CLEAR MICRO-PROCESSOR INSTRUCTION.
928                     *CLEAR THE BR7 BIT, PERFORM THE JUMP INSTRUCTION.
```

```
*VERIFY THE JUMP DID NOT OCCUR BY CLOCKING THE INSTRUCTION
*IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
*BR WITH THE LOWEST 8 BITS OF THE CRAM PC.  AT THIS POINT
*THE BR DATA IS MOVED TO PORT4.  IF THIS DATA IS CORRECT
*THE CRAM PC IS CORRECT, IF THE CRAM PC IS NOT RIGHT
*THEN PORT4 CONTAINS A 37
************************************************************


****************** TEST 25 ***************************
*
*MAIN MEMORY PAGE DUAL ADDRESS TEST.
*IN THIS TEST WE WILL VERIFY THAT PAGES DO
*NOT DUAL ADDRESS.  THIS TEST IS DIFFERENT FROM THE
*PREVIOUS DUAL ADDRESS TESTS IN THAT THE OTHER
*TEST REALLY DIDN'T CHECK PAGE DUAL ADDRESSING
************************************************************


****************** TEST 26 ***************************
*
*JUMP FIELD,PAGE TEST
*
*IN THIS TEST WE WILL MAKE SURE A JUMP FIELD INSTRUCTION
*WORKS.  TO DO THIS, WE'LL PUT THE DESIRED PAGE, FIELD
*INFORMATION IN IBUS*<13> THEN ISSUE A JUMP FIELD
*THEN WE'LL READ PC REG. AND VERIFY.
*
************************************************************


****************** TEST 27 ***************************
*
*JUMP TEST, JUMP ALWAYS, JUMP CHANGE FIELD
*
*IN THIS TEST, WE WILL CHECK THE ABILITY OF THE
*MICRO-PROCESSOR TO JUMP (BRANCH AND ALWAYS INSTRUCTION)
*TO LOCATIONS, FIELDS FROM OTHER LOCATIONS FIELDS.
*WE ALREADY KNOW THAT THE BRANCH INSTR WORKS FROM
*OTHER TEST.                PROCEDURE:
*       1.  START ADDR 0, FIELD 0
*       2.  **CALCULATE NEW ADDRS, FIELD VIA INC,
*       3.  CAUSE JUMP (BRANCH) TO NEW ADDRESS
*       4.  READ PC FROM IBUS*12 AND IBUS*13
*       5.  REPEAT STEP 2-4 256.TIMES
*
*       TO CALCULATE NEW ADDRESS:
*       1.  INC LOW BYTE OF ADDRESS FOR PC ADDRESS 0-7
*       2.  INC LOW BYTE OF N ADDRESS FOR PC ADDRESS 8-11
*           BITS REPRESENTED AS BITS 0-3.  WHEN 0-3 OVERFLOWS,
*           RESTARTS AT ZERO.
*                 NET RESULT IS JUMPS FROM:
*           FIELD,PAGE                    LOC
*                    0                    0
*                    1                    1
*                    2                    2
```

```
*                    3                          3
*                   10                          7
*                   11                         11
*                   :TO                         :
*                   17                        377
*****************************************************


******************** TEST 28 ************************
*
*JUMP TEST, JUMP ALWAYS, JUMP CHANGE FIELD
*
*IN THIS TEST, WE WILL CHECK THE ABILITY OF THE
*MICRO-PROCESSOR TO JUMP (BRANCH AND ALWAYS INSTRUCTION)
*TO LOCATIONS, FIELDS FROM OTHER LOCATIONS FIELDS.
*WE ALREADY KNOW THAT THE BRANCH INSTR WORKS FROM
*OTHER TESTS.          PROCEDURE:
*        1.  START ADDR 0, FIELD 0
*        2.  **CALCULATE NEW ADDRS, FIELD VIA DEC.
*        3.  CAUSE JUMP (BRANCH) TO NEW ADDRESS
*        4.  READ PC FROM IBUS*12 AND IBUS*13
*        5.  REPEAT STEP 2-4 256.TIMES
*
*        TO CALCULATE NEW ADDRESS:
*        1.  DEC LOW BYTE OF ADDRESS FOR PC ADDRESS 0-7
*        2.  DEC LOW BYTE OF N ADDRESS FOR PC ADDRESS 8-11
*            BITS REPRESENTED AS BITS 0-3.  WHEN 0-3 OVERFLOWS,
*            RESTARTS AT ZERO
*                NET RESULT IS JUMPS FROM:
*              FIELD,PAGE                   LOC:
*                    0                      0
*                   17                      377
*                   16                      376
*                   15                      375
*                   :TO                     :
*                   00                      000
*****************************************************


******************** TEST 29 ************************
*
*IN THIS TEST WE'LL VERIFY THAT THE Z BIT CAN BE READ FROM
*IBUS*<13>. WE ALREADY KNOW THAT THE Z BIT WORKS PROPERLY,
*ALL WE WANT TO KNOW HERE IS THAT IT CAN BE READ.
*****************************************************



******************** TEST 30 ************************
*
*IN THIS TEST WE'LL VERIFY THAT THE C BIT CAN BE READ FROM
*IBUS*<13>. WE ALREADY KNOW THAT THE C BIT WORKS PROPERLY
*ALL WE WANT TO KNOW HERE IS THAT IT BE READ.
*
*****************************************************
```

```
1041
1042                    ********** TEST 31 **********
1043                    *TEST OF PROGRAM CLOCK BIT
1044                    *DO A MASTER CLEAR, VERIFY THAT PROGRAM CLOCK IS SET
1045                    *WRITE PROGRAM CLOCK BIT TO A ONE, VERIFY THAT IT CLEARS,
1046                    *AND THEN SETS SOME TIME LATER
1047                    ************************************************
1048
1049                    ********** TEST 32 **********
1050                    *FORCE POWER FAIL TEST
1051                    *SET FORCE POWER FAIL BIT VERIFY THAT PROCESSOR TRAPS TO 24
1052                    *GOING DOWN AND COMING UP.  VERIFY ALSO THAT BUS INIT WAS
1053                    *BLOCKED FROM GETTING TO THE M8200,4,7 DURING THE POWER FAIL
1054                    ************************************************
1055
1056
1057                    ********** TEST 33 **********
1058                    *MICRO-PROCESSOR NOISE TEST
1059                    *WRITE ALL ZERO'S THEN ALL ONE'S THEN A DATA PATTERN
1060                    *TO THE IBUS* AND IBUS REGISTERS AND TO THE SP AND MAIN MEM
1061                    *THEN GO BACK AND READ THE DATA PATTERNS TO VERIFY THAT
1062                    *READING AND WRITING OF OTHER LOCATIONS AND REGISTERS
1063                    *DID NOT CHANGE THE DATA.
1064                    ************************************************
1065
1066
1067                    ********** TEST 34 **********
1068                    *THIS TEST IS DESIGNED TO MAKE SURE THAT A NODST INSTRUCTION
1069                    *DOES NOT WRITE INTO PORT B OF THE MULTIPORT RAM.
1070                    *TO DO THIS, WE'LL PUT A 125 INTO INDAT2, THEN WE'LL PUT A
1071                    *125 INTO BOTH SP1 AND BR. LAST WE'LL DO A NODST BR, SUBOC, SP1
1072                    *IF THERE IS A WRITE INTO PORTB, INADT2 WILL CONTAIN A 377
1073                    ************************************************
1074
1075
1076                    ********** TEST 35 **********
1077                    *
1078                    *EXTENDED CRAM TEST FOR M8206.  IN THIS TEST WE WILL LOAD DATA
1079                    *THROUGHOUT THE CRAM (TEST DATA IS JUST 4K OF DIAG. CODE) AND
1080                    *THEN READ IT BACK AND VERIFY THAT IT IS CORRECT
1081                    ************************************************
1082
1083
1084                    ********** TEST 36 **********
1085                    *
1086                    *THIS TEST LOADS MICRO-CODE INTO A M8206 MCPU THEN EXECUTES IT.
1087                    *THE MICRO-CODE IS DESIGNED TO WRITE ALL ONES INTO THE SEL REGS.
1088                    *
1089                    ************************************************
1090
1091
1092                    ********** TEST 37 **********
1093                    *
1094                    *NEGATIVE ADDRESS TEST.
1095                    *       IN THIS TEST, WE'LL MAKE SURE THAT THE M8207
1096
```

```
1097                    *       DOES NOT RESPOND TO AN ADDRESS THAT ISN'T ASSIGNED
1098                    *        TO IT
1099                    *
1100                    ************************************************************
1101
1102
1103                    ******************** TEST 38 ****************************
1104                    *
1105                    *BYTE ADDRESSING TEST
1106                    *       HERE, WE'RE GOING TO MAKE SURE THAT WE CAN
1107                    *        WRITE INTO ONLY A HIGH OR LOW BYTE OF THE MCPU.
1108                    *
1109                    ************************************************************
1110
1111
1112                    ******************** TEST 39 ****************************
1113                    *
1114                    *IN THIS TEST WE'RE GOING TO MAKE SURE THAT THE PC
1115                    *REG COUNTS UP PROPERLY.  THE PC REG SHOULD INCREMENT
1116                    *ONCE AFTER EACH INSTRUCTION.
1117                    *
1118                    ************************************************************
1119
1120
1121                    ******************** TEST 40 ****************************
1122                    *
1123                    *IN THIS TEST WE'LL MAKE SURE THAT 'BRANCH FIELD H'' DOESN'T
1124                    *GET STUCK HIGH.
1125                    *FIRST WE'LL CLEAR THE PC HIGH REG. THEN WE'LL DO A BRANCH INSTR
1126                    *WITH BAB BITS 11+12 SET.  IF PCR  BITS 8+9 SET THEN WE'LL KNOW
1127                    *WE WERE SUCCESSFUL IF PCR BITS 8+9 FAIL TO SET, WE,LL KNOW
1128                    *THAT THE MAX SELECTED THE WRONG INPUT TO BE CLOCKED INTO THE PCR.
1129                    ************************************************************
1130
1131
1132                    ******************** TEST 41 ****************************
1133                    *
1134                    *IN THIS TEST WE'RE GOING TO MAKE SURE THAT ONLY SP0
1135                    *IS SELECTED FOR SOURCE WHEN THE DESTINATION
1136                    *IS THE OUTBUS
1137                    *FIRST WE'LL WRITE EACH SP ADDRS INTO ITSELF THEN WE'LL
1138                    *MOV SP TO OBUS4.  THAT SHOULD SELECT
1139                    *SP ADDRESS 0.  IF ANY OTHER DATA SHOWS UP, WE'LL
1140                    *BLAME IT ON THE SELECTION OF A DIFFERENT SCRATCH PAD.
1141                    ************************************************************
1142
1143
1144                    ******************** TEST 42 ****************************
1145                    *
1146                    *IN THIS TEST WE ARE GOING TO MAKE SURE THAT THE
1147                    *SIGNAL 'MOV INST H'' (AND ITS ASSOC. TRIBS) DOESN'T GET
1148                    *STUCK HIGH.  IN ORDER TO DO THIS WE'LL CLEAR THE PC HIGH REG
1149                    *PUT KNOWN DATA IN THE BREG AND SP1 THEN WE'LL BRANCH
1150                    *WITH CROM BITS 0-3 SET AS WELL AS CROM BIT 9 WITH CROM BITS 8 AND 11 CLEAR.
1151                    *IF 'MOV INST H'' GETS STUCK HIGH, THE PC REG HIGH WILL GET LOADED
1152                    *WITH THE CONTENTS OF THE ALU
```

```
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
```

****

********** TEST 43 **********
*TEST THAT MASTER CLEAR, CLEARS BIT$ IN THE NPR CONTROL REGISTER AND
*MICROPROCESSOR MISCELLANEOUS REGISTER-FIRST WE'LL SET THE
*PRIORITY UP SO THAT WHEN WE SET THE BUS REQUEST BIT THAT IT WON'T BUG US
*THEN WE'LL SET ALL THE BITS IN BOTH REGS EXCEPT THE
*NPR REQUEST. WE'LL LOOK TO SEE THAT ALL GOT SET, NEXT
*WE'LL DO A MASTER CLEAR AND BE SURE THAT THEY ALL CLEAR.
****

8.0 ERROR INFORMATION

8.1 ERROR REPORTING

ERRORS ARE REPORTED BY THE PROGRAM AS THEY OCCUR (IF NOT INHIBITED). THE REPORT CONFORMS TO THE DIAGNOSTIC SUPERVISOR ERROR REPORT FORMAT, AND CONSISTS OF A DESCRIPTION OF THE ERROR, THE TEST NUMBER, SUBTEST NUMBER, PC OF THE ERROR CALL, DEVICE ADDRESS, AND BASIC AND EXTENDED ERROR INFORMATION.

THE FOLLOWING EXAMPLES PROVIDE TYPICAL ERROR REPORTS:

```
CZDMQ DVC FTL ERR 00045 TST 027 SUB 000 PC:022572

MASTER   CLEAR  FAILED  TO  CLEAR  PC  REG,  CONTENTS=000624
CZDMQ DVC FTL ERR 00015 TST 042 SUB 000 PC:027234

UNIT=00, FAILING UNIT ADDRESS=160170
JUMP TEST ERROR
FROM ADDR      TO ADDR       BAD ADDR
000402         000000        000114
```

FOR ALL OTHER ERRORS, THE REPORT MAY BE MORE EXTENSIVE AND REQUIRE ADDITIONAL DATA TO BE REPORTED.

9.0 HISTORY

- MODIFIED AUGUST 1980 FOR THE FOLLOWING REASONS:

  1) CANCELL DEPO CZDMQA1
  2) CANCELL DEPO CZDMQA2
  3) DETECT BAD TIMING ON INTERNAL CLOCK.

CZDMQD.P11 12-JAN-82 09:50 PROGRAM DOCUMENT

```
1209
1210                    -   MODIFIED JULY 1981 TO FIX TEST 43 MAR BITS IN IBUS* 10.
1211
1212                    -   MODIFIED JANUARY 1982
1213
1214                        1) ERRONEOUS DATA WAS NOT CLEARED IN HIGH BYTE OF REGISTER 5
1215                           IN TEST 40.
1216                           FIX:  CHANGE BIC  #374,R5   TO BIC #177774,R5.
1217
1218                    a
```

1219
1220
1221
1222
1223

CZDMQD.P11 12-JAN-82 09:50 PROGRAM DOCUMENT

```
1224
1225            002000                  .TITLE CZDMQD0 M8207 STATIC DIAG #2
1226                                            .=2000
1227
1228
1229
1230
1231
1232                                            .MCALL  SVC
1233    002000                                  SVC                             ; INITIALIZE SUPERVISOR MACROS
1234
1235
1236
1237
1238
1239    002000                                  BGNMOD  CZDMQ
1240
1241
1242            000000                  $LSTIN= 0
1243            000000                  $LSTTAG= 0
1244            000000                  SVCINS= 0       ; LIST INSTRUCTIONS, SHIFTED RIGHT
1245            000000                  SVCTST= 0       ; LIST TEST TAGS, SHIFTED RIGHT
1246            000000                  SVCSUB= 0       ; LIST SUBTEST TAGS, SHIFTED RIGHT
1247            000000                  SVCGBL= 0       ; LIST GLOBAL TAGS, SHIFTED RIGHT
1248            000000                  SVCTAG= 0       ; LIST OTHER TAGS, SHIFTED RIGHT
1249
1250                                    ;       CHANGE THE VALUES OF THE SVC... SYMBOLS TO BE ZERO IF YOU WISH
1251                                    ;       TO ALIGN THE MACRO CALLS AND THEIR EXPANSIONS.  CHANGE THE
1252                                    ;       SYMBOLS TO BE MINUS-ONE TO NOT LIST THE EXPANSIONS.  YOU MAY
1253                                    ;       CHANGE THE SYMBOLS AT ANY POINT IN YOUR PROGRAM.
1254
1255
```

```
                                .SBTTL  PROGRAM HEADER
                                ;++
                                ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                                ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                                ;--

002000                                  POINTER BGNAU,BGNDU


                                        HEADER  CZDMQ,D,0,240.,0
002000                          LSNAME::                ;DIAGNOSTIC NAME
002000  103                             .ASCII /C/
002001  132                             .ASCII /Z/
002002  104                             .ASCII /D/
002003  115                             .ASCII /M/
002004  121                             .ASCII /Q/
002005  000                             .BYTE   0
002006  000                             .BYTE   0
002007  000                             .BYTE   0
002010                          LSREV::                 ;REVISION LEVEL
002010  104                             .ASCII  /D/
002011                          LSDEPO::                ;0
002011  060                             .ASCII  /0/
002012                          LSUNIT::                ;NUMBER OF UNITS
002012  000000                          .WORD   0
002014                          LSTIML::                ;LONGEST TEST TIME
002014  000360                          .WORD   240.
002016                          LSHPCP::                ;POINTER TO H.W. QUES.
002016  027344                          .WORD   LSHARD
002020                          LSSPCP::                ;POINTER TO S.W. QUES.
002020  000000                          .WORD   0
002022                          LSHPTP::                ;PTR. TO DEF. H.W. PTABLE
002022  002262                          .WORD   LSHW
002024                          LSSPTP::                ;PTR. TO S.W. PTABLE
002024  000000                          .WORD   0
002026                          LSLADP::                ;DIAG. END ADDRESS
002026  030144                          .WORD   LSLAST
002030                          LSSTA::                 ;RESERVED FOR APT STATS
002030  000000                          .WORD   0
002032                          LSCO::
002032  000000                          .WORD   0
002034                          LSDTYP::                ;DIAGNOSTIC TYPE
002034  000000                          .WORD   0
002036                          LSAPT::                 ;APT EXPANSION
002036  000000                          .WORD   0
002040                          LSDTP::                 ;PTR. TO DISPATCH TABLE
002040  002132                          .WORD   LSDISPATCH
002042                          LSPRIO::                ;DIAGNOSTIC RUN PRIORITY
002042  000000                          .WORD   0
002044                          LSENVI::                ;FLAGS DESCRIBE HOW IT WAS SETUP
002044  000000                          .WORD   0
002046                          LSEXP1::                ;EXPANSION WORD
002046  000000                          .WORD   0
002050                          LSMREV::                ;SVC REV AND EDIT #
002050  003                             .BYTE   CSREVISION
002051  003                             .BYTE   CSEDIT
```

CZDMQD.P11 12-JAN-82 09:50

PROGRAM HEADER

```
1312  002052
1313  002052  000000          LSEF::          ;DIAG. EVENT FLAGS
1314  002054  000000                  .WORD   0
1315  002056                  .WORD   0
1316  002056  000000  LSSPC::
1317  002060                  .WORD   0
1318  002060  002730  LSDEVP::         ; POINTER TO DEVICE TYPE LIST
1319  002062                  .WORD   LSDVTYP
1320  002062  000000  LSREPP::         ;PTR. TO REPORT CODE
1321  002064                  .WORD   0
1322  002064  000000  LSEXP4::
1323  002066                  .WORD   0
1324  002066  000000  LSEXP5::
1325  002070                  .WORD   0
1326  002070  012144  LSAUT::          ;PTR. TO ADD UNIT CODE
1327  002072                  .WORD   LSAU
1328  002072  012140  LSDUT::          ;PTR. TO DROP UNIT CODE
1329  002074                  .WORD   LSDU
1330  002074  000000  LSLUN::          ;LUN FOR EXERCISERS TO FILL
1331  002076                  .WORD   0
1332  002076  002312  LSDESP::         ;POINTER TO DIAG. DESCRIPTION
1333  002100                  .WORD   LSDESC
1334  002100  104035  LSLOAD::         ;GENERATE SPECIAL AUTOLOAD EMT
1335  002102                  EMT     ESLOAD
1336  002102  000000  LSETP::          ;POINTER TO ERRTBL
1337  002104                  .WORD   0
1338  002104  011340  LSICP::          ;PTR. TO INIT CODE
1339  002106                  .WORD   LSINIT
1340  002106  012134  LSCCP::          ;PTR. TO CLEAN-UP CODE
1341  002110                  .WORD   LSCLEAN
1342  002110  012042  LSACP::          ;PTR. TO AUTO CODE
1343  002112                  .WORD   LSAUTO
1344  002112  002122  LSPRT::          ;PTR. TO PROTECT TABLE
1345  002114                  .WORD   LSPROT
1346  002114  000000  LSTEST::         ;TEST NUMBER
1347  002116                  .WORD   0
1348  002116  000000  LSDLY::          ;DELAY COUNT
1349  002120                  .WORD   0
1350  002120  000000  LSHIME::         ;PTR. TO HIGH MEM
1351                          .WORD   0
1352
1353  002122                  BGNPROT
1354  002122          LSPROT::
1355  002122  177777                  .WORD   -1
1356  002124  177777                  .WORD   -1
1357  002126  177777                  .WORD   -1
1358  002130                  ENDPROT
1359
```

CZDMQD.P11 12-JAN-82 09:50 DISPATCH TABLE

```
1360
1361
1362                    .SBTTL  DISPATCH TABLE
1363
1364                    ;////////////////////////////////////////////////////////////////////////////////
1365                    ;/ THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
1366                    ;/ IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
1367                    ;////////////////////////////////////////////////////////////////////////////////
1368
1369 002130                       DISPATCH 43
1370 002130  000053                .WORD   43
1371 002132                 L$DISPATCH::
1372 002132  012146                .WORD   T1
1373 002134  012256                .WORD   T2
1374 002136  012422                .WORD   T3
1375 002140  012552                .WORD   T4
1376 002142  012712                .WORD   T5
1377 002144  013154                .WORD   T6
1378 002146  013356                .WORD   T7
1379 002150  013570                .WORD   T8
1380 002152  014112                .WORD   T9
1381 002154  014470                .WORD   T10
1382 002156  014736                .WORD   T11
1383 002160  015202                .WORD   T12
1384 002162  015432                .WORD   T13
1385 002164  015676                .WORD   T14
1386 002166  016142                .WORD   T15
1387 002170  016406                .WORD   T16
1388 002172  016652                .WORD   T17
1389 002174  017116                .WORD   T18
1390 002176  017362                .WORD   T19
1391 002200  017642                .WORD   T20
1392 002202  020122                .WORD   T21
1393 002204  020376                .WORD   T22
1394 002206  020654                .WORD   T23
1395 002210  021130                .WORD   T24
1396 002212  021404                .WORD   T25
1397 002214  021576                .WORD   T26
1398 002216  021744                .WORD   T27
1399 002220  022322                .WORD   T28
1400 002222  022560                .WORD   T29
1401 002224  022774                .WORD   T30
1402 002226  023210                .WORD   T31
1403 002230  023452                .WORD   T32
1404 002232  024044                .WORD   T33
1405 002234  025116                .WORD   T34
1406 002236  025216                .WORD   T35
1407 002240  025364                .WORD   T36
1408 002242  026042                .WORD   T37
1409 002244  026202                .WORD   T38
1410 002246  026310                .WORD   T39
1411 002250  026446                .WORD   T40
1412 002252  026544                .WORD   T41
1413 002254  026644                .WORD   T42
1414 002256  026770                .WORD   T43
1415
```

```
1416
1417
1418
```

CZDMQD.P11 12-JAN-82 09:50 DEFAULT HARDWARE P-TABLE

```
                        .SBTTL DEFAULT HARDWARE P-TABLE

                        ;////////////////////////////////////////////////////////////////////////
                        ;// THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                        ;// THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                        ;// IS IDENTICAL TO THE STRUCTURE OF THE RUN-TIME P-TABLE.
                        ;////////////////////////////////////////////////////////////////////////

                        .ENABL  AMA
002260                          BGNHW   DFPTBL
002260  000013                  .WORD   L10001-L$HW/2
002262                  L$HW::
002262                  DFPTBL::
002262  000007                  .WORD   7               ;MICRO-CPU TYPE.
002264  160170                  .WORD   160170  ;M8200,4,7 CRS ADDRESS
002266  000300                  .WORD   300     ;M8200,4,7 VECTOR ADDRESS
002270  005000                  .WORD   5000    ;INTERRUPT PRIORITY LEVEL
002272  000003                  .WORD   3       ;LINE UNIT TYPE
002274  000056                  .WORD   56      ;SWITCH PACK #1 (DDCMP LINE #)
002276  000000                  .WORD   0       ;SWITCH PACK #2 (BM873 BOOT ADDRESS)
002300  000000                  .WORD   0       ;SWITCH PACK #3
002302  000000                  .WORD   0       ;TEST CONNECTOR INSTALLED FLAG
002304  000004                  .WORD   4       ;CONTAINS BAUD RATE 4=56K BAUD DEFAULT
                                                ;0=2.4K , 1=4.8K , 2=9.6K , 3=19.2K , 4=56K
                                                ;5=250K , 6=500K , 7=1 MEG BAUD
002306  000000                  .WORD   0       ;0=RUN SW OFF, 1=SW ON

                                ENDHW
002310                  L10001:
```

```
CZDMQD.P11    12-JAN-82 09:50          SOFTWARE P-TABLE

                                       .SBTTL  SOFTWARE P-TABLE

                                       ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                       ;+ THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
                                       ;+ PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
                                       ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

002310                                         BGNSW   SFPTBL
002310  000000                               .WORD   L10002-L$SW/2
002312                                 L$SW::
002312                                 SFPTBL::

                                               ENDSW
002312                                 L10002:
```

```
                                .SBTTL  GLOBAL EQUATES SECTION

                                ;////////////////////////////////////////////////////////////////////////////
                                ;/      THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
                                ;/      ARE USED IN MORE THAN ONE TEST.
                                ;////////////////////////////////////////////////////////////////////////////

002312                                  EQUALS
                                ;
                                ; BIT DIFINITIONS
                                ;
        100000                  BIT15== 100000
        040000                  BIT14== 40000
        020000                  BIT13== 20000
        010000                  BIT12== 10000
        004000                  BIT11== 4000
        002000                  BIT10== 2000
        001000                  BIT09== 1000
        000400                  BIT08== 400
        000200                  BIT07== 200
        000100                  BIT06== 100
        000040                  BIT05== 40
        000020                  BIT04== 20
        000010                  BIT03== 10
        000004                  BIT02== 4
        000002                  BIT01== 2
        000001                  BIT00== 1
                                ;
        001000                  BIT9==  BIT09
        000400                  BIT8==  BIT08
        000200                  BIT7==  BIT07
        000100                  BIT6==  BIT06
        000040                  BIT5==  BIT05
        000020                  BIT4==  BIT04
        000010                  BIT3==  BIT03
        000004                  BIT2==  BIT02
        000002                  BIT1==  BIT01
        000001                  BIT0==  BIT00
                                ;
                                ; EVENT FLAG DEFINITIONS
                                ;   EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                ;
        000040                  EF.START==      32.                     ; START COMMAND WAS ISSUED
        000037                  EF.RESTART==    31.                     ; RESTART COMMAND WAS ISSUED
```

```
CZDMQD.P11    12-JAN-82 09:50        GLOBAL EQUATES SECTION

  1525          000036               EF.CONTINUE==   30.                ; CONTINUE COMMAND WAS ISSUED
  1526          000035               EF.NEW==        29.                ; A NEW PASS HAS BEEN STARTED
  1527          000034               EF.PWR==        28.                ; A POWER-FAIL/POWER-UP OCCURRED
  1528                               ;
  1529                               ;
  1530                               ; PRIORITY LEVEL DEFINITIONS
  1531                               ;
  1532          000340               PRI07== 340
  1533          000300               PRI06== 300
  1534          000240               PRI05== 240
  1535          000200               PRI04== 200
  1536          000140               PRI03== 140
  1537          000100               PRI02== 100
  1538          000040               PRI01== 40
  1539          000000               PRI00== 0
  1540                               ;
  1541                               ;OPERATOR FLAG BITS
  1542                               ;
  1543          000004               EVL==       4
  1544          000010               LOT==      10
  1545          000020               ADR==      20
  1546          000040               IDU==      40
  1547          000100               ISR==     100
  1548          000200               UAM==     200
  1549          000400               BOE==     400
  1550          001000               PNT==    1000
  1551          002000               PRI==    2000
  1552          004000               IXE==    4000
  1553          010000               IBE==   10000
  1554          020000               IER==   20000
  1555          040000               LOE==   40000
  1556          100000               HOE==  100000
  1557
  1558
  1559
  1560
  1561                               ;;*****************************************************************************
  1562                               ;* PROGRAM EVENT FLAG DEFINITIONS
  1563                               ;;*****************************************************************************
  1564
  1565
  1566
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL DATA SECTION

```
                        .SBTTL  GLOBAL DATA SECTION

                        ;////////////////////////////////////////////////////////////////////////////////
                        ;/      THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
                        ;/      IN MORE THAN ONE TEST.
                        ;////////////////////////////////////////////////////////////////////////////////

                        ;;*****************************************************************************
                        ;* STORAGE FOR DEVICE REGISTERS
                        ;;*****************************************************************************
                                DESCRIPT        <M8207 DIAG. #2 OF 2>
002312                  L$DESC::
002312 034115 030062 020067             .ASCIZ  /M8207 DIAG. #2 OF 2/
002320 044504 043501 020056
002326 031043 047440 020106
002334 000062
                                .EVEN

                        ;*****************************************************************************
                        ;* PROGRAM CONTROL PARAMETERS
                        ;*****************************************************************************
002336 000000           NEXT:   .WORD   0               ;ADDRESS OF NEXT TEST TO BE EXECUTED
002340 000000           LOCK:   .WORD   0               ;ADDRESS FOR LOCK CURRENT DATA

                        ;*****************************************************************************
                        ;* MISCELLANEOUS STORAGE
                        ;*****************************************************************************
002342 000000           LOGDEV: .WORD   0               ;LOGICAL DEVICE NUMBER
002344 000000           PSTACK: .WORD   0               ;BASE LEVEL PROGRAM STACK POINTER
002346 000000           SUBRPC: .WORD   0               ;PC OF SUBR CALL FOR ERROR REPORTS
002350 000000           ERRFLG: .WORD   0               ;SUBROUTINE ERROR FLAG
002352 000000           RETADR: .WORD   0               ;SUBR ERROR RETURN ADDRESS
002354 000000           STRTSW: .WORD   0               ;SWITCHES AT START OF PROGRAM
002356 000000           STAT:   .WORD   0               ;KM STATUS WORD STORAGE
002360 000000           CLKX:   .WORD   0               ;
002362 000000           MASKX:  .WORD   0               ;
002364 000000           SAVSP:  .WORD   0               ;STACK POINTER STORAGE
002366 000000           SAVPC:  .WORD   0               ;PROGRAM COUNTER STORAGE
002370 000000           ZERO:   .WORD   0               ;
002372 000001           ONE:    .WORD   1               ;
002374 000000           MEMLIM: .WORD   0               ;HIGHEST LOCATION FOR NPR'S
002376 000001           KMACTV: .BLKW   1               ;M8200,4,7 SELECTED ACTIVE
002400 000001           KMNUM:  .BLKW   1               ;OCTAL NUMBER OF M8200,4,7'S
002402 000001           SAVACT: .BLKW   1               ;ORIGINAL ACTIVE DEVICES
002404 000001           SAVNUM: .BLKW   1               ;WORKABLE NUMBER
002406 000000           FLAG:   .WORD   0               ;SCRATCH STORAGE
002410 000000           RUN:    .WORD   0               ;POINTER TO RUNNING DEVICES
002412 000000           FADR:   .WORD   0
002414 000000           WTYPE:  .WORD   0               ;M82XX NUMBER FOR TYPE OF MICO-CPU
002416 000000           $REG5:  .WORD   0               ;STORAGE USED FOR ERROR MSG DATA
002420 000000           $REG4:  .WORD   0
002422 000000           $REG3:  .WORD   0
002424 000000           $REG2:  .WORD   0
002426 000000           $REG1:  .WORD   0
002430 000000           $REG0:  .WORD   0
002432 000000           TYPE:   .WORD   0               ;=0 FOR DMP,=1 FOR M8206
```

```
002434  000000    MR0:     .WORD   0                      ;MEMLOC USED INSTEAD OF R0.
002436  003777    MEMSZ:   .WORD   3777                   ;INDICATES MEMORIE SIZE, LAST ADDR.
002440  000000    TEMP:    .WORD   0
002442  000000    $TEMP0:  .WORD   0
002444  000000    $TMP0:   .WORD   0
002446  000000    $GDADR:  .WORD   0                      ;CONTAINS ADDRESS OF 'GOOD' DATA
002450  000000    $BDADR:  .WORD   0                      ;CONTAINS ADDRESS OF 'BAD' DATA
002452  000000    $GDDAT:  .WORD   0                      ;CONTAINS 'GOOD' DATA
002454  000000    $BDDAT:  .WORD   0                      ;CONTAINS 'BAD' DATA
002456  000000             .WORD   0                      ;RESERVED--NOT TO BE USED
002460  000000             .WORD   0
002462  000000    FTIME:   .WORD   0
002464  000000    SAVE4:   .WORD   0
002466  000000    SAVE6:   .WORD   0
002470  000000    RUNB:    .WORD   0                      ;0= RUN OFF, 1= RUN SW ON
002472  000000    RUNINH:  .WORD   0                      ;0=RUN SW OFF, 1=RUN SW ON

                  ;*************************************************************************
                  ;* PROGRAM CONTROL FLAGS
                  ;*************************************************************************
002474     000    INIFLG:  .BYTE   0                      ;PROGRAM INITIALIZING FLAG
        002476             .EVEN
002476     000    LOKFLG:  .BYTE   0                      ;LOCK ON CURRENT TEST FLAG
002477     000    QV.FLG:  .BYTE   0                      ;QUICK VERIFY FLAG
                           .EVEN

                  ;*************************************************************************
                  ;* DEFINITION OF M8200,4,7 STATUS WORDS - STAT1,STAT2,STAT3
                  ;*
                  ;*
                  ;*      STAT1 - BITS 00-08 IS M8200,4,7 VECTOR ADDRESS
                  ;*              BIT15=1 LINE UNIT IS AN M8203
                  ;*              BIT14=0 NO TEST CONNECTOR(S) USED
                  ;*              BIT14=1 H-XXX TEST CONNECTOR WILL BE USED
                  ;*              BIT13=0 LINE UNIT IS AN M8201
                  ;*              BIT13=1 LINE UNIT IS AN M8202
                  ;*              BIT12=1 NO LINE UNIT
                  ;*              BITS 09-11 IS M8200,4,7 PRIORITY LEVEL
                  ;*
                  ;*      STAT2 - LOW BYTE IS SWITCH PACK #1 (DDCMP LINE NUMBER)
                  ;*              HIGH BYTE IS SWITCH PACK #2 (BM873 BOOT ADDRESS)
                  ;*
                  ;*      STAT3 - BIT0=1 DO FREE RUNNING TESTS ON M8200,4,7
                  ;*
                  ;*************************************************************************
002500  000000    STAT1:   .WORD   0
002502  000000    STAT2:   .WORD   0
002504  000000    STAT3:   .WORD   0

                  ;*************************************************************************
                  ;* POINTERS TO M8200,4,7 VECTORS AND REGISTERS
                  ;*************************************************************************
002506  000000    KMRVEC: 0        ;POINTER TO M8200,4,7 RCV INTRPT VECTOR
002510  000000    KMRLVL: 0        ;POINTER TO M8200,4,7 RCV INTRPT SERVICE PS
002512  000000    KMTVEC: 0        ;POINTER TO M8200,4,7 TX INTRPT VECTOR
002514  000000    KMTLVL: 0        ;POINTER TO M8200,4,7 TX INTRPT SERVICE PS
```

```
CZDMQD.P11    12-JAN-82 09:50          GLOBAL DATA SECTION

 1679  002516  000000          KMCSR:  0       ;POINTER TO M8200,4,7 CONTROL STATUS REGISTER
 1680  002520  000000          KMCSRH: 0       ;POINTER TO M8200,4,7 CONTROL STATUS REGISTER HIGH BYTE
 1681  002522  000000          KMCTL:  0       ;POINTER TO M8200,4,7 CONTROL OUT REGISTER
 1682  002524  000000          KMP04:  0       ;POINTER TO M8200,4,7 PORT REGISTER - SEL4
 1683  002526  000000          KMP06:  0       ;POINTER TO M8200,4,7 PORT REGISTER - SEL6
 1684
 1685                          ;;****  PRIMARY REG ADRS STORAGE FOR THIS UNIT  ******
 1686                          ;THESE LOCATIONS WILL BE LOADED FOR THE CURRENT UNIT, IN INIT CODE
 1687  002530                  REGADR:
 1688
 1689                          ;;****  STACK USED FOR SUBROUTINE LINKAGE  *****
 1690  002530  000100                  .BLKW   100
 1691  002730                  SSTACK:
 1692
 1693
 1694
 1695
 1696
 1697
 1698
```

```
                                        .SBTTL  GLOBAL TEXT SECTION

                                        ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                                        ;%      THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
                                        ;%      MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
                                        ;%      MORE THAN ONE TEST.
                                        ;%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

                                        ;;******************************************************************
                                        ;* NAMES OF DEVICES SUPPORTED BY PROGRAM
                                        ;;******************************************************************
002730                                          DEVTYP  <M8200,M8204,OR M8207>
002730                                  L$DVTYP::
002730  034115  030062  026060                  .ASCIZ  /M8200,M8204,OR M8207/
002736  034115  030062  026064
002744  051117  046440  031070
002752  033460     000
        002756                                  .EVEN


                                        ;
                                        ; FORMAT STATEMENTS USED IN PRINT CALLS
                                        ;
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL SUBROUTINES

```
1728                    .SBTTL  GLOBAL SUBROUTINES
1729
1730                    ;////////////////////////////////////////////////////////////////////////////
1731                    ;/      THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST
1732                    ;////////////////////////////////////////////////////////////////////////////
1733
1734
1735                    ;-------------------------------------------------------------------------
1736                    ; MACRO'S NEEDED TO CALL SUBROUTINES
1737                    ;-------------------------------------------------------------------------
1738
1739                    .MACRO  POPSP2
1740                            22626
1741                    .ENDM
1742
```

```
1743
1744
1745
1746
1747
```

```
1748
1749                    ;////////////////////////////////////////////////////////////////////////////////
1750                    ;/      THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST
1751                    ;////////////////////////////////////////////////////////////////////////////////
1752
1753
1754
1755                    ;-------------------------------------------------------------------
1756                    ; MACRO'S NEEDED TO CALL SUBROUTINES
1757                    ;-------------------------------------------------------------------
1758                    ;
1759                    .MACRO  K4ONLY  ?N2
1760                            CMP     MEMSZ,#2000
1761                            BNE     N2
1762                            EXIT    TST
1763                            .ENDM
1764                    .MACRO  ED$CALL XY
1765                            .LIST
1766                            ;*************** TEST 'XY' ****************
1767                            .NLIST
1768                            .ENDM
1769                            .MACRO  BADHEAD
1770                            .RADIX  10
1771                            ED$CALL \T$TESTNUM+1
1772                            .RADIX  8
1773                            .ENDM
```

```
1774                    .MACRO  MYINT
1775                            .LIST
1776                            MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
1777                            .NLIST
1778                    .ENDM
1779
1780                    .MACRO  MACEX   ?N2
1781                            .LIST
1782                            ;DO NOT DO TEST IF M8200
1783                            .NLIST
1784                            TST     TYPE
1785                            BNE     N2
1786                            EXIT    TST
1787                    N2:
1788                    .ENDM   MACEX
1789                    .MACRO  MACEX2  ?N2
1790                            .LIST
1791                            ;DO NOT DO TEST IF M8200
1792                            .NLIST
1793                            CMP     WTYPE,#0
1794                            BNE     N2
1795                            EXIT TST
1796                    N2:
1797                    .ENDM   MACEX2
1798                    .MACRO  K4ONLY  ?N2
1799                            .LIST
1800                            ;DO NOT DO TEST IF M8200, OR M8204
1801                            .NLIST
1802                            CMP     MEMSZ,#2000
1803                            BNE     N2
1804                            EXIT    TST
1805                    N2:
1806                                                    ;NOTE THIS TEST IS ONLY DESIGNED FOR 4K MODULE.
1807                    .ENDM
1808
1809                    .MACRO  CLRMAR
1810                            ROMCLK
1811                            004000
1812                    .ENDM   CLRMAR
1813                    .MACRO  ROMCLK
1814                    .LIST
1815                            JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
1816                    .NLIST
1817                    .ENDM
1818
1819                    .MACRO  SROMCLK
1820                            .LIST
1821                            JSR     R5,..SROMCLK
1822                            .NLIST
1823                    .ENDM
1824                    .MACRO  SKIP06  NNN
1825                            .LIST
1826                            ;GOTO 'NNN' IF M8206
1827                            .NLIST
1828                            CMP     WTYPE,#6                ;SEE IF M8206
1829                            BEQ     NNN
```

```
CZDMQD.P11    12-JAN-82 09:50          GLOBAL SUBROUTINES

                                       .ENDM
                                       .MACRO  SKIP07  NNN
                                               .LIST
                                               ;GOTO 'NNN' IF M8207
                                               .NLIST
                                               CMP     WTYPE,#7                ;SEE IF M8200,4,7
                                               BEQ     NNN
                                       .ENDM
                                       .MACRO  SKIP04  NNN
                                               .LIST
                                               ;GOTO 'NNN' IF M8204
                                               .NLIST
                                               CMP     WTYPE,#4                ;SEE IF M8204
                                               BEQ     NNN
                                       .ENDM
                                       .MACRO  MSTCLR
                                       JSR     R5,.MSTCLR              ;CLEAR M8200,4,7
                                       .ENDM

002756                                 .MSTCLR:
002756  112777  000100  177534                 MOVB    #BIT6,@KMCSRH   ;SET INST.
002764  142777  000300  177526                 BICB    #BIT6!BIT7,@KMCSRH
002772  000205                                 RTS     R5

002774  000024                         PATCH:  .BLKW   20.             ;PATCH AREA.


003044                                 ENDBUG:
                                       ;               UNSAFE TO PATCH ANY OTHER AREA.
003044                                 .ROMCLK:
003044  000240                                 NOP
003046  000240                                 NOP
003050  152777  000002  177442       .REGT:  BISB    #BIT1,@KMCSRH
003056  012577  177444                       MOV     (R5)+,@KMP06
003062  152777  000003  177430               BISB    #BIT1!BIT0,@KMCSRH
003070  142777  000007  177422               BICB    #BIT2!BIT1!BIT0,@KMCSRH
003076  000205                               RTS     R5

003100                                 .SROMCLK:
003100  000240                                 NOP
003102  022737  000006  002414               CMP     #6,WTYPE
003110  001357                               BNE     .REGT
003112  152777  000002  177400               BISB    #BIT1,@KMCSRH
003120  012577  177402                       MOV     (R5)+,@KMP06
003124  000240                               NOP
003126  000240                               NOP
003130  142777  000007  177362               BICB    #7,@KMCSRH
003136                                 1$:
003136  152777  000001  177354               BISB    #BIT0,@KMCSRH           ;STEP INSTR.
003144  142777  000007  177346               BICB    #BIT2!BIT1!BIT0,@KMCSRH
003152  000240                               NOP
003154  000240                               NOP
003156  152777  000002  177334               BISB    #2,@KMCSRH
003164                                 2$:
003164  000205                               RTS     R5
```

```
CZDMQD.P11    12-JAN-82 09:50        GLOBAL SUBROUTINES

 1886  003166
 1887
 1888  003166                        CLRALL:
 1889  003166  004537  003044              ;CLEARS C & Z BITS AND BR
 1890  003172  000400                      ROMCLK
 1891  003174                        JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1892  003174  004537  003044        400                       ;0 TO BR
 1893  003200  063220                      ROMCLK
 1894  003202                        JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1895  003202  004537  003044        63220                     ;SP(0) TO BR
 1896  003206  060400                      ROMCLK
 1897  003210                        JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1898  003210  004537  003100        60400                     ;BR,SP(0) + BR
 1899  003214  000000                      SROMCLK
 1900  003216  000207                JSR     R5,..SROMCLK
 1901                                0
 1902  003220                        RTS     PC
 1903
 1904  003220                        SETBR0:
 1905  003220  004537  003044              ;SETS BR0 BIT
 1906  003224  000401                      ROMCLK
 1907  003226  000207                JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1908                                401                       ;1 TO BR
 1909  003230                        RTS     PC
 1910
 1911                                SETBR1:
 1912  003230                              ;THIS SUBROUTINE SETS BR1 BIT
 1913  003230  004537  003044
 1914  003234  000402                ROMCLK                    ;NEXT WORD IS INSTRUCTION
 1915  003236  000207                JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1916                                000402                    ;BR_002
 1917  003240                        RTS     PC
 1918
 1919                                SETBR4:
 1920  003240                              ;THIS SUBROUTINE SETS BR4 BIT
 1921  003240  004537  003044
 1922  003244  000420                ROMCLK                    ;NEXT WORD IS INSTRUCTION
 1923  003246  000207                JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1924                                420
 1925  003250                        RTS     PC
 1926
 1927                                SETBR7:
 1928  003250                              ;THIS SUBROUTINE SETS BR7 BIT
 1929  003250  004537  003044
 1930  003254  000600                ROMCLK                    ;NEXT WORD IS INSTRUCTION
 1931  003256  000207                JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1932                                600
 1933  003260                        RTS     PC
 1934
 1935                                SETC:
 1936  003260                              ;THIS SUBROUTINE SETS THE C BIT
 1937  003260  004537  003044
 1938  003264  000777                ROMCLK                    ;NEXT WORD IS INSTRUCTION
 1939  003266                        JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
 1940  003266  004537  003044        000777                    ;BR_377
 1941  003272  063220                ROMCLK                    ;NEXT WORD IS INSTRUCTION
                                      JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
                                      063220                    ;SP(0)_BR
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL SUBROUTINES

```
1942  003274
1943  003274  004537  003044          ROMCLK                  ;NEXT WORD IS INSTRUCTION
1944  003300  060400                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
1945  003302                          060400                  ;BR SP(0)+BR
1946  003302  004537  003100          SROMCLK                 ;NOW WE MUST CLOCK THE BITS INTO IBUS <13>
1947  003306  000000                  JSR     R5,..SROMCLK
1948  003310  000207                  0
1949                                  RTS     PC
1950  003312                  SETZ:
1951                                  ;THIS SUBROUTINE SETS THE Z BIT
1952
1953  003312                          ROMCLK                  ;NEXT WORD IS INSTRUCTION
1954  003312  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
1955  003316  000777                  000777                  ;BR 377
1956  003320                          SROMCLK                 ;NOW CLOCK THE BITS INTO IBUS<13>
1957  003320  004537  003100          JSR     R5,..SROMCLK
1958  003324  000777                  0777
1959  003326  000207                  RTS     PC
1960
1961  003330                  RAMDAT:
1962                                  ;THIS SUBROUTINE LOADS R4 WITH THE LOWEST
1963                                  ;8 BITS OF THE CRAM PC.
1964
1965  003330  005004                  CLR     R4
1966  003332  017605  000000          MOV     @(SP),R5        ;GOOD DATA
1967  003336  062716  000002          ADD     #2,(SP)         ;ADJUST STACK
1968  003342                          SKIP06  1$      ;IF M8206,WE'LL GET PC A DIFFERENT WAY.
1969                                  ;GOTO 1$ IF M8206
1970  003352                          SKIP07  1$      ;IF M8200,4,7 WE'LL GET PC A DIFFERENT WAY.
1971                                  ;GOTO 1$ IF M8207
1972  003362  005011                  CLR     (R1)            ;CLEAR BIT10
1973  003364  052711  000400          BIS     #BIT8,(R1)      ;CLOCK INSTRUCTION IN CRAM THAT
1974                                                          ;JUMPED TO, IT LOADS BR WITH IT
1975  003370  005011                  CLR     (R1)            ;CLR BIT8
1976  003372                          ROMCLK                  ;NEXT WORD IS INSTRUCTION
1977  003372  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
1978  003376  061225                  061225                  ;MOV BR TO PORT 5
1979  003400  116104  000005          MOVB    5(R1),R4        ;PUT 'FOUND" IN R4
1980  003404  000207                  RTS     PC              ;RETURN
1981
1982  003406                  1$:     ROMCLK                  ;READ PC LOW REG DIRECTLY.
1983  003406  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
1984  003412  121244                  121244                  ;IBUS*<12> TO PORT 4
1985  003414  116104  000004          MOVB    4(R1),R4        ;PUT INTO R4
1986  003420  000207                  RTS PC                  ;EXIT
1987
1988  003422                  WROM:
1989                                  ;THIS SUBROUTINE WRITES THE ROMMAP INTO THE CRAM
1990
1991                            ;     BIT     #BIT15,STAT1    ;BE SURE M8200,4,7 HAS CRAM
1992                            ;     BEQ     2$              ;SKIP IF NO CRAM
1993  003422                          SKIP07  2$
1994                                  ;GOTO 2$ IF M8207
1995  003432  005000                  CLR     R0              ;R0=CRAM ADDRESS
1996  003434  012702  012146          MOV     #ROMMAP,R2      ;R2 POINTS TO ROMMAP
1997  003440  012711  002000  1$:     MOV     #BIT10,(R1)     ;SET ROM0
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL SUBROUTINES

```
1998  003444  010061  000004          MOV     R0,4(R1)          ;LOAD CRAM ADDRESS
1999  003450  012261  000006          MOV     (R2)+,6(R1)       ;LOAD WORD TO BE WRITTEN
2000  003454  052711  020000          BIS     #BIT13,(R1)       ;WRITE IT!
2001  003460  005200                  INC     R0                ;NEXT ADDRESS
2002  003462  023700  002436          CMP     MEMSZ,R0          ;DONE YET?
2003  003466  001364                  BNE     1$                ;BR IF NO
2004  003470  005011                  CLR     (R1)              ;CLEAR SEL0
2005  003472  000207          2$:     RTS     PC                ;RETURN
2006
2007  003474                  MEMSET:
2008                                  ;THIS SUBROUTINE LOADS CRAM WITH SPECIAL INSTRUCTIONS
2009                                  ;FOR THE CRAM JUMP TEST. ALL CRAM LOCATIONS ARE LOADED
2010                                  ;WITH INSTRUCTIONS THAT MOVE A 37 TO THE BR, EXCEPT THE
2011                                  ;FOLLOWING CRAM ADDRESSES: 0,1,4,7,525,1777. THESE LOCATIONS
2012                                  ;CONTAIN INSTRUCTIONS WHICH LOAD THE BR WITH THE LOWEST
2013                                  ;8 BITS OF THAT CRAM ADDRESS.
2014
2015  003474                          SKIP07  3$        ;IF M8200,4,7 CAN'T WRITE CRAM!
2016                                  ;GOTO 3$ IF M8207
2017  003504  005000                  CLR     R0                ;R0 = CRAM ADDRESS
2018  003506  012711  002000  1$:     MOV     #BIT10,(R1)       ;SET ROMO
2019  003512  010061  000004          MOV     R0,4(R1)          ;LOAD CRAM ADDRESS
2020  003516  012761  000437  000006  MOV     #437,6(R1)        ;LOAD INSTRUCTION
2021  003524  052711  020000          BIS     #BIT13,(R1)       ;WRITE INSTRUCTION IN CRAM
2022  003530  005200                  INC     R0                ;NEXT ADDRESS
2023  003532  023700  002436          CMP     MEMSZ,R0          ;DONE YET?
2024  003536  001363                  BNE     1$                ;BR IF NO
2025  003540  005000                  CLR     R0                ;INDEX REGISTER
2026  003542  012711  002000  2$:     MOV     #BIT10,(R1)       ;SET ROMO
2027  003546  016061  003602  000004  MOV     CRAMA(R0),4(R1)   ;LOAD CRAM ADDRESS IN SEL4
2028  003554  016061  003616  000006  MOV     INSTU(R0),6(R1)   ;LOAD INSTRUCTION TO BE WRITTEN
2029  003562  052711  020000          BIS     #BIT13,(R1)       ;WRITE CRAM!
2030  003566  005720                  TST     (R0)+             ;NEXT
2031  003570  022700  000014          CMP     #14,R0            ;DONE YET?
2032  003574  001362                  BNE     2$                ;BR IF NO
2033  003576  005011                  CLR     (R1)              ;CLEAR ALL BITS
2034  003600  000207          3$:     RTS     PC                ;RETURN
2035
2036  003602  000000  000001  000004  CRAMA:  .WORD   0,1,4,7,1777,525
2037  003610  000007  001777  000525
2038
2039  003616  000400                  INSTU:  000400            ;BR_0
2040  003620  000401                          000401            ;BR_1
2041  003622  000404                          000404            ;BR_4
2042  003624  000407                          000407            ;BR_7
2043  003626  000777                          000777            ;BR_377
2044  003630  000525                          000525            ;BR_125
2045
2046
2047                                  ;ROUTINE TO SAVE GENERAL REGISTERS FOR ERROR ROUTINE.
2048                                  ; CALL = JSR     PC,SV05
2049  003632  010537  002416  SV05:   MOV     R5,$REG5
2050  003636  010437  002420          MOV     R4,$REG4
2051  003642  010337  002422          MOV     R3,$REG3
2052  003646  010237  002424          MOV     R2,$REG2
2053  003652  010137  002426          MOV     R1,$REG1
```

```
2054  003656  013737  002434  002430          MOV     MR0,$REGO
2055  003664  000207                          RTS     PC
2056
2057
```

```
                                .SBTTL  GLOBAL ERROR REPORT SECTION

                                ;//////////////////////////////////////////////////////////////////
                                ;/      THE GLOBAL ERROR REPORT SECTION CONTAINS ERROR MESSAGES
                                ;/      THAT ARE USED IN MORE THAN ONE TEST.
                                ;//////////////////////////////////////////////////////////////////

003666  047045  047445  022466  TFM1:   .ASCIZ  /%N%O6%S4%O6%S4%O4%N/
003674  032123  047445  022466
003702  032123  047445  022464
003710  000116
003712  047045  047445  022463  TFM2:   .ASCIZ  /%N%O3%S7%O3%N/
003720  033523  047445  022463
003726  000116
003730  047045  047445  022463  TFM3:   .ASCIZ  /%N%O3%S10%O3%S4%O4%N/
003736  030523  022460  031517
003744  051445  022464  032117
003752  047045     000
003755     045  022516  031517  TFM4:   .ASCIZ  /%N%O3%S7%O3%N/
003762  051445  022467  031517
003770  047045     000
003773     045  022516  033117  TFM5:   .ASCIZ  /%N%O6%S5%O6%S3%O6%N/
004000  051445  022465  033117
004006  051445  022463  033117
004014  047045     000
004017     045  022516  051101  TFM36:  .ASCIZ  /%N%AREGISTER ADDRESS ERROR,ADDRESS = %O6%A,UNIT = %O2/
004024  043505  051511  042524
004032  020122  042101  051104
004040  051505  020123  051105
004046  047522  026122  042101
004054  051104  051505  020123
004062  020075  047445  022466
004070  026101  047125  052111
004076  036440  022440  031117
004104     000
004105     045  022516  020101  TFM41:  .ASCIZ  /%N%A CSR HIGH BYTE GOT WRITTEN INTO ON A LOW BYTE XFER/
004112  051503  020122  044510
004120  044107  041040  052131
004126  020105  047507  020124
004134  051127  052111  042524
004142  020116  047111  047524
004150  047440  020116  020101
004156  047514  020127  054502
004164  042524  054040  042506
004172  000122
004174  047045  040445  041440  TFM42:  .ASCIZ  /%N%A CSR LOW BYTE GOT WRITTEN INTO ON A HIGH BYTE XFER/
004202  051123  046040  053517
004210  041040  052131  020105
004216  047507  020124  051127
004224  052111  042524  020116
004232  047111  047524  047440
004240  020116  020101  044510
```

```
2114  004246  044107  041040  052131
2115  004254  020105  043130  051105
2116  004262     000
2117  004263     045  022516  047101  TFM40:  .ASCIZ  /%N%ANEG ADDR TEST DUAL ADDR ERROR-BAD ADDR = %06/
2118  004270  043505  040440  042104
2119  004276  020122  042524  052123
2120  004304  042040  040525  020114
2121  004312  042101  051104  042440
2122  004320  051122  051117  041055
2123  004326  042101  040440  042104
2124  004334  020122  020075  047445
2125  004342  000066
2126  004344  040445  051440  051103  TFM43:  .ASCIZ  /%A SCRATCH PAD %03%A DUAL ADDRESS ERROR WITH SP%02/
2127  004352  052101  044103  050040
2128  004360  042101  022440  031517
2129  004366  040445  042040  040525
2130  004374  020114  042101  051104
2131  004402  051505  020123  051105
2132  004410  047522  020122  044527
2133  004416  044124  051440  022520
2134  004424  031117     000
2135  004427     045  022524  052101  TFM44:  .ASCIZ  /%T%ATHE MAR REG, CONTENTS= %06/
2136  004434  042510  046440  051101
2137  004442  051040  043505  020054
2138  004450  047503  052116  047105
2139  004456  051524  020075  047445
2140  004464  000066
2141  004466  052045  040445  044124  TFM45:  .ASCIZ  /%T%ATHE PC REG, CONTENTS= %06/
2142  004474  020105  041520  051040
2143  004502  043505  020054  047503
2144  004510  052116  047105  051524
2145  004516  020075  047445  000066
2146  004524  047045  040445  047516  TFM45A: .ASCII  /%N%ANOTE: THIS ERROR MAY BE FALSELY GENERATED IF THE/
2147  004532  042524  020072  044124
2148  004540  051511  042440  051122
2149  004546  051117  046440  054501
2150  004554  041040  020105  040506
2151  004562  051514  046105  020131
2152  004570  042507  042516  040522
2153  004576  042524  020104  043111
2154  004604  052040  042510
2155  004610  047045  040445  052522          .ASCIZ  /%N%ARUN BIT (SW7 OF E28) IS ON/
2156  004616  020116  044502  020124
2157  004624  051450  033527  047440
2158  004632  020106  031105  024470
2159  004640  044440  020123  047117
2160  004646     000
2161  004647     045  047101  051120  TFM46:  .ASCIZ  '%ANPR/MISC REGS DATA FAILURE, GOOD =%06%A, BAD =%06'
2162  004654  046457  051511  020103
2163  004662  042522  051507  042040
2164  004670  052101  020101  040506
2165  004676  046111  051125  026105
2166  004704  043440  047517  020104
2167  004712  022475  033117  040445
2168  004720  020054  040502  020104
2169  004726  022475  033117     000
```

```
2170  004733     045  050101  020103  TFM47:  .ASCIZ  '%APC INCR. INCORRECT: S/B= %06%A ; WAS = %06''
2171  004740  047111  051103  020056
2172  004746  047111  047503  051122
2173  004754  041505  035124  051440
2174  004762  041057  020075  047445
2175  004770  022466  020101  020073
2176  004776  040527  020123  020075
2177  005004  047445  000066
2178  005010  040515  052123  051105  TMMC:   .ASCIZ  /MASTER CLEAR FAILED TO CLEAR /
2179  005016  041440  042514  051101
2180  005024  043040  044501  042514
2181  005032  020104  047524  041440
2182  005040  042514  051101  000040
2183  005046  047045  052045  047045  FM1:    .ASCIZ  /%N%T%N/
2184  005054     000
2185
2186
2187
2188  005055     000                  EM0:    .ASCIZ  //
2189  005056  051103  046501  042040  EM1:    .ASCIZ  /CRAM DATA ERROR/
2190  005064  052101  020101  051105
2191  005072  047522  000122
2192  005076  051103  046501  042040  EM2:    .ASCIZ  /CRAM DUAL ADDRESSING ERROR/
2193  005104  040525  020114  042101
2194  005112  051104  051505  044523
2195  005120  043516  042440  051122
2196  005126  051117     000
2197  005131     112  046525  020120  EM3:    .ASCIZ  /JUMP ERROR/
2198  005136  051105  047522  000122
2199  005144  051103  046501  045040  EM4:    .ASCIZ  /CRAM JUMP TEST FAULT/
2200  005152  046525  020120  042524
2201  005160  052123  043040  052501
2202  005166  052114     000
2203  005171     111  050117  046440  EM5:    .ASCIZ  /IOP MAIN MEMORY TEST/
2204  005176  044501  020116  042515
2205  005204  047515  054522  052040
2206  005212  051505  000124
2207  005216  047511  020120  040515  EM6:    .ASCIZ  /IOP MAR TEST/
2208  005224  020122  042524  052123
2209  005232     000
2210  005233     102  020122  044522  EM7:    .ASCIZ  /BR RIGHT SHIFT ERROR/
2211  005240  044107  020124  044123
2212  005246  043111  020124  051105
2213  005254  047522  000122
2214  005260  040515  020122  052504  EM10:   .ASCIZ  /MAR DUAL ADDRESSING ERROR/
2215  005266  046101  040440  042104
2216  005274  042522  051523  047111
2217  005302  020107  051105  047522
2218  005310  000122
2219  005312  052512  050115  043040  EM11:   .ASCIZ  /JUMP FIELD ERROR/
2220  005320  042511  042114  042440
2221  005326  051122  051117     000
2222  005333     112  046525  020120  EM12:   .ASCIZ  /JUMP TEST ERROR/
2223  005340  042524  052123  042440
2224  005346  051122  051117     000
2225  005353     103  047117  044504  EM16:   .ASCIZ  /CONDITION CODE TESTING,Z & C/
```

```
2226  005360  044524  047117  041440
2227  005366  042117  020105  042524
2228  005374  052123  047111  026107
2229  005402  020132  020046  000103
2230  005410  046103  041517  020113  EMB1:   .ASCIZ  /CLOCK TIME TOO FAST/
2231  005416  044524  042515  052040
2232  005424  047517  043040  051501
2233  005432  000124
2234  005434                          EM35:
2235  005434  047506  041522  020105 EM17:   .ASCIZ  /FORCE POWER FAIL ERROR/
2236  005442  047520  042527  020122
2237  005450  040506  046111  042440
2238  005456  051122  051117     000
2239  005463     111  052502  025123  EM27:   .ASCIZ  "IBUS* WRITE/READ ERROR"
2240  005470  053440  044522  042524
2241  005476  051057  040505  020104
2242  005504  051105  047522  000122
2243
2244  005512  041111  051525  047457  EM29:   .ASCIZ  'IBUS/OBUS WRITE/READ ERROR'
2245  005520  052502  020123  051127
2246  005526  052111  027505  042522
2247  005534  042101  042440  051122
2248  005542  051117     000
2249
2250  005545     120  046507  041440  EMB50:  .ASCIZ  'PGM CLOCK WOULD NOT CLEAR'
2251  005552  047514  045503  053440
2252  005560  052517  042114  047040
2253  005566  052117  041440  042514
2254  005574  051101     000
2255  005577     120  046507  041440  EMB51:  .ASCIZ  'PGM CLOCK WOULD NOT SET'
2256  005604  047514  045503  053440
2257  005612  052517  042114  047040
2258  005620  052117  051440  052105
2259  005626     000
2260  005627     045  022516  025101  STM:    .ASCIZ  '%N%A************************************************************'
2261  005634  025052  025052  025052
2262  005642  025052  025052  025052
2263  005650  025052  025052  025052
2264  005656  025052  025052  025052
2265  005664  025052  025052  025052
2266  005672  025052  025052  025052
2267  005700  025052  025052  025052
2268  005706  025052  025052  025052
2269  005714  025052  025052  025052
2270  005722     000
2271  005723     000                  DH0:    .ASCIZ  //
2272
2273
2274  005724  054105  042520  052103  DH1:    .ASCIZ  /EXPECTED  FOUND  ADDRESS/
2275  005732  042105  020040  047506
2276  005740  047125  020104  040440
2277  005746  042104  042522  051523
2278  005754     000
2279  005755     105  050130  041505  DH2:    .ASCIZ  /EXPECTED  FOUND/
2280  005762  042524  020104  043040
2281  005770  052517  042116     000
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL ERROR REPORT SECTION

```
2282  005775    106  047522  020115  DH3:    .ASCIZ  /FROM ADDR  TO ADDR  BAD ADDR/
2283  006002  042101  051104  020040
2284  006010  047524  040440  042104
2285  006016  020122  041040  042101
2286  006024  040440  042104  000122
2287
2288
2289                                  .EVEN
2290
2291
2292
2293                                  ;-------------------------------------------------------------------
2294                                  ; MACRO'S NEEDED TO REPORT ERRORS
2295                                  ;-------------------------------------------------------------------
2296
2297                                  .MACRO  MDT1
2298                                          PRINTB  #TFM1,$REG2,$REG4,$REG0
2299                                  .ENDM
2300
2301                                  .MACRO  MDT2
2302                                          PRINTB  #TFM1,$REG5,$REG4,$REG2
2303                                  .ENDM
2304
2305                                  .MACRO  MDT3
2306                                          PRINTB  #TFM2,$REG5,$REG4
2307                                  .ENDM
2308
2309                                  .MACRO  MDT4
2310                                          PRINTB  #TFM3,$REG5,$REG4,FLAG
2311                                  .ENDM
2312
2313                                  .MACRO  MDT5
2314                                          PRINTB  #TFM3,$REG5,$REG4,$REG2
2315                                  .ENDM
2316
2317                                  .MACRO  MDT0
2318                                  .ENDM
2319                                  .MACRO  MDT6
2320                                          PRINTB  #TFM4,$REG2,$REG4
2321                                  .ENDM
2322
2323                                  .MACRO  MDT7
2324                                          PRINTB  #TFM4,$REG5,$REG4
```

```
2325                    .ENDM
2326                    .MACRO  MDT8
2327                            PRINTB #TFM5,FADR,$REG5,$REG4
2328                    .ENDM
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL ERROR REPORT SECTION

```
2329
2330                    .MACRO  $MD     ERRNN   ERNB    ERHM    ERFM
2331                            BGNMSG  ERR'ERRNN
```

```
                        PRINTB  #FM1,#EM'ERNB
                        PRINTB  #FM1,#DH'ERHM
                        MDT'ERFM
                        PRINTB  #STM
                        ENDMSG
                .ENDM

                .MACRO  ERROR   ECB
                        JSR     PC,SV05
                        ERRDF   'ECB',EM0,ERR'ECB'
                .ENDM


006032                  SMD     1,1,1,1
006032          ERR1::
006032  012746  005056          MOV     #EM1,-(SP)
006036  012746  005046          MOV     #FM1,-(SP)
006042  012746  000002          MOV     #2,-(SP)
006046  010600                  MOV     SP,R0
006050  104414                  TRAP    C$PNTB
006052  062706  000006          ADD     #6,SP
006056  012746  005724          MOV     #DH1,-(SP)
006062  012746  005046          MOV     #FM1,-(SP)
006066  012746  000002          MOV     #2,-(SP)
006072  010600                  MOV     SP,R0
006074  104414                  TRAP    C$PNTB
006076  062706  000006          ADD     #6,SP
006102  013746  002430          MOV     $REG0,-(SP)
006106  013746  002420          MOV     $REG4,-(SP)
006112  013746  002424          MOV     $REG2,-(SP)
006116  012746  003666          MOV     #TFM1,-(SP)
006122  012746  000004          MOV     #4,-(SP)
006126  010600                  MOV     SP,R0
006130  104414                  TRAP    C$PNTB
006132  062706  000012          ADD     #12,SP
006136  012746  005627          MOV     #STM,-(SP)
006142  012746  000001          MOV     #1,-(SP)
006146  010600                  MOV     SP,R0
006150  104414                  TRAP    C$PNTB
006152  062706  000004          ADD     #4,SP
006156          L10003:
006156  104423                  TRAP    C$MSG
006160                          $MD     2,2,1,1
006160          ERR2::
006160  012746  005076          MOV     #EM2,-(SP)
006164  012746  005046          MOV     #FM1,-(SP)
006170  012746  000002          MOV     #2,-(SP)
006174  010600                  MOV     SP,R0
006176  104414                  TRAP    C$PNTB
006200  062706  000006          ADD     #6,SP
006204  012746  005724          MOV     #DH1,-(SP)
006210  012746  005046          MOV     #FM1,-(SP)
006214  012746  000002          MOV     #2,-(SP)
006220  010600                  MOV     SP,R0
006222  104414                  TRAP    C$PNTB
```

```
 2388  006224  062706  000006            ADD     #6,SP
 2389  006230  013746  002430            MOV     $REG0,-(SP)
 2390  006234  013746  002420            MOV     $REG4,-(SP)
 2391  006240  013746  002424            MOV     $REG2,-(SP)
 2392  006244  012746  003666            MOV     #TFM1,-(SP)
 2393  006250  012746  000004            MOV     #4,-(SP)
 2394  006254  010600                    MOV     SP,R0
 2395  006256  104414                    TRAP    C$PNTB
 2396  006260  062706  000012            ADD     #12,SP
 2397  006264  012746  005627            MOV     #STM,-(SP)
 2398  006270  012746  000001            MOV     #1,-(SP)
 2399  006274  010600                    MOV     SP,R0
 2400  006276  104414                    TRAP    C$PNTB
 2401  006300  062706  000004            ADD     #4,SP
 2402  006304                     L10004:
 2403  006304  104423                    TRAP    C$MSG
 2404  006306                            $MD     3,1,1,2
 2405  006306                     ERR3::
 2406  006306  012746  005056            MOV     #EM1,-(SP)
 2407  006312  012746  005046            MOV     #FM1,-(SP)
 2408  006316  012746  000002            MOV     #2,-(SP)
 2409  006322  010600                    MOV     SP,R0
 2410  006324  104414                    TRAP    C$PNTB
 2411  006326  062706  000006            ADD     #6,SP
 2412  006332  012746  005724            MOV     #DH1,-(SP)
 2413  006336  012746  005046            MOV     #FM1,-(SP)
 2414  006342  012746  000002            MOV     #2,-(SP)
 2415  006346  010600                    MOV     SP,R0
 2416  006350  104414                    TRAP    C$PNTB
 2417  006352  062706  000006            ADD     #6,SP
 2418  006356  013746  002424            MOV     $REG2,-(SP)
 2419  006362  013746  002420            MOV     $REG4,-(SP)
 2420  006366  013746  002416            MOV     $REG5,-(SP)
 2421  006372  012746  003666            MOV     #TFM1,-(SP)
 2422  006376  012746  000004            MOV     #4,-(SP)
 2423  006402  010600                    MOV     SP,R0
 2424  006404  104414                    TRAP    C$PNTB
 2425  006406  062706  000012            ADD     #12,SP
 2426  006412  012746  005627            MOV     #STM,-(SP)
 2427  006416  012746  000001            MOV     #1,-(SP)
 2428  006422  010600                    MOV     SP,R0
 2429  006424  104414                    TRAP    C$PNTB
 2430  006426  062706  000004            ADD     #4,SP
 2431  006432                     L10005:
 2432  006432  104423                    TRAP    C$MSG
 2433  006434                            $MD     4,3,2,3
 2434  006434                     ERR4::
 2435  006434  012746  005131            MOV     #EM3,-(SP)
 2436  006440  012746  005046            MOV     #FM1,-(SP)
 2437  006444  012746  000002            MOV     #2,-(SP)
 2438  006450  010600                    MOV     SP,R0
 2439  006452  104414                    TRAP    C$PNTB
 2440  006454  062706  000006            ADD     #6,SP
 2441  006460  012746  005755            MOV     #DH2,-(SP)
 2442  006464  012746  005046            MOV     #FM1,-(SP)
 2443  006470  012746  000002            MOV     #2,-(SP)
```

```
2444   006474   010600                  MOV     SP,R0
2445   006476   104414                  TRAP    C$PNTB
2446   006500   062706   000006         ADD     #6,SP
2447   006504   013746   002420         MOV     $REG4,-(SP)
2448   006510   013746   002416         MOV     $REG5,-(SP)
2449   006514   012746   003712         MOV     #TFM2,-(SP)
2450   006520   012746   000003         MOV     #3,-(SP)
2451   006524   010600                  MOV     SP,R0
2452   006526   104414                  TRAP    C$PNTB
2453   006530   062706   000010         ADD     #10,SP
2454   006534   012746   005627         MOV     #STM,-(SP)
2455   006540   012746   000001         MOV     #1,-(SP)
2456   006544   010600                  MOV     SP,R0
2457   006546   104414                  TRAP    C$PNTB
2458   006550   062706   000004         ADD     #4,SP
2459   006554                  L10006:
2460   006554   104423                  TRAP    C$MSG
```

```
2461  006556                          SMD     5,4,2,3
2462  006556                  ERR5::
2463  006556  012746  005144          MOV     #EM4,-(SP)
2464  006562  012746  005046          MOV     #FM1,-(SP)
2465  006566  012746  000002          MOV     #2,-(SP)
2466  006572  010600                  MOV     SP,R0
2467  006574  104414                  TRAP    C$PNTB
2468  006576  062706  000006          ADD     #6,SP
2469  006602  012746  005755          MOV     #DH2,-(SP)
2470  006606  012746  005046          MOV     #FM1,-(SP)
2471  006612  012746  000002          MOV     #2,-(SP)
2472  006616  010600                  MOV     SP,R0
2473  006620  104414                  TRAP    C$PNTB
2474  006622  062706  000006          ADD     #6,SP
2475  006626  013746  002420          MOV     $REG4,-(SP)
2476  006632  013746  002416          MOV     $REG5,-(SP)
2477  006636  012746  003712          MOV     #TFM2,-(SP)
2478  006642  012746  000003          MOV     #3,-(SP)
2479  006646  010600                  MOV     SP,R0
2480  006650  104414                  TRAP    C$PNTB
2481  006652  062706  000010          ADD     #10,SP
2482  006656  012746  005627          MOV     #STM,-(SP)
2483  006662  012746  000001          MOV     #1,-(SP)
2484  006666  010600                  MOV     SP,R0
2485  006670  104414                  TRAP    C$PNTB
2486  006672  062706  000004          ADD     #4,SP
2487  006676                  L10007:
2488  006676  104423                  TRAP    C$MSG
```

```
2489  006700
2490  006700
2491  006700  012746  005171          GLOBAL ERROR REPORT SECTION
2492  006704  012746  005046                  $MD     6,5,1,4
2493  006710  012746  000002          ERR6::
2494  006714  010600                          MOV     #EM5,-(SP)
2495  006716  104414                          MOV     #FM1,-(SP)
2496  006720  062706  000006                  MOV     #2,-(SP)
2497  006724  012746  005724                  MOV     SP,R0
2498  006730  012746  005046                  TRAP    C$PNTB
2499  006734  012746  000002                  ADD     #6,SP
2500  006740  010600                          MOV     #DH1,-(SP)
2501  006742  104414                          MOV     #FM1,-(SP)
2502  006744  062706  000006                  MOV     #2,-(SP)
2503  006750  013746  002406                  MOV     SP,R0
2504  006754  013746  002420                  TRAP    C$PNTB
2505  006760  013746  002416                  ADD     #6,SP
2506  006764  012746  003730                  MOV     FLAG,-(SP)
2507  006770  012746  000004                  MOV     $REG4,-(SP)
2508  006774  010600                          MOV     $REG5,-(SP)
2509  006776  104414                          MOV     #TFM3,-(SP)
2510  007000  062706  000012                  MOV     #4,-(SP)
2511  007004  012746  005627                  MOV     SP,R0
2512  007010  012746  000001                  TRAP    C$PNTB
2513  007014  010600                          ADD     #12,SP
2514  007016  104414                          MOV     #STM,-(SP)
2515  007020  062706  000004                  MOV     #1,-(SP)
2516  007024                                  MOV     SP,R0
2517  007024  104423                          TRAP    C$PNTB
2518  007026                                  ADD     #4,SP
2519  007026                          L10010:
2520  007026  012746  005216                  TRAP    C$MSG
2521  007032  012746  005046                  $MD     7,6,1,5
2522  007036  012746  000002          ERR7::
2523  007042  010600                          MOV     #EM6,-(SP)
2524  007044  104414                          MOV     #FM1,-(SP)
2525  007046  062706  000006                  MOV     #2,-(SP)
2526  007052  012746  005724                  MOV     SP,R0
2527  007056  012746  005046                  TRAP    C$PNTB
2528  007062  012746  000002                  ADD     #6,SP
2529  007066  010600                          MOV     #DH1,-(SP)
2530  007070  104414                          MOV     #FM1,-(SP)
2531  007072  062706  000006                  MOV     #2,-(SP)
2532  007076  013746  002424                  MOV     SP,R0
2533  007102  013746  002420                  TRAP    C$PNTB
2534  007106  013746  002416                  ADD     #6,SP
2535  007112  012746  003730                  MOV     $REG2,-(SP)
2536  007116  012746  000004                  MOV     $REG4,-(SP)
2537  007122  010600                          MOV     $REG5,-(SP)
2538  007124  104414                          MOV     #TFM3,-(SP)
2539  007126  062706  000012                  MOV     #4,-(SP)
2540  007132  012746  005627                  MOV     SP,R0
2541  007136  012746  000001                  TRAP    C$PNTB
2542  007142  010600                          ADD     #12,SP
2543  007144  104414                          MOV     #STM,-(SP)
2544  007146  062706  000004                  MOV     #1,-(SP)
                                              MOV     SP,R0
                                              TRAP    C$PNTB
                                              ADD     #4,SP
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL ERROR REPORT SECTION

```
2545  007152
2546  007152  104423
2547  007154
2548  007154
2549  007154  012746  005233
2550  007160  012746  005046
2551  007164  012746  000002
2552  007170  010600
2553  007172  104414
2554  007174  062706  000006
2555  007200  012746  005755
2556  007204  012746  005046
2557  007210  012746  000002
2558  007214  010600
2559  007216  104414
2560  007220  062706  000006
2561  007224  013746  002420
2562  007230  013746  002416
2563  007234  012746  003712
2564  007240  012746  000003
2565  007244  010600
2566  007246  104414
2567  007250  062706  000010
2568  007254  012746  005627
2569  007260  012746  000001
2570  007264  010600
2571  007266  104414
2572  007270  062706  000004
2573  007274
2574  007274  104423
2575  007276
2576  007276
2577  007276  012746  005260
2578  007302  012746  005046
2579  007306  012746  000002
2580  007312  010600
2581  007314  104414
2582  007316  062706  000006
2583  007322  012746  005755
2584  007326  012746  005046
2585  007332  012746  000002
2586  007336  010600
2587  007340  104414
2588  007342  062706  000006
2589  007346  013746  002420
2590  007352  013746  002424
2591  007356  012746  003755
2592  007362  012746  000003
2593  007366  010600
2594  007370  104414
2595  007372  062706  000010
2596  007376  012746  005627
2597  007402  012746  000001
2598  007406  010600
2599  007410  104414
2600  007412  062706  000004
```

```
L10011:
        TRAP    C$MSG
        SMD     10,7,2,3
ERR10::
        MOV     #EM7,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     #DH2,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     $REG4,-(SP)
        MOV     $REG5,-(SP)
        MOV     #TFM2,-(SP)
        MOV     #3,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #10,SP
        MOV     #STM,-(SP)
        MOV     #1,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #4,SP
L10012:
        TRAP    C$MSG
        SMD     11,10,2,6
ERR11::
        MOV     #EM10,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     #DH2,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     $REG4,-(SP)
        MOV     $REG2,-(SP)
        MOV     #TFM4,-(SP)
        MOV     #3,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #10,SP
        MOV     #STM,-(SP)
        MOV     #1,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #4,SP
```

```
2601  007416
2602  007416  104423
2603  007420
2604  007420
2605  007420  012746  005233
2606  007424  012746  005046
2607  007430  012746  000002
2608  007434  010600
2609  007436  104414
2610  007440  062706  000006
2611  007444  012746  005755
2612  007450  012746  005046
2613  007454  012746  000002
2614  007460  010600
2615  007462  104414
2616  007464  062706  000006
2617  007470  013746  002420
2618  007474  013746  002416
2619  007500  012746  003755
2620  007504  012746  000003
2621  007510  010600
2622  007512  104414
2623  007514  062706  000010
2624  007520  012746  005627
2625  007524  012746  000001
2626  007530  010600
2627  007532  104414
2628  007534  062706  000004
2629  007540
2630  007540  104423
2631  007542
2632  007542
2633  007542  012746  005260
2634  007546  012746  005046
2635  007552  012746  000002
2636  007556  010600
2637  007560  104414
2638  007562  062706  000006
2639  007566  012746  005755
2640  007572  012746  005046
2641  007576  012746  000002
2642  007602  010600
2643  007604  104414
2644  007606  062706  000006
2645  007612  013746  002420
2646  007616  013746  002416
2647  007622  012746  003712
2648  007626  012746  000003
2649  007632  010600
2650  007634  104414
2651  007636  062706  000010
2652  007642  012746  005627
2653  007646  012746  000001
2654  007652  010600
2655  007654  104414
2656  007656  062706  000004
```

```
L10013:
        TRAP    C$MSG
        $MD     12,7,2,7
ERR12::
        MOV     #EM7,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     #DH2,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     $REG4,-(SP)
        MOV     $REG5,-(SP)
        MOV     #TFM4,-(SP)
        MOV     #3,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #10,SP
        MOV     #STM,-(SP)
        MOV     #1,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #4,SP
L10014:
        TRAP    C$MSG
        $MD     13,10,2,3
ERR13::
        MOV     #EM10,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     #DH2,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     $REG4,-(SP)
        MOV     $REG5,-(SP)
        MOV     #TFM2,-(SP)
        MOV     #3,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #10,SP
        MOV     #STM,-(SP)
        MOV     #1,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #4,SP
```

```
2657  007662
2658  007662  104423
2659  007664
2660  007664
2661  007664  012746  005312
2662  007670  012746  005046
2663  007674  012746  000002
2664  007700  010600
2665  007702  104414
2666  007704  062706  000006
2667  007710  012746  005755
2668  007714  012746  005046
2669  007720  012746  000002
2670  007724  010600
2671  007726  104414
2672  007730  062706  000006
2673  007734  013746  002420
2674  007740  013746  002424
2675  007744  012746  003755
2676  007750  012746  000003
2677  007754  010600
2678  007756  104414
2679  007760  062706  000010
2680  007764  012746  005627
2681  007770  012746  000001
2682  007774  010600
2683  007776  104414
2684  010000  062706  000004
2685  010004
2686  010004  104423
```

```
L10015:
        TRAP    C$MSG
        $MD     14,11,2,6
ERR14::
        MOV     #EM11,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     #DH2,-(SP)
        MOV     #FM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #6,SP
        MOV     $REG4,-(SP)
        MOV     $REG2,-(SP)
        MOV     #TFM4,-(SP)
        MOV     #3,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #10,SP
        MOV     #STM,-(SP)
        MOV     #1,-(SP)
        MOV     SP,R0
        TRAP    C$PNTB
        ADD     #4,SP
L10016:
        TRAP    C$MSG
```

```
2687  010006
2688  010006
2689  010006  012746  005333          SMD     15,12,3,8
2690  010012  012746  005046  ERR15::
2691  010016  012746  000002          MOV     #EM12,-(SP)
2692  010022  010600                  MOV     #FM1,-(SP)
2693  010024  104414                  MOV     #2,-(SP)
2694  010026  062706  000006          MOV     SP,R0
2695  010032  012746  005775          TRAP    C$PNTB
2696  010036  012746  005046          ADD     #6,SP
2697  010042  012746  000002          MOV     #DH3,-(SP)
2698  010046  010600                  MOV     #FM1,-(SP)
2699  010050  104414                  MOV     #2,-(SP)
2700  010052  062706  000006          MOV     SP,R0
2701  010056  013746  002420          TRAP    C$PNTB
2702  010062  013746  002416          ADD     #6,SP
2703  010066  013746  002412          MOV     $REG4,-(SP)
2704  010072  012746  003773          MOV     $REG5,-(SP)
2705  010076  012746  000004          MOV     FADR,-(SP)
2706  010102  010600                  MOV     #TFM5,-(SP)
2707  010104  104414                  MOV     #4,-(SP)
2708  010106  062706  000012          MOV     SP,R0
2709  010112  012746  005627          TRAP    C$PNTB
2710  010116  012746  000001          ADD     #12,SP
2711  010122  010600                  MOV     #STM,-(SP)
2712  010124  104414                  MOV     #1,-(SP)
2713  010126  062706  000004          MOV     SP,R0
2714  010132  L10017:
2715  010132  104423                  TRAP    C$PNTB
2716  010134                          ADD     #4,SP
2717  010134
2718  010134  012746  005353          TRAP    C$MSG
2719  010140  012746  005046          SMD     16,16,2,7
2720  010144  012746  000002  ERR16::
2721  010150  010600                  MOV     #EM16,-(SP)
2722  010152  104414                  MOV     #FM1,-(SP)
2723  010154  062706  000006          MOV     #2,-(SP)
2724  010160  012746  005755          MOV     SP,R0
2725  010164  012746  005046          TRAP    C$PNTB
2726  010170  012746  000002          ADD     #6,SP
2727  010174  010600                  MOV     #DH2,-(SP)
2728  010176  104414                  MOV     #FM1,-(SP)
2729  010200  062706  000006          MOV     #2,-(SP)
2730  010204  013746  002420          MOV     SP,R0
2731  010210  013746  002416          TRAP    C$PNTB
2732  010214  012746  003755          ADD     #6,SP
2733  010220  012746  000003          MOV     $REG4,-(SP)
2734  010224  010600                  MOV     $REG5,-(SP)
2735  010226  104414                  MOV     #TFM4,-(SP)
2736  010230  062706  000010          MOV     #3,-(SP)
2737  010234  012746  005627          MOV     SP,R0
2738  010240  012746  000001          TRAP    C$PNTB
2739  010244  010600                  ADD     #10,SP
2740  010246  104414                  MOV     #STM,-(SP)
2741  010250  062706  000004          MOV     #1,-(SP)
2742  010254                          MOV     SP,R0
                                      TRAP    C$PNTB
                                      ADD     #4,SP
                                L10020:
```

CZDMQD.P11 12-JAN-82 09:50 GLOBAL ERROR REPORT SECTION

```
2743  010254  104423                    TRAP    C$MSG
2744
```

```
                                                SMD     17,17,0,0
 010256
 010256                                 ERR17::
 010256  012746  005434                         MOV     #EM17,-(SP)
 010262  012746  005046                         MOV     #FM1,-(SP)
 010266  012746  000002                         MOV     #2,-(SP)
 010272  010600                                 MOV     SP,R0
 010274  104414                                 TRAP    C$PNTB
 010276  062706  000006                         ADD     #6,SP
 010302  012746  005723                         MOV     #DH0,-(SP)
 010306  012746  005046                         MOV     #FM1,-(SP)
 010312  012746  000002                         MOV     #2,-(SP)
 010316  010600                                 MOV     SP,R0
 010320  104414                                 TRAP    C$PNTB
 010322  062706  000006                         ADD     #6,SP
 010326  012746  005627                         MOV     #STM,-(SP)
 010332  012746  000001                         MOV     #1,-(SP)
 010336  010600                                 MOV     SP,R0
 010340  104414                                 TRAP    C$PNTB
 010342  062706  000004                         ADD     #4,SP
 010346                                 L10021:
 010346  104423                                 TRAP    C$MSG
 010350                                         SMD     29,29,2,3
 010350                                 ERR29::
 010350  012746  005512                         MOV     #EM29,-(SP)
 010354  012746  005046                         MOV     #FM1,-(SP)
 010360  012746  000002                         MOV     #2,-(SP)
 010364  010600                                 MOV     SP,R0
 010366  104414                                 TRAP    C$PNTB
 010370  062706  000006                         ADD     #6,SP
 010374  012746  005755                         MOV     #DH2,-(SP)
 010400  012746  005046                         MOV     #FM1,-(SP)
 010404  012746  000002                         MOV     #2,-(SP)
 010410  010600                                 MOV     SP,R0
 010412  104414                                 TRAP    C$PNTB
 010414  062706  000006                         ADD     #6,SP
 010420  013746  002420                         MOV     $REG4,-(SP)
 010424  013746  002416                         MOV     $REG5,-(SP)
 010430  012746  003712                         MOV     #TFM2,-(SP)
 010434  012746  000003                         MOV     #3,-(SP)
 010440  010600                                 MOV     SP,R0
 010442  104414                                 TRAP    C$PNTB
 010444  062706  000010                         ADD     #10,SP
 010450  012746  005627                         MOV     #STM,-(SP)
 010454  012746  000001                         MOV     #1,-(SP)
 010460  010600                                 MOV     SP,R0
 010462  104414                                 TRAP    C$PNTB
 010464  062706  000004                         ADD     #4,SP
 010470                                 L10022:
 010470  104423                                 TRAP    C$MSG
 010472                                         SMD     35,35,2,3
 010472                                 ERR35::
 010472  012746  005434                         MOV     #EM35,-(SP)
 010476  012746  005046                         MOV     #FM1,-(SP)
 010502  012746  000002                         MOV     #2,-(SP)
 010506  010600                                 MOV     SP,R0
```

```
2801  010510  104414
2802  010512  062706  000006          TRAP    C$PNTB
2803  010516  012746  005755          ADD     #6,SP
2804  010522  012746  005046          MOV     #DH2,-(SP)
2805  010526  012746  000002          MOV     #FM1,-(SP)
2806  010532  010600                  MOV     #2,-(SP)
2807  010534  104414                  MOV     SP,R0
2808  010536  062706  000006          TRAP    C$PNTB
2809  010542  013746  002420          ADD     #6,SP
2810  010546  013746  002416          MOV     $REG4,-(SP)
2811  010552  012746  003712          MOV     $REG5,-(SP)
2812  010556  012746  000003          MOV     #TFM2,-(SP)
2813  010562  010600                  MOV     #3,-(SP)
2814  010564  104414                  MOV     SP,R0
2815  010566  062706  000010          TRAP    C$PNTB
2816  010572  012746  005627          ADD     #10,SP
2817  010576  012746  000001          MOV     #STM,-(SP)
2818  010602  010600                  MOV     #1,-(SP)
2819  010604  104414                  MOV     SP,R0
2820  010606  062706  000004          TRAP    C$PNTB
2821  010612                          ADD     #4,SP
                                L10023:
2822  010612  104423                  TRAP    C$MSG
2823
2824  010614                  BGNMSG  ERR36
2825  010614                  ERR36::
2826  010614                  PRINTB  #STM
2827  010614  012746  005627          MOV     #STM,-(SP)
2828  010620  012746  000001          MOV     #1,-(SP)
2829  010624  010600                  MOV     SP,R0
2830  010626  104414                  TRAP    C$PNTB
2831  010630  062706  000004          ADD     #4,SP
2832  010634                  ENDMSG
2833  010634                  L10024:
2834  010634  104423                  TRAP    C$MSG
2835
2836  010636                  BGNMSG  ERR40
2837  010636                  ERR40::
2838  010636                  PRINTF  #TFM40,R2
2839  010636  010246                  MOV     R2,-(SP)
2840  010640  012746  004263          MOV     #TFM40,-(SP)
2841  010644  012746  000002          MOV     #2,-(SP)
2842  010650  010600                  MOV     SP,R0
2843  010652  104417                  TRAP    C$PNTF
2844  010654  062706  000006          ADD     #6,SP
2845  010660                  PRINTB  #STM
2846  010660  012746  005627          MOV     #STM,-(SP)
2847  010664  012746  000001          MOV     #1,-(SP)
2848  010670  010600                  MOV     SP,R0
2849  010672  104414                  TRAP    C$PNTB
2850  010674  062706  000004          ADD     #4,SP
2851  010700                  ENDMSG
2852  010700                  L10025:
2853  010700  104423                  TRAP    C$MSG
2854  010702                  BGNMSG  ERR41
2855  010702                  ERR41::
2856  010702                  PRINTF  #TFM41
```

```
2857  010702  012746  004105          MOV     #TFM41,-(SP)
2858  010706  012746  000001          MOV     #1,-(SP)
2859  010712  010600                  MOV     SP,R0
2860  010714  104417                  TRAP    C$PNTF
2861  010716  062706  000004          ADD     #4,SP
2862  010722                  PRINTB  #STM
2863  010722  012746  005627          MOV     #STM,-(SP)
2864  010726  012746  000001          MOV     #1,-(SP)
2865  010732  010600                  MOV     SP,R0
2866  010734  104414                  TRAP    C$PNTB
2867  010736  062706  000004          ADD     #4,SP
2868  010742                  ENDMSG
2869  010742                  L10026:
2870  010742  104423                  TRAP    C$MSG
2871  010744                  BGNMSG  ERR42
2872  010744                  ERR42::
2873  010744                  PRINTF  #TFM42
2874  010744  012746  004174          MOV     #TFM42,-(SP)
2875  010750  012746  000001          MOV     #1,-(SP)
2876  010754  010600                  MOV     SP,R0
2877  010756  104417                  TRAP    C$PNTF
2878  010760  062706  000004          ADD     #4,SP
2879  010764                  PRINTB  #STM
2880  010764  012746  005627          MOV     #STM,-(SP)
2881  010770  012746  000001          MOV     #1,-(SP)
2882  010774  010600                  MOV     SP,R0
2883  010776  104414                  TRAP    C$PNTB
2884  011000  062706  000004          ADD     #4,SP
2885  011004                  ENDMSG
2886  011004                  L10027:
2887  011004  104423                  TRAP    C$MSG
2888
2889  011006                  BGNMSG  ERR43
2890  011006                  ERR43::
2891  011006                  PRINTF  #TFM43,R5,R4
2892  011006  010446                  MOV     R4,-(SP)
2893  011010  010546                  MOV     R5,-(SP)
2894  011012  012746  004344          MOV     #TFM43,-(SP)
2895  011016  012746  000003          MOV     #3,-(SP)
2896  011022  010600                  MOV     SP,R0
2897  011024  104417                  TRAP    C$PNTF
2898  011026  062706  000010          ADD     #10,SP
2899  011032                  PRINTB  #STM
2900  011032  012746  005627          MOV     #STM,-(SP)
2901  011036  012746  000001          MOV     #1,-(SP)
2902  011042  010600                  MOV     SP,R0
2903  011044  104414                  TRAP    C$PNTB
2904  011046  062706  000004          ADD     #4,SP
2905  011052                  ENDMSG
2906  011052                  L10030:
2907  011052  104423                  TRAP    C$MSG
2908  011054                  BGNMSG  ERR44
2909  011054                  ERR44::
2910  011054                  PRINTF  #TFM44,#TMMC,R4
2911  011054  010446                  MOV     R4,-(SP)
2912  011056  012746  005010          MOV     #TMMC,-(SP)
```

```
2913  011062  012746  004427          MOV     #TFM44,-(SP)
2914  011066  012746  000003          MOV     #3,-(SP)
2915  011072  010600                  MOV     SP,R0
2916  011074  104417                  TRAP    C$PNTF
2917  011076  062706  000010          ADD     #10,SP
2918  011102                  PRINTB  #STM
2919  011102  012746  005627          MOV     #STM,-(SP)
2920  011106  012746  000001          MOV     #1,-(SP)
2921  011112  010600                  MOV     SP,R0
2922  011114  104414                  TRAP    C$PNTB
2923  011116  062706  000004          ADD     #4,SP
2924  011122                  ENDMSG
2925  011122                  L10031:
2926  011122  104423                  TRAP    C$MSG
2927  011124                  BGNMSG  ERR45
2928  011124                  ERR45::
2929  011124                  PRINTF  #TFM45,#TMMC,R4
2930  011124  010446                  MOV     R4,-(SP)
2931  011126  012746  005010          MOV     #TMMC,-(SP)
2932  011132  012746  004466          MOV     #TFM45,-(SP)
2933  011136  012746  000003          MOV     #3,-(SP)
2934  011142  010600                  MOV     SP,R0
2935  011144  104417                  TRAP    C$PNTF
2936  011146  062706  000010          ADD     #10,SP
2937  011152                  PRINTB  #TFM45A
2938  011152  012746  004524          MOV     #TFM45A,-(SP)
2939  011156  012746  000001          MOV     #1,-(SP)
2940  011162  010600                  MOV     SP,R0
2941  011164  104414                  TRAP    C$PNTB
2942  011166  062706  000004          ADD     #4,SP
2943  011172                  PRINTB  #STM
2944  011172  012746  005627          MOV     #STM,-(SP)
2945  011176  012746  000001          MOV     #1,-(SP)
2946  011202  010600                  MOV     SP,R0
2947  011204  104414                  TRAP    C$PNTB
2948  011206  062706  000004          ADD     #4,SP
2949  011212                  ENDMSG
2950  011212                  L10032:
2951  011212  104423                  TRAP    C$MSG
2952  011214                  BGNMSG  ERR46
2953  011214                  ERR46::
2954  011214                  PRINTF  #TFM46,$GDDAT,R4
2955  011214  010446                  MOV     R4,-(SP)
2956  011216  013746  002452          MOV     $GDDAT,-(SP)
2957  011222  012746  004647          MOV     #TFM46,-(SP)
2958  011226  012746  000003          MOV     #3,-(SP)
2959  011232  010600                  MOV     SP,R0
2960  011234  104417                  TRAP    C$PNTF
2961  011236  062706  000010          ADD     #10,SP
2962  011242                  PRINTB  #STM
2963  011242  012746  005627          MOV     #STM,-(SP)
2964  011246  012746  000001          MOV     #1,-(SP)
2965  011252  010600                  MOV     SP,R0
2966  011254  104414                  TRAP    C$PNTB
2967  011256  062706  000004          ADD     #4,SP
2968  011262                  ENDMSG
```

```
CZDMQD.P11    12-JAN-82 09:50        GLOBAL ERROR REPORT SECTION

 2969  011262
 2970  011262  104423                L10033:
 2971                                        TRAP    C$MSG
 2972  011264                        BGNMSG  ERR47
 2973  011264                        ERR47::
 2974  011264                        PRINTF  #TFM47,R5,R4
 2975  011264  010446                        MOV     R4,-(SP)
 2976  011266  010546                        MOV     R5,-(SP)
 2977  011270  012746  004733               MOV     #TFM47,-(SP)
 2978  011274  012746  000003               MOV     #3,-(SP)
 2979  011300  010600                        MOV     SP,R0
 2980  011302  104417                        TRAP    C$PNTF
 2981  011304  062706  000010               ADD     #10,SP
 2982  011310                        PRINTB  #STM
 2983  011310  012746  005627               MOV     #STM,-(SP)
 2984  011314  012746  000001               MOV     #1,-(SP)
 2985  011320  010600                        MOV     SP,R0
 2986  011322  104414                        TRAP    C$PNTB
 2987  011324  062706  000004               ADD     #4,SP
 2988  011330                        ENDMSG
 2989  011330                        L10034:
 2990  011330  104423                        TRAP    C$MSG
 2991
```

```
2992
2993
2994
                        .SBTTL  REPORT CODING SECTION
```

```
CZDMQD.P11    12-JAN-82 09:50        REPORT CODING SECTION
  2995                               ;++
```

```
CZDMQD.P11    12-JAN-82 09:50          REPORT CODING SECTION

  2996                                 ; THE REPORT CODING SECTION CONTAINS THE
  2997                                 ; 'PRINTS'' CALLS THAT GENERATE STATISTICAL REPORTS.
  2998                                 ;--
  2999
  3000  011332                               BGNRPT
  3001  011332                         L$RPT::
  3002
  3003
  3004  011332                               EXIT    RPT
  3005  011332  000167                       .WORD   J$JMP
  3006  011334  000000                       .WORD   L10035-2-.
  3007
```

CZDMQD.P11 12-JAN-82 09:50 REPORT CODING SECTION

```
3008
3009  011336
3010  011336
3011  011336  104425
3012
```

```
L10035:  ENDRPT
         TRAP    C$RPT
```

3013
3014

CZDMQD.P11 12-JAN-82 09:50 REPORT CODING SECTION

3015
3016

```
CZDMQD.P11    12-JAN-82 09:50          INITIALIZE SECTION
  3017                                 .SBTTL  INITIALIZE SECTION
```

```
                                ;////////////////////////////////////////////////////////////////////////
                                ;// THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ;// AT THE BEGINNING OF EACH PASS.
                                ;////////////////////////////////////////////////////////////////////////

                                        BGNINIT
011340                          L$INIT::

                                ;INITIALIZE SUBROUTINE STACK
011340  012705  002730                  MOV     #SSTACK,R5
                                ;STORE BASE LEVEL PROGRAM STACK POINTER
011344  010637  002344                  MOV     SP,PSTACK
011350  005737  002462                  TST     FTIME
011354  001011                          BNE     1$
011356  013737  000004  002464          MOV     @#4,SAVE4
011364  013737  000006  002466          MOV     @#6,SAVE6
011372  012737  000001  002462          MOV     #1,FTIME
011400  013737  002464  000004  1$:     MOV     SAVE4,@#4
011406  013737  002466  000006          MOV     SAVE6,@#6

                                ;SEE IF PROGRAM JUST STARTED, BR IF YES
011414                                  READEF  #EF.START
011414  012700  000040                  MOV     #EF.START,R0
011420  104447                          TRAP    C$REFG
011422                                  BCOMPLETE       NEWST
011422  103414                          BCS     NEWST
                                ;SEE IF THIS IS A NEW PASS, BR IF YES
011424                                  READEF  #EF.NEW
011424  012700  000035                  MOV     #EF.NEW,R0
011430  104447                          TRAP    C$REFG
011432                                  BCOMPLETE       NEWST
011432  103410                          BCS     NEWST
                                ;SEE IF PROGRAM WAS JUST CONTINUED
011434                                  READEF  #EF.CONTINUE
011434  012700  000036                  MOV     #EF.CONTINUE,R0
011440  104447                          TRAP    C$REFG
011442                                  BCOMPLETE       ENDIT
011442  103576                          BCS     ENDIT
                                ;SEE IF PROGRAM JUST RESTARTED, BR IF NOT
011444                                  READEF  #EF.RESTART
011444  012700  000037                  MOV     #EF.RESTART,R0
011450  104447                          TRAP    C$REFG
011452                                  BNCOMPLETE      GETPRM
011452  103003                          BCC     GETPRM

011454                          NEWST:
                                ;RESET LOGICAL DEVICE TO -1
011454  012737  177777  002342          MOV     #-1,LOGDEV

                                ;GET UNIBUS ADRS, VECTOR, PRIORITY LEVEL, LINE UNIT,SWITCH
                                ;    PACKS, TEST CONNECTOR INFO. FOR THIS M8200,4,7 (CURRENT LOGICAL
                                ;    DEVICE).
011462                          GETPRM:
011462  005237  002342                  INC     LOGDEV
011466  023737  002342  002012          CMP     LOGDEV,L$UNIT
```

```
3074  011474  002367                          BGE     NEWST
3075  011476                                  GPHARD  LOGDEV,R1
3076  011476  013700  002342                  MOV     LOGDEV,R0
3077  011502  104442                          TRAP    CSGPHRD
3078  011504  010001                          MOV     R0,R1
3079  011506                                  BNCOMPLETE      GETPRM
3080  011506  103365                          BCC     GETPRM
3081  011510  012137  002414                  MOV     (R1)+,WTYPE
3082                                  ;GET ADDRESS OF M8200,4,7
3083  011514  011137  002516                  MOV     (R1),KMCSR
3084                                  ;GET POINTER TO M8200,4,7 CSR HI BYTE
3085  011520  011137  002520                  MOV     (R1),KMCSRH
3086  011524  005237  002520                  INC     KMCSRH
3087                                  ;GET POINTER TO M8200,4,7 CTL OUT REG
3088  011530  011137  002522                  MOV     (R1),KMCTL
3089  011534  062737  000002  002522          ADD     #2,KMCTL
3090                                  ;GET POINTER TO M8200,4,7 PORT REG - SEL 4
3091  011542  011137  002524                  MOV     (R1),KMPO4
3092  011546  062737  000004  002524          ADD     #4,KMPO4
3093                                  ;GET POINTER TO M8200,4,7 PORT REG - SEL 6
3094  011554  012137  002526                  MOV     (R1)+,KMPO6
3095  011560  062737  000006  002526          ADD     #6,KMPO6
3096                                  ;GET POINTER TO RCV VECTOR
3097  011566  011137  002506                  MOV     (R1),KMRVEC
3098                                  ;GET POINTER TO RCV PRIORITY LEVEL
3099  011572  011137  002510                  MOV     (R1),KMRLVL
3100  011576  062737  000002  002510          ADD     #2,KMRLVL
3101                                  ;GET POINTER TO TX VECTOR
3102  011604  011137  002512                  MOV     (R1),KMTVEC
3103  011610  062737  000004  002512          ADD     #4,KMTVEC
3104                                  ;GET POINTER TO TX PRIORITY LEVEL
3105  011616  011137  002514                  MOV     (R1),KMTLVL
3106  011622  062737  000006  002514          ADD     #6,KMTLVL
3107                                  ;PUT VECTOR INTO STAT1
3108  011630  016137  000020  002472          MOV     20(R1),RUNINH
3109  011636  012137  002500                  MOV     (R1)+,STAT1
3110                                  ;PUT PRIORITY INTO STAT1
3111  011642  052137  002500                  BIS     (R1)+,STAT1
3112                                  ;SEE IF NO LINE UNIT, SET BIT IF YES
3113  011646  005711                          TST     (R1)
3114  011650  001004                          BNE     50000$
3115  011652  052737  010000  002500          BIS     #BIT12,STAT1
3116  011660  000416                          BR      4$
3117  011662                                  50000$:
3118                                  ;SEE IF M8201 LINE UNIT, SET BIT IF YES
3119  011662  021127  000001                  CMP     (R1),#1
3120  011666  001001                          BNE     50001$
3121  011670  000412                          BR      4$
3122  011672                                  50001$:
3123                                  ;SEE IF M8202 LINE UNIT, SET BIT IF YES
3124  011672  021127  000002                  CMP     (R1),#2
3125  011676  001004                          BNE     50002$
3126  011700  052737  020000  002500          BIS     #BIT13,STAT1
3127  011706  000403                          BR      4$
3128  011710                                  50002$:
3129                                  ;SET BIT FOR M8203 LINE UNIT
```

```
3130  011710  052737  100000  002500          BIS     #BIT15,STAT1
3131  011716                                4$:
3132                                        ;SET BIT IN STAT1 FOR TEST CONNECTOR
3133  011716  056137  000006  002500          BIS     6(R1),STAT1
3134  011724  062701  000002                  ADD     #2,R1
3135                                        ;SET SWITCH PACK #1 IN STAT2 LOW BYTE
3136  011730  012137  002502                  MOV     (R1)+,STAT2
3137                                        ;SET SWITCH PACK #2 IN STAT2 HIGH BYTE
3138  011734  111137  002503                  MOVB    (R1),STAT2+1
3139
3140                                        ;INCREMENT LOGICAL UNIT (DEVICE) NUMBER
3141                                        ;       INC     LOGDEV
3142  011740  000240                          NOP
3143  011742  000240                          NOP
3144
3145  011744  012737  002000  002436          MOV     #2000,MEMSZ
3146  011752  005037  002432                  CLR     TYPE
3147  011756  123727  002414  000000          CMPB    WTYPE,#0
3148  011764  001425                          BEQ     ENDIT
3149  011766  123727  002414  000004          CMPB    WTYPE,#4        ;KMC?
3150  011774  001004                          BNE     5$
3151  011776  012737  000001  002432          MOV     #1,TYPE
3152  012004  000415                          BR      ENDIT
3153  012006  012737  007777  002436  5$:     MOV     #7777,MEMSZ
3154  012014  123727  002414  000006          CMPB    WTYPE,#6
3155  012022  001003                          BNE     7$
3156  012024  012737  000001  002432          MOV     #1,TYPE
3157  012032  013737  002472  002470  7$:     MOV     RUNINH,RUNB
3158  012040                          6$:
3159  012040                          ENDIT:
3160  012040                                  ENDINIT
3161  012040                          L10036:
3162  012040  104411                          TRAP    C$INIT
3163
3164                                  .EVEN
3165  012042                                  BGNAUTO
3166  012042                          L$AUTO::
3167                                          ;DEVICE DOES NOT HAVE A "READY"
3168  012042  013701  002516          MOV     KMCSR,R1        ;R1 CONTAINS BASE M8200,4,7 ADDRESS
3169  012046  012705  000004          MOV     #4,R5           ;4 REGISTERS TO BE TESTED
3170  012052  012737  012104  000004  MOV     #2$,4           ;SET OUT TIMEOUT TRAP
3171  012060  012737  000240  000006  MOV     #240,6          ;LEVEL 7
3172  012066  005711                  1$:     TST     (R1)    ;REFERENCE DEVICE REGISTERS
3173  012070  000240                          NOP
3174  012072  062701  000002          ADD     #2,R1           ;NEXT REGISTER
3175  012076  005305                  DEC     R5              ;DEC REGISTER COUNT
3176  012100  001372                  BNE     1$              ;BR IF NOT LAST REGISTER
3177  012102  000405                  BR      3$
3178
3179  012104  062706  000004          2$:     ADD     #4,SP
3180  012110                          DODU    LOGDEV
3181  012110  013700  002342          MOV     LOGDEV,R0
3182  012114  104451                  TRAP    C$DODU
3183
3184  012116  013737  002464  000004  3$:     MOV     SAVE4,4
3185  012124  013737  002466  000006          MOV     SAVE6,6
```

CZDMQD.P11 12-JAN-82 09:50 INITIALIZE SECTION

```
3186  012132
3187  012132
3188  012132  104461          ENDAUTO
3189                L10037:     TRAP    C$AUTO
```

CZDMQD.P11 12-JAN-82 09:50 CLEANUP CODING SECTION

```
3190
3191
3192
3193
3194
3195
3196
3197  012134                        .SBTTL  CLEANUP CODING SECTION
3198  012134
3199  012134
3200  012134  104433
3201
3202  012136
3203  012136
3204  012136  104412
3205
3206
3207
3208
3209
```

```
;////////////////////////////////////////////////////////////////////////
;// THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
;// AT THE END OF EACH PASS.
;////////////////////////////////////////////////////////////////////////

        BGNCLN
L$CLEAN::
        BRESET
        TRAP    C$RESET

        ENDCLN
L10040:
        TRAP    C$CLEAN
```

```
                                .SBTTL  DROP UNIT SECTION

                                ;///////////////////////////////////////////////////////////////////////////
                                ;/ THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
                                ;/ TO NO LONGER BE TESTED.
                                ;///////////////////////////////////////////////////////////////////////////

 012140                                 BGNDU
 012140                         L$DU::
                                ;ISSUE UNIBUS RESET TO CLEAN UP
 012140                                 BRESET
 012140  104433                         TRAP    C$RESET
 012142                                 ENDDU
 012142                         L10041:
 012142  104453                         TRAP    C$DU
```

```
3230
3231
3232                    .SBTTL  ADD UNIT SECTION
3233
3234                    ;////////////////////////////////////////////////////////////////////
3235                    ;// THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
3236                    ;//  TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING.  IF
3237                    ;//  'EF.AUNIT" IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
3238                    ;////////////////////////////////////////////////////////////////////
3239
3240 012144                       BGNAU
3241 012144            L$AU::
3242 012144                       ENDAU
3243 012144            L10042:
3244 012144  104452               TRAP    C$AU
3245
3246
3247
3248
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS

                                       .SBTTL  HARDWARE TESTS


                                       ;START OF CODE BLOCK WHICH IS USED AS DATA
012146                                 ROMMAP:

                                               BADHEAD
012146                                         ;*************** TEST 1 ****************
                                               ;*VERIFY THAT REFERENCING UNIBUS DEVICE REGISTERS
                                               ;*DOES NOT CAUSE A TIME OUT TRAP
012146                                         BADHEAD
                                               ;*************** TEST 1 ****************

012146                                 BGNTST
012146                                 T1::
012146  013701  002516                         MOV     KMCSR,R1        ;R1 CONTAINS BASE M8200,4,7 ADDRESS
012152  012705  000004                         MOV     #4,R5           ;4 REGISTERS TO BE TESTED
012156  012737  012214  000004                 MOV     #2$,4           ;SET OUT TIMEOUT TRAP
012164  012737  000240  000006                 MOV     #240,6          ;LEVEL 7
012172  005711                         1$:     TST     (R1)            ;REFERENCE DEVICE REGISTERS
012174  000240                               NOP
012176                                       ESCAPE  TST
012176  104410                               TRAP    C$ESCAPE
012200  000054                               .WORD   L10043-.
012202  062701  000002                       ADD     #2,R1           ;NEXT REGISTER
012206  005305                               DEC     R5              ;DEC REGISTER COUNT
012210  001370                               BNE     1$              ;BR IF NOT LAST REGISTER
012212  000410                               BR      3$

012214  062706  000004                 2$:     ADD     #4,SP
012220                                       ERROR   36              ;TIME OUT ERROR
012224  104455                               TRAP    C$ERDF
012226  000044                               .WORD   36
012230  005055                               .WORD   EM0
012232  010614                               .WORD   ERR36

012234  013737  002464  000004         3$:     MOV     SAVE4,4
012242  013737  002466  000006                 MOV     SAVE6,6
012250                                       ESCAPE  TST
012250  104410                               TRAP    C$ESCAPE
012252  000002                               .WORD   L10043-.

012254                                 ENDTST
012254                                 L10043:
012254  104401                               TRAP    C$ETST
                                       .EVEN


012256                                         BADHEAD
                                               ;*************** TEST 2 ****************
                                               ;*TEST OF BR RIGHT SHIFT
                                               ;*VERIFY THAT A DEST OF BR RSH (011) OF A MICRO-INSTRUCTION
                                               ;*SHIFTS THE RESULTING BR DATA RIGHT ONCE.
012256                                         BADHEAD
                                               ;*************** TEST 2 ****************
```

```
3305
3306  012256                          BGNTST
3307  012256                          T2::
3308
3309  012256                                                ;R1 CONTAINS BASE M8200,4,7 ADDRESS
3310  012262  013701  002516                MSTCLR
                                      MOV     KMCSR,R1      ;MASTER CLEAR M8200,4,7
3311  012266  005011                 CLR     (R1)          ;R1 = M8200,4,7 BASE ADDRESS
3312  012270  012705  052525         MOV     #52525,R5     ;CLEAR SELO
3313  012274  010561  000004         MOV     R5,4(R1)      ;START WITH 125
3314  012300                        ROMCLK                  ;PORT4 125
3315  012300  004537  003044         JSR     R5,..ROMCLK   ;NEXT WORD IS INSTRUCTION
3316  012304  120500                 120500                ;CLOCK INSTRUCTION
3317  012306                        ROMCLK                  ;PORT4 TO BR-REG
3318  012306  004537  003044         JSR     R5,..ROMCLK   ;NEXT WORD IS INSTRUCTION
3319  012312  061620                 061620                ;CLOCK INSTRUCTION
3320  012314                        ROMCLK                  ;BR RSH BR, SHIFT BR RIGHT
3321  012314  004537  003044         JSR     R5,..ROMCLK   ;NEXT WORD IS INSTRUCTION
3322  012320  061225                 061225                ;CLOCK INSTRUCTION
3323  012322  006005                 ROR     R5            ;PORT5 BR
3324  012324  005004                 CLR     R4            ;R5 = "EXPECTED"
3325  012326  116104  000005         MOVB    5(R1),R4      ;R4 = "FOUND"
3326  012332  120504                 CMPB    R5,R4         ;DID BR SHIFT RIGHT ONCE?
3327  012334  001410                 BEQ     1$            ;BR IF YES
3328  012336                        ERROR   12            ;BR RIGHT SHIFT ERROR
3329  012342  104455                 TRAP    C$ERDF
3330  012344  000014                 .WORD   12
3331  012346  005055                 .WORD   EM0
3332  012350  007420                 .WORD   ERR12
3333
3334  012352                                                ;SHOULD BE 52
3335  012352  104410                ESCAPE TST
3336  012354  000044                 TRAP    C$ESCAPE
3337  012356                         .WORD   L10044-.
3338  012356                  1$:
3339  012356  004537  003044                ROMCLK                  ;NEXT WORD IS INSTRUCTION
3340  012362  061620                 JSR     R5,..ROMCLK   ;CLOCK INSTRUCTION
3341  012364                         061620                ;BR RSH BR, SHFT BR RIGHT AGAIN
3342  012364  004537  003044        ROMCLK                  ;NE4XT WORD IS INSTRUCTION
3343  012370  061225                 JSR     R5,..ROMCLK   ;CLOCK INSTRUCTION
3344  012372  006005                 061225                ;PORT5 BR
3345  012374  116104  000005         ROR     R5            ;R5 = "EXPECTED"
3346  012400  120504                 MOVB    5(R1),R4      ;R4 = "FOUND"
3347  012402  001406                 CMPB    R5,R4         ;DID BR SHIFT RIGHT?
3348  012404                         BEQ     2$            ;BR IF YES
3349  012410  104455                 ERROR   12            ;BR RIGHT SHIFT ERROR
3350  012412  000014                 TRAP    C$ERDF
3351  012414  005055                 .WORD   12
3352  012416  007420                 .WORD   EM0
3353                                  .WORD   ERR12
3354  012420                                                ;S/B 25
3355  012420                  2$:
3356  012420                  ENDTST
3357  012420  104401          L10044:
3358                                  TRAP    C$ETST
3359  012422                          BADHEAD
3360                                  ;*************** TEST 3 ****************
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS

                                       ;*IOP CRAM WRITE/READ TEST
                                       ;*FLOAT A 1 THROUGH EACH CRAM LOCATION
                                       BADHEAD
012422                                 ;**************** TEST 3 ****************


012422                                 BGNTST
012422                                 T3::
012422                                       MACEX
                                             ;DO NOT DO TEST IF M8200
012430  104432                               TRAP    C$EXIT
012432  000116                               .WORD   L10045-.
012434                                       MYINT
012434  013701  002516                       MOV     KMCSR,R1            ;RECORD DEVICE ADDR.
                                                                         ;R1 CONTAINS BASE M8200,4,7 ADDRESS
012440  005037  002434                       CLR     MR0                 ;MR0 = CRAM ADDRESS
012444  012702  000001               ADR4:   MOV     #1,R2               ;R2 = WRITE DATA
012450                               ADR5:
012450                                       BGNSEG
012450  104404                               TRAP    C$BSEG
012452  012711  002000               3$:     MOV     #BIT10,(R1)         ;SET ROM0
012456  013761  002434  000004               MOV     MR0,4(R1)           ;WRITE ADDRESS TO SEL4
012464  010261  000006                       MOV     R2,6(R1)            ;LOAD SEL6 WITH WRITE DATA
012470  052711  020000                       BIS     #BIT13,(R1)         ;WRITE SEL6 INTO CRAM
012474  016104  000006                       MOV     6(R1),R4            ;READ CRAM INTO "FOUND"
012500  020204                               CMP     R2,R4               ;IS DATA CORRECT?
012502  001410                               BEQ     4$                  ;BR IF OK
012504                                       ERROR   1                   ;ERROR
012510  104455                               TRAP    C$ERDF
012512  000001                               .WORD   1
012514  005055                               .WORD   EM0
012516  006032                               .WORD   ERR1
012520                                       ESCAPE  SEG
012520  104410                               TRAP    C$ESCAPE
012522  000002                               .WORD   10000$-.
012524                               4$:     ENDSEG
012524                               10000$:
012524  104405                               TRAP    C$ESEG
012526  000241                               CLC                         ;CLEAR CARRY
012530  006102                               ROL     R2                  ;SHIFT WRITE DATA
012532  001346                               BNE     ADR5                ;BSR IF NOT DONE THIS ADDRESS
012534  005237  002434                       INC     MR0                 ;BUMP TO NEXT CRAM ADDRESS
012540  023737  002436  002434               CMP     MEMSZ,MR0           ;DONE YET?
012546  001336                               BNE     ADR4                ;BR IF NO
012550                               5$:
012550                               ENDTST
012550                               L10045:
012550  104401                               TRAP    C$ETST

012552                                       BADHEAD
                                             ;**************** TEST 4 ****************
                                             ;*IOP CRAM WRITE/READ TEST
                                             ;*FLOAT A 0 THROUGH EACH CRAM LOCATION
012552                                       BADHEAD
                                             ;**************** TEST 4 ****************


012552                               BGNTST
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
012552                          T4::
012552                          MACEX
                                        ;DO NOT DO TEST IF M8200
012560  104432                  TRAP    C$EXIT
012562  000126                  .WORD   L10046-.
012564                          MYINT
012564  013701  002516          MOV     KMCSR,R1            ;RECORD DEVICE ADDR.
012570                          MSTCLR                      ;MASTER CLEAR M8200,4,7
012574  005037  002434          CLR     MR0                 ;MR0 = CRAM ADDRESS
012600  012702  000001  ADR1:   MOV     #1,R2               ;R2 = WRITE DATA
012604                  ADR2:
012604                          BGNSEG
012604  104404                  TRAP    C$BSEG
012606  005102                  COM     R2                  ;MAKE IT A FLOATING ZERO
012610  012711  002000  3$:     MOV     #BIT10,(R1)         ;SET ROM0
012614  013761  002434  000004  MOV     MR0,4(R1)           ;WRITE ADDRESS TO SEL4
012622  010261  000006          MOV     R2,6(R1)            ;LOAD SEL6 WITH WRITE DATA
012626  052711  020000          BIS     #BIT13,(R1)         ;WRITE SEL6 INTO CRAM
012632  016104  000006          MOV     6(R1),R4            ;READ CRAM INTO "FOUND"
012636  020204                  CMP     R2,R4               ;IS DATA CORRECT?
012640  001410                  BEQ     4$                  ;BR IF OK
012642                          ERROR   1                   ;ERROR
012646  104455                  TRAP    C$ERDF
012650  000001                  .WORD   1
012652  005055                  .WORD   EM0
012654  006032                  .WORD   ERR1
012656                          ESCAPE  SEG
012656  104410                  TRAP    C$ESCAPE
012660  000002                  .WORD   10000$-.
012662                  4$:     ENDSEG
012662                  10000$:
012662  104405                  TRAP    C$ESEG
012664  005102                  COM     R2                  ;BACK TO FLOATING ONE
012666  000241                  CLC                         ;CLEAR CARRY
012670  006102                  ROL     R2                  ;SHIFT WRITE DATA
012672  001344                  BNE     ADR2                ;BR IF NOT DONE THIS ADDRESS
012674  005237  002434          INC     MR0                 ;BUMP TO NEXT CRAM ADDRESS
012700  023737  002436  002434  CMP     MEMSZ,MR0           ;DONE YET?
012706  001334                  BNE     ADR1                ;BR IF NO
012710                  5$:
012710                  ENDTST
012710                  L10046:
012710  104401                  TRAP    C$ETST

012712                          BADHEAD
                                ;*************** TEST 5 ***************
                                ;*IOP CRAM DUAL ADDRESSING TEST
                                ;*WRITE EACH ADDRESS INTO ITSELF, READ EACH
                                ;*ADDRESS TO VERIFY CORRECT ADDRESSING
012712                          BADHEAD
                                ;*************** TEST 5 ***************

012712                  BGNTST
012712                  T5::
012712                  MACEX
                                        ;DO NOT DO TEST IF M8200
```

```
3473  012720  104432                  TRAP    C$EXIT
3474  012722  000230                  .WORD   L10047-.
3475  012724                          MYINT
3476  012724  013701  002516          MOV     KMCSR,R1        ;RECORD DEVICE ADDR.
3477                                                          ;R1 CONTAINS BASE M8200,4,7 ADDRESS
3478  012730                          MSTCLR                  ;MASTER CLEAR M8200,4,7
3479  012734  005037  002434          CLR     MR0             ;MR0 =CRAM ADDRESS
3480  012740                          BGNSEG
3481  012740  104404                  TRAP    C$BSEG
3482  012742  013702  002434  1$:     MOV     MR0,R2          ;SAVE R2 FOR TYPEOUT
3483  012746  012711  002000          MOV     #BIT10,(R1)     ;SET ROM0
3484  012752  013761  002434  000004  MOV     MR0,4(R1)       ;WRITE ADDRESS TO SEL4
3485  012760  013761  002434  000006  MOV     MR0,6(R1)       ;LOAD SEL6 WITH WRITE DATA
3486  012766  052711  020000          BIS     #BIT13,(R1)     ;WRITE CRAM
3487  012772                          SKIP06  15$             ;IF M8206,SKIP NEXT INSTR.
3488                                  ;GOTO 15$ IF M8206
3489  013002  005061  000006          CLR     6(R1)           ;CLEAR SEL 6
3490  013006                  15$:
3491  013006  016104  000006          MOV     6(R1),R4        ;SHOULD READ BACK OWN ADDRESS
3492  013012  023704  002434          CMP     MR0,R4          ;IS DATA CORRECT?
3493  013016  001410                  BEQ     2$              ;BR IF YES
3494  013020                          ERROR   1               ;DATA ERROR
3495  013024  104455                  TRAP    C$ERDF
3496  013026  000001                  .WORD   1
3497  013030  005055                  .WORD   EM0
3498  013032  006032                  .WORD   ERR1
3499  013034                          ESCAPE  SEG
3500  013034  104410                  TRAP    C$ESCAPE
3501  013036  000002                  .WORD   10000$-.
3502  013040                  2$:     ENDSEG
3503  013040                  10000$:
3504  013040  104405                  TRAP    C$ESEG
3505  013042                          BGNSEG
3506  013042  104404                  TRAP    C$BSEG
3507  013044  005237  002434          INC     MR0             ;BUMP TO NEXT ADDRESS
3508  013050  023737  002436  002434  CMP     MEMSZ,MR0       ;DONE WRITING YET?
3509  013056  001331                  BNE     1$              ;BR IF NO
3510  013060  005037  002434          CLR     MR0             ;RESTART AT ADDRESS 0
3511  013064  013702  002434  3$:     MOV     MR0,R2          ;SAVE R2 FOR TYPEOUT
3512  013070  012711  002000          MOV     #BIT10,(R1)     ;SET ROM0
3513  013074  013761  002434  000004  MOV     MR0,4(R1)       ;SEL4 = CRAM ADDRESS
3514  013102  016104  000006          MOV     6(R1),R4        ;READ CRAM INTO "FOUND"
3515  013106  023704  002434          CMP     MR0,R4          ;IS DATA CORRECT?
3516  013112  001411                  BEQ     4$              ;BR IF YES
3517  013114                          ERROR   2               ;DUAL ADDRESSING ERROR
3518  013120  104455                  TRAP    C$ERDF
3519  013122  000002                  .WORD   2
3520  013124  005055                  .WORD   EM0
3521  013126  006160                  .WORD   ERR2
3522  013130                          ESCAPE  SEG
3523  013130  104410                  TRAP    C$ESCAPE
3524  013132  000002                  .WORD   10001$-.
3525  013134                          ENDSEG
3526  013134                  10001$:
3527  013134  104405                  TRAP    C$ESEG
3528  013136                  4$:                             ;LOOP TO 3$ IF SW09=1
```

```
3529  013136  005237  002434          INC     MR0             ;BUMP TO NEXT ADDRESS
3530  013142  023737  002436  002434  CMP     MEMSZ,MR0       ;DONE WRITING YET?
3531  013150  001345                  BNE     3$              ;BR IF NO
3532  013152                          5$:
3533  013152                          ENDTST
3534  013152                          L10047:
3535  013152  104401                  TRAP    C$ETST
3536
3537
3538  013154                          BADHEAD
3539                                  ;*************** TEST 6 ***************
3540                                  ;*IOP MAIN MEMORY TEST
3541                                  ;*FLOAT A 1 THROUGH ALL MAIN MEMORY LOCATIONS
3542  013154                          BADHEAD
3543                                  ;*************** TEST 6 ***************
3544
```

```
3545  013154                        BGNTST
3546  013154                        T6::
3547  013154                              MYINT
3548  013154  013701  002516              MOV     KMCSR,R1        ;RECORD DEVICE ADDR.
3549                                                              ;R1 CONTAINS BASE M8200,4,7 ADDRESS
3550  013160                              MSTCLR                  ;MASTER CLEAR M8200,4,7
3551  013164  005037  002406              CLR     FLAG            ;START WITH ADDRESS 0
3552  013170  012737  000001  002434 1$:  MOV     #1,MR0          ;START WITH BIT 0
3553  013176  042737  003777  013232 65$: BIC     #3777,66$       ;CLEAR ADDRESS FIELD OF INSTRUCTION
```

```
 3554  013204  042737  000037  013240          BIC     #37,68$         ;CLEAR ADDRESS FIELD OF INSTRUCTION
```

```
3555  013212  153737  002406  013232          BISB    FLAG,66$        ;ADD ADDRESS TO INSTRUCTION7
3556  013220  153737  002407  013240          BISB    FLAG+1,68$      ;ADD ADDRESS TO INSTRUCTION
3557  013226                                  ROMCLK                  ;NEXT WORD IS INSTRUCTION,
3558  013226  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
3559  013232  010000                  66$:    010000
3560  013234                                  ROMCLK
3561  013234  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
3562  013240  004000                  68$:    004000                  ;LOAD MAR HI
3563  013242  013761  002434  000004          MOV     MR0,4(R1)       ;WRITE PATTERN IN PORT4
3564  013250                                  ROMCLK                  ;NEXT WORD IS INSTRUCTION,
3565  013250  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
3566  013254  122500                          122500                  ;MOVE PORT4 TO MEMORY
3567  013256                                  ROMCLK                  ;NEXT WORD IS INSTRUCTION,
3568  013256  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
3569  013262  040620                          040620                  ;MOVE MEMORY TO BR
3570  013264                                  ROMCLK                  ;NEXT WORD IS INSTRUCTION,
3571  013264  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
3572  013270  061225                          61225                   ;MOVE BR TO PORT5
3573  013272  013705  002434                  MOV     MR0,R5          ;PUT "EXPECTED" IN R5
3574  013276  116104  000005                  MOVB    5(R1),R4        ;PUT "FOUND" IN R4
3575  013302  120504                          CMPB    R5,R4           ;DATA CORRECT?
3576  013304  001410                          BEQ     67$             ;BR IF YES
3577  013306                                  ERROR   6               ;DATA ERROR
3578  013312  104455                          TRAP    C$ERDF
3579  013314  000006                          .WORD   6
3580  013316  005055                          .WORD   EM0
3581  013320  006700                          .WORD   ERR6
3582  013322                                  ESCAPE  TST
3583  013322  104410                          TRAP    C$ESCAPE
3584  013324  000030                          .WORD   L10050-.
3585  013326                          67$:                            ;SW09=1?
3586  013326  000241                          CLC                     ;CLEAR CARRY
3587  013330  106137  002434                  ROLB    MR0             ;SHIFT BIT IN MR0
3588  013334  001320                          BNE     65$             ;DONE IF MR0=0
3589  013336                                  BREAK
3590  013336  104422                          TRAP    C$BRK
3591  013340  005237  002406                  INC     FLAG            ;NEXT ADDRESS
3592  013344  023737  002436  002406          CMP     MEMSZ,FLAG      ;LAST ADDRESS?
3593  013352  001306                          BNE     1$              ;BR IF NO
3594  013354                          2$:
3595  013354                          ENDTST
3596  013354                          L10050:
3597  013354  104401                          TRAP    C$ETST
3598  013356
3599  013356                                  BADHEAD
3600                                          ;*************** TEST 7 ***************
3601                                          ;*IOP MAIN MEMORY TEST
3602                                          ;*FLOAT A 0 THROUGH ALL MAIN MEMORY LOCATIONS
3603  013356                                  BADHEAD
3604                                          ;*************** TEST 7 ***************
3605
3606  013356                          BGNTST
3607  013356                          17::
3608  013356                                  MYINT
3609  013356  013701  002516                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
3610                                                                  ;R1 CONTAINS BASE M8200,4,7 ADDRESS
```

```
3611  013362                                          MSTCLR          ;MASTER CLEAR M8200,4,7
3612  013366  005037  002406                         CLR     FLAG    ;START WITH ADDRESS 0
3613  013372  012737  000001  002434          1$:    MOV     #1,MR0  ;START WITH BIT 0
3614  013400  005137  002434                  64$:    COM     MR0     ;CHANGE TO FLOATING 0
3615  013404  042737  003777  013440          65$:    BIC     #3777,66$       ;CLEAR ADDRESS FIELD OF INSTRUCTION
3616  013412  042737  000037  013446                  BIC     #37,68$         ;CLEAR ADDRESS FIELD OF INSTRUCTION
3617  013420  153737  002406  013440                  BISB    FLAG,66$        ;ADD ADDRESS TO INSTRUCTION
3618  013426  153737  002407  013446                  BISB    FLAG+1,68$      ;ADD ADDRESS TO INSTRUCTION
3619  013434                                          ROMCLK          ;NEXT WORD IS INSTRUCTION,
3620  013434  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3621  013440  010000                          66$:    010000          ;LOAD MAR LO WITH ADDRESS IN FLAG
3622  013442                                          ROMCLK          ;NEXT WORD IS INSTRUCTION,
3623  013442  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3624  013446  004000                          68$:    004000          ;LOAD MAR HI
3625  013450  013761  002434  000004                  MOV     MR0,4(R1)       ;WRITE PATTERN IN PORT4
3626  013456                                          ROMCLK          ;NEXT WORD IS INSTRUCTION,
3627  013456  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3628  013462  122500                                  122500          ;MOVE PORT4 TO MEMORY
3629  013464                                          ROMCLK          ;NEXT WORD IS INSTRUCTION,
3630  013464  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3631  013470  040620                                  040620          ;MOVE MEMORY TO BR
3632  013472                                          ROMCLK          ;NEXT WORD IS INSTRUCTION,
3633  013472  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3634  013476  061225                                  61225           ;MOVE BR TO PORT5
3635  013500  013705  002434                          MOV     MR0,R5          ;PUT "EXPECTED" IN R5
3636  013504  116104  000005                          MOVB    5(R1),R4        ;PUT "FOUND" IN R4
3637  013510  120504                                  CMPB    R5,R4           ;DATA CORRECT?
3638  013512  001406                                  BEQ     67$             ;BR IF YES
3639  013514                                          ERROR   6               ;DATA ERROR
3640  013520  104455                                  TRAP    C$ERDF
3641  013522  000006                                  .WORD   6
3642  013524  005055                                  .WORD   EM0
3643  013526  006700                                  .WORD   ERR6
3644  013530                                  67$:    ESCAPE  TST
3645  013530  104410                                  TRAP    C$ESCAPE
3646  013532  000034                                  .WORD   L10051-.
3647  013534  005137  002434                          COM     MR0             ;CHANGE TO FLOATING 1
3648  013540  000241                                  CLC                     ;CLEAR CARRY
3649  013542  106137  002434                          ROLB    MR0             ;SHIFT BIT IN MR0
3650  013546  001314                                  BNE     64$             ;DONE IF MR0=0
3651  013550                                          BREAK
3652  013550  104422                                  TRAP    C$BRK
3653  013552  005237  002406                          INC     FLAG            ;NEXT ADDRESS
3654  013556  023737  002436  002406                  CMP     MEMSZ,FLAG      ;LAST ADDRESS?
3655  013564  001302                                  BNE     1$              ;BR IF NO
3656  013566                                  2$:
3657  013566                                  ENDTST
3658  013566                                  L10051:
3659  013566  104401                                  TRAP    C$ETST
3660
3661  013570                                          BADHEAD
3662                                                  ;*************** TEST 8 ***************
3663                                                  ;*IOP MAIN MEMORY DUAL ADDRESSING TEST
3664                                                  ;*LOAD EACH MEMORY LOCATION WITH ITS OWN ADDRESS
3665                                                  ;*READ BACK EACH LOCATION TO VERIFY CORRECT ADDRESSING
3666  013570                                          BADHEAD
```

```
                                                ;*************** TEST 8 ***************

                                        BGNTST
                                        T8::
                                                MYINT
013570  013701  002516                          MOV     KMCSR,R1        ;RECORD DEVICE ADDR.
                                                                        ;R1 CONTAINS BASE M8200,4,7 ADDRESS
013574                                          MSTCLR                  ;MASTER CLEAR M8200,4,7
013600  005037  002406                          CLR     FLAG            ;START WITH ADDRESS 0
013604  013702  002406                  1$:     MOV     FLAG,R2         ;PUT DATA IN R2
013610  042737  003777  013644                  BIC     #3777,2$        ;CLEAR ADDRESS FIELD OF INSTRUCTION
013616  042737  000037  013652                  BIC     #37,7$          ;CLEAR ADDRESS FIELD OF INSTRUCTION
013624  153737  002406  013644                  BISB    FLAG,2$         ;ADD ADDRESS TO INSTRUCTION
013632  153737  002407  013652                  BISB    FLAG+1,7$       ;ADD ADDRESS TO INSTRUCTION
013640                                          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
013640  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
013644  010000                          2$:     010000                  ;LOAD MAR LO
013646                                          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
013646  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
013652  004000                          7$:     004000                  ;LOAD MAR HI
013654  010261  000004                          MOV     R2,4(R1)
013660                                          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
013660  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
013664  122500                                  122500                  ;MOVE PORT4 TO MEMORY
013666                                          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
013666  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
013672  040620                                  040620                  ;MOVE MEMORY TO THE BR
013674                                          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
013674  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
013700  061225                                  61225                   ;MOVE BR TO PORT5
013702  010205                                  MOV     R2,R5           ;PUT "EXPECTED" IN R5
013704  116104  000005                          MOVB    5(R1),R4        ;PUT "FOUND" IN R4
013710  120504                                  CMPB    R5,R4           ;DATA CORRECT?
013712  001406                                  BEQ     3$              ;BR IF YES
013714                                          ERROR   6               ;DATA ERROR
013720  104455                                  TRAP    C$ERDF
013722  000006                                  .WORD   6
013724  005055                                  .WORD   EM0
013726  006700                                  .WORD   ERR6
013730                                  3$:     ESCAPE  TST
013730  104410                                  TRAP    C$ESCAPE
013732  000156                                  .WORD   L10052-.
013734  005237  002406                          INC     FLAG            ;NEXT ADDRESS
013740  023737  002436  002406                  CMP     MEMSZ,FLAG      ;LAST ADDRESS?
013746  001316                                  BNE     1$              ;BR IF NO
013750  012737  013762  002340                  MOV     #4$,LOCK        ;NEW SCOPE 1
013756  005037  002406                          CLR     FLAG            ;RESTART AT ADDRESS 0
013762  013702  002406                  4$:     MOV     FLAG,R2         ;PUT DATA IN R2
013766  042737  003777  014014                  BIC     #3777,5$        ;CLEAR ADDRESS FIELD OF INSTRUCTION
013774  042737  000037  014022                  BIC     #37,8$          ;CLEAR ADDRESS FIELD OF INSTRUCTION
014002  153737  002406  014014                  BISB    FLAG,5$         ;ADD ADDRESS TO INSTRUCTION
014010                                          ROMCLK                  ;NEXT WORD IS INSTRUCTION, ROMCLK PC+5304
014010  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
014014  010000                          5$:     010000                  ;LOAD THE MAR LO
014016                                          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
014016  004537  003044                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS

  3723  014022  004000                 8$:     004000                  ;LOAD MAR HI
  3724  014024                                 ROMCLK                  ;NEXT WORD IS INSTRUCTION,
  3725  014024  004537  003044                 JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
  3726  014030  040620                         040620                  ;MOVE MEMORY TO THE BR
  3727  014032                                 ROMCLK                  ;NEXT WORD IS INSTRUCTION,
  3728  014032  004537  003044                 JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
  3729  014036  061225                         61225                   ;MOV BR TO PORT5
  3730  014040  010205                         MOV     R2,R5           ;PUT 'EXPECTED' IN R5
  3731  014042  116104  000005                 MOVB    5(R1),R4        ;PUT 'FOUND' IN R4
  3732  014046  120504                         CMPB    R5,R4           ;DATA CORRECT?
  3733  014050  001406                         BEQ     6$              ;BR IF YES
  3734  014052                                 ERROR   6               ;ADDRESSING ERROR
  3735  014056  104455                         TRAP    C$ERDF
  3736  014060  000006                         .WORD   6
  3737  014062  005055                         .WORD   EM0
  3738  014064  006700                         .WORD   ERR6
  3739  014066                         6$:     ESCAPE  TST
  3740  014066  104410                         TRAP    C$ESCAPE
  3741  014070  000020                         .WORD   L10052-.
  3742  014072                                 BREAK
  3743  014072  104422                         TRAP    C$BRK
  3744  014074  005237  002406                 INC     FLAG            ;NEXT ADDRESS
  3745  014100  023737  002436  002406         CMP     MEMSZ,FLAG      ;IS IT THE LAST
  3746  014106  001325                         BNE     4$              ;BR IF NO
  3747  014110                         9$:
  3748  014110                         ENDTST
  3749  014110                         L10052:
  3750  014110  104401                         TRAP    C$ETST
  3751
  3752  014112                                 BADHEAD
                                               ;*************** TEST 9 ****************
                                               ;*IOP MAR TEST
                                               ;*PERFORM DUAL ADDRESSING TEST
                                               ;*USING MAR AUTO-INC FEATURE
  3757  014112                                 BADHEAD
                                               ;*************** TEST 9 ****************

  3760  014112                         BGNTST
  3761  014112                         T9::
  3762  014112                                 K4ONLY                  ;FOR 4K CPUS ONLY.
                                               ;DO NOT DO TEST IF M8200, OR M8204
  3764  014122  104432                         TRAP    C$EXIT
  3765  014124  000342                         .WORD   L10053-.
  3766  014126                                 MYINT
  3767  014126  013701  002516                 MOV     KMCSR,R1        ;RECORD DEVICE ADDR.
                                               ;R1 CONTAINS BASE M8200,4,7 ADDRESS
  3769  014132                                 MSTCLR                  ;MASTER CLEAR M8200,4,7
  3770  014136  005002                         CLR     R2              ;START WITH A ZERO
  3771  014140  013703  002436                 MOV     MEMSZ,R3        ;GET MEMORY SIZE
  3772  014144  005203                         INC     R3              ;STOP ADDR=MEMSZ+1
  3773  014146                                 ROMCLK                  ;NEXT WORD IS INSTRUCTION,
  3774  014146  004537  003044                 JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
  3775  014152  010000                         010000                  ;LOAD MAR WITH A ZERO
  3776  014154                                 CLRMAR
  3777  014154  004537  003044                 JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
  3778  014162  010261  000004         1$:     MOV     R2,4(R1)        ;WRITE DATA TO PORT4
```

```
3779  014166
3780  014166  004537  003044          ROMCLK                  ;NEXT WORD IS INSTRUCTION.
3781  014172  136500                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3782  014174  005202                  136500                  ;MEM_PORT4, AUTO-INC MAR
3783  014176  020302                  INC     R2              ;INCREMENT DATA
3784  014200  001370                  CMP     R3,R2   ;DONE YET?
3785  014202  005002                  BNE     1$              ;BR IF NO
3786  014204                          CLR     R2              ;RESTART WITH A ZERO
3787  014204  004537  003044          ROMCLK                  ;NEXT WORD IS INSTRUCTION.
3788  014210  010000                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3789  014212                          010000                  ;LOAD MAR WITH A ZERO
3790  014212  004537  003044          CLRMAR
3791  014220                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3792  014220                  2$:
3793  014220  004537  003100          SROMCLK                 ;NEXT WORD IS INSTRUCTION.
3794  014224  055224                  JSR     R5,..SROMCLK
3795  014226  010205                  055224                  ;MOVE MEM TO PORT4
3796  014230  116104  000004          MOV     R2,R5           ;PUT "EXPECTED" IN R5
3797  014234  120504                  MOVB    4(R1),R4        ;PUT "FOUND" IN R4
3798  014236  001406                  CMPB    R5,R4           ;DATA CORRECT?
3799  014240                          BEQ     3$              ;BR IF YES
3800  014244  104455                  ERROR   11              ;MAR ERROR
3801  014246  000013                  TRAP    C$ERDF
3802  014250  005055                  .WORD   11
3803  014252  007276                  .WORD   EM0
3804  014254                          .WORD   ERR11
3805  014254  004537  003100  3$:     SROMCLK
3806  014260  000000                  JSR     R5,..SROMCLK
3807  014262  005004                  0                       ;DUMP NOP INSTR. TO CLK AUTO INC IN MAR.
3808  014264                          CLR     R4
3809  014264  004537  003044          ROMCLK          ;READ IBUS* <15> (MAR HIGH)
3810  014270  121325                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3811                                  121325          ;MAR HIGH _POT 5
3812  014272
3813  014272  004537  003044          ROMCLK          ;READ IBUS* <14> (MAR LOW)
3814  014276  121304                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3815  014300  016104  000004          121304
3816  014304  042704  160000          MOV     4(R1),R4        ;ADD TO MAR HIGH.
3817  014310  005202                  BIC     #160000,R4
3818  014312  020237  002436          INC     R2
3819  014316  001002                  CMP     R2,MEMSZ
3820  014320  052702  010000          BNE     35$
3821  014324                          BIS     #10000,R2       ;IF AT HIGH LIMIT,ADD IN OVERFLOW BIT.
3822  014324  020204          35$:
3823  014326  001406                  CMP     R2,R4   ;ADDR. OK?
3824  014330                          BEQ     4$
3825  014334  104455                  ERROR   11              ;ERROR MAR ADDR. BAD IN IBUS <14>AND <15>
3826  014336  000013                  TRAP    C$ERDF
3827  014340  005055                  .WORD   11
3828  014342  007276                  .WORD   EM0
3829                                  .WORD   ERR11
3830                                                  ;EXPECTED (R4) IS COMBINATION OF
3831  014344                                          ;IBUS* <14> AND <15>
3832  014344                  4$:
3833  014344  104410          ESCAPE  TST
3834  014346  000120          TRAP    C$ESCAPE
                                  .WORD   L10053-.
```

```
3835  014350                          BREAK
3836  014350  104422                  TRAP    C$BRK
3837  014352  032702  010000          BIT     #10000,R2       ;DONE YET?
3838  014356  001720                  BEQ     2$
3839                                  :*
3840                                  :*THIS SECTION OF CODE ADDED TO MAKE SURE
3841                                  :*THAT MASTER CLEAR, CLEARS THE MAR
3842                                  :*
3843
3844  014360                          SKIP06  40$
3845                                  ;GOTO 40$ IF M8206
3846  014370  005737  002470          TST     RUNB
3847  014374  001034                  BNE     40$
3848  014376  005737  002472          TST     RUNINH
3849  014402  001031                  BNE     40$
3850  014404  052711  040000          BIS     #40000,(R1)     ;SET MASTER CLEAR
3851  014410  005011                  CLR     (R1)            ;CLEAR MASTER CLEAR
3852  014412                          ROMCLK                  ;WE MUST FIRST CLOCK
3853  014412  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3854  014416  121325                  121325                  ;THE MAR LATCH REGS
3855  014420                          ROMCLK                  ;BEFORE WE CAN READ THEM
3856  014420  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3857  014424  121304                  121304
3858  014426                          ROMCLK                  ;READ IBUS* <15> PUT IN PORT5
3859  014426  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3860  014432  121325                  121325                  ;MAR HIGH
3861  014434                          ROMCLK                  ;READ IBUS* <14>, PUT IN PORT4
3862  014434  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
3863  014440  121304                  121304                  ;MAR LOW
3864  014442  005002                  CLR     R2              ;EXPECT MAR CLEAR
3865  014444  016104  000004          MOV     4(R1),R4        ;READ PORTS 4&5. THEY CONTAIN
3866                                                          ;THE CONTENTS OF THE MAR
3867                                                          ;MASTER CLEAR SHOULD HAVE
3868                                                          ;CLEARED THE MAR
3869                                                          ;BRANCH END TST IF CLEAR
3870  014450  001406                  BEQ     40$
3871  014452                          ERROR   44
3872  014456  104455                  TRAP    C$ERDF
3873  014460  000054                  .WORD   44
3874  014462  005055                  .WORD   EM0
3875  014464  011054                  .WORD   ERR44
3876  014466                  40$:
3877  014466                  ENDTST
3878  014466                  L10053:
3879  014466  104401                  TRAP    C$ETST
3880
3881  014470                          BADHEAD
3882                                  ;*************** TEST 10 ***************
3883                                  ;*IOP (CRAM) ODT BITS TEST
3884                                  ;*LOAD MAR WITH A 0 INC MAR UNTIL IT OVERFLOWS
3885                                  ;*VERIFY THAT IBUS* 10 BITS IS SET ONLY WHEN MAR BIT 8 IS A ONE
3886                                  ;*AND THAT IBUS* 10 BIT6 IS SET ON MAR OVERFLOW
3887  014470                          BADHEAD
3888                                  ;*************** TEST 10 ***************
3889
3890  014470                  BGNTST
```

```
3891  014470
3892  014470
3893
3894  014476  104432
3895  014500  000234
3896  014502
3897  014502  013701  002516
3898
3899  014506
3900  014512  005002
3901  014514
3902  014514  004537  003044
3903  014520  010000
3904  014522
3905  014522
3906  014522  004537  003044
3907  014526  121204
3908  014530  005005
3909  014532  032702  000400
3910  014536  001402
3911  014540  012705  000040
3912  014544  016104  000004
3913  014550  042704  177637
3914  014554  020504
3915  014556  001410
3916  014560
3917  014564  104455
3918  014566  000007
3919  014570  005055
3920  014572  007026
3921  014574
3922  014574  104410
3923  014576  000136
3924  014600
3925  014600
3926  014600  004537  003044
3927  014604  014000
3928  014606  005202
3929  014610  022702  002000
3930  014614  001342
3931  014616
3932  014616  004537  003044
3933  014622  121204
3934  014624  012705  000100
3935  014630  016104  000004
3936  014634  042704  177627
3937  014640  020504
3938  014642  001406
3939  014644
3940  014650  104455
3941  014652  000007
3942  014654  005055
3943  014656  007026
3944  014660
3945  014660  004537  003044
3946  014664  010000
```

```
T10::
        MACEX
        ;DO NOT DO TEST IF M8200
        TRAP    C$EXIT
        .WORD   L10054-.
        MYINT
        MOV     KMCSR,R1            ;RECORD DEVICE ADDR.
                                    ;R1 CONTAINS BASE M8200,4,7 ADDRESS
        MSTCLR                      ;MASTER CLEAR M8200,4,7
        CLR     R2                  ;R2=SAME AS MAR CONTENTS
        ROMCLK                      ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
        010000                      ;MAR_0
1$:
        ROMCLK                      ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
        121204                      ;PORT4=IBUS*10
        CLR     R5                  ;R5='EXPECTED'
        BIT     #BIT8,R2            ;IS BIT8 SET IN MAR?
        BEQ     .+6                 ;BR IF NO
        MOV     #BIT5,R5            ;IF YES THEN SET BIT5
        MOV     4(R1),R4            ;R4='FOUND'
        BIC     #177637,R4          ;CLEAR UNWANTED BITS
        CMP     R5,R4               ;BITS 5&6 SHOULD BE CLEAR.
        BEQ     15$                 ;BR IF OK
        ERROR   7                   ;ERROR BITS 5&6 NOT CLEAR
        TRAP    C$ERDF
        .WORD   7
        .WORD   EM0
        .WORD   ERR7
        ESCAPE  TST
        TRAP    C$ESCAPE
        .WORD   L10054-.
15$:
        ROMCLK                      ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
        014000                      ;INC MAR
        INC     R2                  ;BUMP MEM ADDRESS
        CMP     #2000,R2            ;OVERFLOWED YET?(OVFL PAGE BITS).
        BNE     1$                  ;BR IF NO
        ROMCLK                      ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
        121204                      ;PART4 IBUS* 10
        MOV     #BIT6,R5            ;R5='EXPECTED'
        MOV     4(R1),R4            ;R4='FOUND'
        BIC     #177627,R4          ;CLEAR UNWANTED BITS
        CMP     R5,R4               ;BIT6 SHOULD BE SET
        BEQ     17$                 ;BR IF OK
        ERROR   7                   ;ERROR, BIT6 NOT SET
        TRAP    C$ERDF
        .WORD   7
        .WORD   EM0
        .WORD   ERR7
17$:    ROMCLK                      ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
        010000                      ;MAR_0
```

```
3947  014666
3948  014666  004537  003044          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
3949  014672  004000          JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
3950  014674                  004000                          ;MAR HI 0
3951  014674  004537  003044          ROMCLK                  ;NEXT WORD IS INSTRUCTION,
3952  014700  121204          JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
3953  014702  005005          121204                          ;PORT4 IBUS* 10
3954  014704  016104  000004  CLR     R5                      ;R5='EXPECTED'
3955  014710  042704  177637  MOV     4(R1),R4                ;R4='FOUND'
3956  014714  020504          BIC     #177637,R4              ;CLEAR UNWANTED BITS
3957  014716  001406          CMP     R5,R4                   ;BITS 5&6 SHOULD BE CLEAR
3958  014720                  BEQ     2$                      ;BR IF OK
3959  014724  104455          ERROR   7                       ;ERROR 5&6 NOT BOTH CLEAR
3960  014726  000007          TRAP    C$ERDF
3961  014730  005055          .WORD   7
3962  014732  007026          .WORD   EM0
3963  014734                  .WORD   ERR7
3964  014734                  2$:
3965  014734                  ENDTST
3966  014734  104401          L10054:
3967                          TRAP    C$ETST
3968  014736
3969                          BADHEAD
3970                          ;*************** TEST 11 ***************
3971                          ;*CRAM TEST OF JUMP(I) NEVER MICRO-PROCESSOR INSTRUCTION.
3972                          ;*PERFORM THE JUMP INSTRUCTION
3973                          ;*VERIFY THE JUMP DID NOT OCCUR BY CLOCKING THE INSTRUCTION
3974                          ;*IN THE LOCATION IT IS AT.  THIS INSTRUCTION LOADS THE
3975                          ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
3976                          ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
3977                          ;*THE CRAM PC IS CORRECT, IF THE CRAM PC IS NOT RIGHT,
3978  014736                  ;*THEN PORT4 CONTAINS A 37
3979                          BADHEAD
3980                          ;*************** TEST 11 ***************
3981  014736                  BGNTST
3982  014736                  T11::
3983  014736                          SKIP04  10$
3984                          ;GOTO 10$ IF M8204
3985  014746                  EXIT    TST                     ;CAN'T DO IF ROM,4K
3986  014746  104432          TRAP    C$EXIT
3987  014750  000230          .WORD   L10055-.
3988  014752                  10$:
3989  014752                  MYINT
3990  014752  013701  002516  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
3991                                                  ;R1 CONTAINS BASE M8200,4,7 ADDRESS
3992  014756                  MSTCLR                  ;MASTER CLEAR M8200,4,7
3993  014762                  BGNSEG
3994  014762  104404          TRAP    C$BSEG
3995  014764  004737  003474  JSR     PC,MEMSET       ;SET MEM AND RAM
3996  014770                  1$:
3997  014770  004737  003166  JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
3998  014774                  SROMCLK                 ;NEXT WORD IS INSTRUCTION,
3999  014774  004537  003100  JSR     R5,..SROMCLK
4000  015000  100400          100400                  ;START AT ROM PC=0
4001  015002                  SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4002  015002  004537  003100  JSR     R5,..SROMCLK
```

```
4003  015006  114377                    114377!<400*0>          ;JUMP TO ROM PC OF 1777
4004  015010  004737  003330            JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4005  015014  000001                    1                       ;EXPECTED DATA
4006  015016  120504                    CMPB    R5,R4           ;IS ROM PC CORRECT?
4007  015020  001406                    BEQ     2$              ;BR IF NO
4008  015022                            ERROR   5               ;ERROR, CRAM PC IS WRONG
4009  015026  104455                    TRAP    C$ERDF
4010  015030  000005                    .WORD   5
4011  015032  005055                    .WORD   EM0
4012  015034  006556                    .WORD   ERR5
4013  015036                    2$:     ESCAPE SEG
4014  015036  104410                    TRAP    C$ESCAPE
4015  015040  000002                    .WORD   10000$-.
4016  015042                            ENDSEG
4017  015042                    10000$:
4018  015042  104405                    TRAP    C$ESEG
4019  015044                            BGNSEG
4020  015044  104404                    TRAP    C$BSEG
4021  015046  004737  003166            JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
4022  015052                            SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4023  015052  004537  003100            JSR     R5,..SROMCLK
4024  015056  100403                    100403                  ;START AT ROM PC=3
4025  015060                            SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4026  015060  004537  003100            JSR     R5,..SROMCLK
4027  015064  100000                    100000!<400*0>  ;JUMP TO ROM PC OF 0
4028  015066  004737  003330            JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4029  015072  000004                    4                       ;EXPECTED DATA
4030  015074  120504                    CMPB    R5,R4           ;IS ROM PC CORRECT?
4031  015076  001406                    BEQ     4$              ;BR IF YES
4032  015100                            ERROR   5               ;ERROR, CROM PC IS WRONG
4033  015104  104455                    TRAP    C$ERDF
4034  015106  000005                    .WORD   5
4035  015110  005055                    .WORD   EM0
4036  015112  006556                    .WORD   ERR5
4037  015114                    4$:     ESCAPE SEG
4038  015114  104410                    TRAP    C$ESCAPE
4039  015116  000002                    .WORD   10001$-.
4040  015120                            ENDSEG
4041  015120                    10001$:
4042  015120  104405                    TRAP    C$ESEG
4043  015122                            BGNSEG
4044  015122  104404                    TRAP    C$BSEG
4045  015124  004737  003166            JSR     PC,CLRALL       ;CLEAR ALL CONDITINS
4046  015130                            SROMCLK                 ;NEXT WORD IS INSTRUCION,
4047  015130  004537  003100            JSR     R5,..SROMCLK
4048  015134  100406                    100406                  ;START AT ROM PC=6
4049  015136                            SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4050  015136  004537  003100            JSR     R5,.SROMCLK
4051  015142  104125                    104125!<400*0>          ;JUMP TO ROM PC OF 525
4052  015144  004737  003330            JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4053  015150  000007                    7                       ;EXPECTED DATA
4054  015152  120504                    CMPB    R5,R4           ;IS ROM PC CORRECT?
4055  015154  001406                    BEQ     6$              ;BR IF YES
4056  015156                            ERROR   5               ;ERROR, CRAM PC IS WRONG
4057  015162  104455                    TRAP    C$ERDF
4058  015164  000005                    .WORD   5
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
4059  015166  005055                .WORD   EM0
4060  015170  006556                .WORD   ERR5
4061  015172                6$:     ESCAPE SEG
4062  015172  104410                TRAP    C$ESCAPE
4063  015174  000002                .WORD   10002$-.
4064  015176                        ENDSEG
4065  015176                10002$:
4066  015176  104405                TRAP    C$ESEG
4067  015200                ENDTST
4068  015200                L10055:
4069  015200  104401                TRAP    C$ETST
4070
4071  015202                        BADHEAD
4072                                ;*************** TEST 12 ***************
4073                                ;*CRAM TEST OF JUMP(I) ALWAYS MICRO-PROCESSOR INSTRUCTION.
4074                                ;*PERFORM THE JUMP INSTRUCTION
4075                                ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
4076                                ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
4077                                ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
4078                                ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
4079                                ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
4080                                ;*THEN PORT4 WILL CONTAIN A 37
4081  015202                        BADHEAD
4082                                ;*************** TEST 12 ***************
4083
4084  015202                BGNTST
4085  015202                T12::
4086  015202                        MACEX2                  ;DON'T DO IF M8200
4087                                ;DO NOT DO TEST IF M8200
4088  015212  104432                TRAP    C$EXIT
4089  015214  000214                .WORD   L10056-.
4090  015216                        MYINT
4091  015216  013701  002516        MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
4092                                                        ;R1 CONTAINS BASE M8200,4,7 ADDRESS
4093  015222                        MSTCLR                  ;MASTER CLEAR M8200,4,7
4094  015226  004737  003474        JSR     PC,MEMSET       ;SET MEM AND RAM
4095  015232                1$:     BGNSEG
4096  015232  104404                TRAP    C$BSEG
4097  015234                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4098  015234  004537  003100        JSR     R5,..SROMCLK
4099  015240  100400                100400                  ;START AT ROM PC=0
4100  015242                        SROMCLK                 ;NEXT WORD IS INSTRUCION,
4101  015242  004537  003100        JSR     R5,..SROMCLK
4102  015246  114777                114377!<400*1>          ;JUMP TO ROM PC OF 1777
4103  015250  004737  003330        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4104  015254  000377                377                     ;EXPECTED DATA
4105  015256  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
4106  015260  001406                BEQ     2$              ;BR IF YES
4107  015262                        ERROR   5               ;ERROR, CRAM PC IS WRONG
4108  015266  104455                TRAP    C$ERDF
4109  015270  000005                .WORD   5
4110  015272  005055                .WORD   EM0
4111  015274  006556                .WORD   ERR5
4112  015276                2$:     ESCAPE SEG
4113  015276  104410                TRAP    C$ESCAPE
4114  015300  000002                .WORD   10000$-.
```

```
                                HARDWARE TESTS

                                        ENDSEG
                                10000$:
                                        TRAP    C$ESEG
                                        BGNSEG
                                        TRAP    C$BSEG
                                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
                                        JSR     R5,..SROMCLK
                                        100403                  ;START AT ROM PC=3
                                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
                                        JSR     R5,..SROMCLK
                                        100000!<400*1>  ;JUMP TO ROM PC OF 0
                                        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
                                        0                       ;EXPECTED DATA
                                        CMPB    R5,R4           ;IS ROM PC CORRECT?
                                        BEQ     4$              ;BR IF YES
                                        ERROR   5       ;ERROR, CRAM PC IS WRONG
                                        TRAP    C$ERDF
                                        .WORD   5
                                        .WORD   EM0
                                        .WORD   ERR5
                                4$:     ESCAPE SEG
                                        TRAP    C$ESCAPE
                                        .WORD   10001$-.
                                        ENDSEG
                                10001$:
                                        TRAP    C$ESEG
                                        BGNSEG
                                        TRAP    C$BSEG
                                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
                                        JSR     R5,..SROMCLK
                                        100406                  ;START AT ROM PC=6
                                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
                                        JSR     R5,..SROMCLK
                                        104125!<400*1>          ;JUMP TO ROM PC OF 525
                                        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
                                        125                     ;EXPECTED DATA
                                        CMPB    R5,R4           ;IS ROM PC CORRECT?
                                        BEQ     6$              ;BR IF YES
                                        ERROR   5               ;ERROR, CRAM PC IS WRONG
                                        TRAP    C$ERDF
                                        .WORD   5
                                        .WORD   EM0
                                        .WORD   ERR5
                                6$:     ESCAPE SEG
                                        TRAP    C$ESCAPE
                                        .WORD   10002$-.
                                        ENDSEG
                                10002$:
                                        TRAP    C$ESEG

                                ENDTST
                                L10056:
                                        TRAP    C$ETST

                                        BADHEAD
                                        ;*************** TEST 13 ***************
                                        ;*CRAM TEST OF JUMP(I) ON C BIT SET MICRO-PROCESSOR INSTRUCTION.
```

```
4171
4172
4173
4174
4175
4176
4177
4178  015432                        ;*SET THE C BIT, PERFORM THE JUMP INSTRUCTION.
                                     ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
                                     ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
                                     ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
                                     ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
                                     ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
                                     ;*THEN PORT4 WILL CONTAIN A 37
                                     BADHEAD
                                     ;*************** TEST 13 ***************
4179
4180
4181  015432                         BGNTST
4182  015432                         T13::
4183  015432                         MACEX2                    ;DON'T DO IF M8200
4184
4185  015442  104432                 ;DO NOT DO TEST IF M8200
4186  015444  000230                 TRAP    C$EXIT
4187  015446                         .WORD   L10057-.
4188  015446  013701  002516         MYINT
4189                                 MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
4190  015452                                                         ;R1 CONTAINS BASE M8200,4,7 ADDRESS
4191  015456  004737  003474         MSTCLR                          ;MASTER CLEAR M8200,4,7
4192  015462                         JSR     PC,MEMSET               ;SET MEM AND RAM
4193  015462  104404                 1$:     BGNSEG
4194  015464  004737  003260         TRAP    C$BSEG
4195  015470                         JSR     PC,SETC                 ;SET THE C BIT'
4196  015470  004537  003100         SROMCLK                         ;NEXT WORD IS INSTRUCTION,
4197  015474  100400                 JSR     R5,..SROMCLK
4198  015476                         100400                          ;START AT ROM PC=0
4199  015476  004537  003100         SROMCLK                         ;NEXT WORD IS INSTRUCION,
4200  015502  115377                 JSR     R5,..SROMCLK
4201  015504  004737  003330         114377!<400*2>                  ;JUMP TO ROM PC OF 1777
4202  015510  000377                 JSR     PC,RAMDAT               ;R4=CRAM PC (LSB 8 BITS)
4203  015512  120504                 377                             ;EXPECTED DATA
4204  015514  001406                 CMPB    R5,R4                   ;IS ROM PC CORRECT?
4205  015516                         BEQ     2$                      ;BR IF YES
4206  015522  104455                 ERROR   5                       ;ERROR, CRAM PC IS WRONG
4207  015524  000005                 TRAP    C$ERDF
4208  015526  005055                 .WORD   5
4209  015530  006556                 .WORD   EM0
4210  015532                         .WORD   ERR5
4211  015532                         2$:                             ;LOOP TO 1$ IF SW09=1
4212  015532  104410                 ESCAPE SEG
4213  015534  000002                 TRAP    C$ESCAPE
4214  015536                         .WORD   10000$-.
4215  015536                         ENDSEG
4216  015536  104405                 10000$:
4217  015540                         TRAP    C$ESEG
4218  015540  104404                 BGNSEG
4219  015542  004737  003260         TRAP    C$BSEG
4220  015546                         JSR     PC,SETC                 ;SET THE C BIT'
4221  015546  004537  003100         SROMCLK                         ;NEXT WORD IS INSTRUCTION,
4222  015552  100403                 JSR     R5,..SROMCLK
4223  015554                         100403                          ;START AT ROM PC=3
4224  015554  004537  003100         SROMCLK                         ;NEXT WORD IS INSTRUCTION,
4225  015560  101000                 JSR     R5,..SROMCLK
4226  015562  004737  003330         100000!<400*2>                  ;JUMP TO ROM PC OF 0
                                     JSR     PC,RAMDAT               ;R4=CRAM PC (LSB 8 BITS)
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
4227  015566  000000                    0                       ;EXPECTED DATA
4228  015570  120504                    CMPB    R5,R4           ;IS ROM PC CORRECT?
4229  015572  001406                    BEQ     4$              ;BR IF YES
4230  015574                            ERROR   5               ;ERROR, CRAM PC IS WRONG
4231  015600  104455                    TRAP    C$ERDF
4232  015602  000005                    .WORD   5
4233  015604  005055                    .WORD   EM0
4234  015606  006556                    .WORD   ERR5
4235  015610                    4$:     ESCAPE SEG              ;LOOP TO 3$ IF SW09=1
4236  015610
4237  015610  104410                    TRAP    C$ESCAPE
4238  015612  000002                    .WORD   10001$-.
4239  015614                            ENDSEG
4240  015614                    10001$:
4241  015614  104405                    TRAP    C$ESEG
4242  015616                            BGNSEG
4243  015616  104404                    TRAP    C$BSEG
4244  015620  004737  003260            JSR     PC,SETC         ;SET THE C BIT'
4245  015624                            SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4246  015624  004537  003100            JSR     R5,..SROMCLK
4247  015630  100406                    100406                  ;START AT ROM PC=6
4248  015632                            SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4249  015632  004537  003100            JSR     R5,..SROMCLK
4250  015636  105125                    104125!<400*2>          ;JUMP TO ROM PC OF 525
4251  015640  004737  003330            JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4252  015644  000125                    125                     ;EXPECTED DATA
4253  015646  120504                    CMPB    R5,R4           ;IS ROM PC CORRECT?
4254  015650  001406                    BEQ     6$              ;BR IF YES
4255  015652                            ERROR   5               ;ERROR, CRAM PC IS WRONG
4256  015656  104455                    TRAP    C$ERDF
4257  015660  000005                    .WORD   5
4258  015662  005055                    .WORD   EM0
4259  015664  006556                    .WORD   ERR5
4260  015666                    6$:     ESCAPE SEG
4261  015666  104410                    TRAP    C$ESCAPE
4262  015670  000002                    .WORD   10002$-.
4263  015672                            ENDSEG
4264  015672                    10002$:
4265  015672  104405                    TRAP    C$ESEG
4266  015674                    ENDTST
4267  015674                    L10057:
4268  015674  104401                    TRAP    C$ETST
4269
4270  015676                            BADHEAD
4271                                    ;*************** TEST 14 ***************
4272                                    ;*CRAM TEST OF JUMP(I) ON Z BIT SET MICRO-PROCESSOR INSTRUCTION.
4273                                    ;*SET THE Z BIT, PERFORM THE JUMP INSTRUCTION.
4274                                    ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
4275                                    ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
4276                                    ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
4277                                    ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
4278                                    ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
4279                                    ;*THEN PORT4 WILL CONTAIN A 37
4280  015676                            BADHEAD
4281                                    ;*************** TEST 14 ***************
4282
```

```
4283  015676                          BGNTST
4284  015676                          T14::
4285  015676                                MACEX2                  ;DON'T DO IF M8200.
4286                                        ;DO NOT DO TEST IF M8200
4287  015706  104432                       TRAP    C$EXIT
4288  015710  000230                       .WORD   L10060-.
4289  015712                          MYINT
4290  015712  013701  002516                MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
4291                                                                ;R1 CONTAINS BASE M8200,4,7 ADDRESS
4292  015716                                MSTCLR                  ;MASTER CLEAR M8200,4,7
4293  015722  004737  003474                JSR     PC,MEMSET       ;SET MEM AND RAM
4294  015726                          1$:   BGNSEG
4295  015726  104404                       TRAP    C$BSEG
4296  015730  004737  003312                JSR     PC,SETZ         ;SET THE Z BIT'
4297  015734                                SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4298  015734  004537  003100                JSR     R5,..SROMCLK
4299  015740  100400                       100400                  ;START AT ROM PC=0
4300  015742                                SROMCLK                 ;NEXT WORD IS INSTRUCION,
4301  015742  004537  003100                JSR     R5,..SROMCLK
4302  015746  115777                       114377!<400*3>          ;JUMP TO ROM PC OF 1777
4303  015750  004737  003330                JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4304  015754  000377                       377                     ;EXPECTED DATA
4305  015756  120504                       CMPB    R5,R4           ;IS ROM PC CORRECT?
4306  015760  001406                       BEQ     2$              ;BR IF YES
4307  015762                                ERROR   5               ;ERROR, CRAM PC IS WRONG
4308  015766  104455                       TRAP    C$ERDF
4309  015770  000005                       .WORD   5
4310  015772  005055                       .WORD   EM0
4311  015774  006556                       .WORD   ERR5
4312  015776                          2$:   ESCAPE SEG
4313  015776  104410                       TRAP    C$ESCAPE
4314  016000  000002                       .WORD   10000$-.
4315  016002                                ENDSEG
4316  016002                          10000$:
4317  016002  104405                       TRAP    C$ESEG
4318  016004                                BGNSEG
4319  016004  104404                       TRAP    C$BSEG
4320  016006  004737  003312                JSR     PC,SETZ         ;SET THE Z BIT'
4321  016012                                SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4322  016012  004537  003100                JSR     R5,..SROMCLK
4323  016016  100403                       100403                  ;START AT ROM PC=3
4324  016020                                SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4325  016020  004537  003100                JSR     R5,..SROMCLK
4326  016024  101400                       100000!<400*3>          ;JUMP TO ROM PC OF 0
4327  016026  004737  003330                JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4328  016032  000000                       0                       ;EXPECTED DATA
4329  016034  120504                       CMPB    R5,R4           ;IS ROM PC CORRECT?
4330  016036  001406                       BEQ     4$              ;BR IF YES
4331  016040                                ERROR   5               ;ERROR, CRAM PC IS WRONG
4332  016044  104455                       TRAP    C$ERDF
4333  016046  000005                       .WORD   5
4334  016050  005055                       .WORD   EM0
4335  016052  006556                       .WORD   ERR5
4336  016054                          4$:   ESCAPE SEG
4337  016054  104410                       TRAP    C$ESCAPE
4338  016056  000002                       .WORD   10001$-.
```

```
016060                          ENDSEG
016060                  10001$:
016060  104405                  TRAP    C$ESEG
016062                          BGNSEG
016062  104404                  TRAP    C$BSEG
016064  004737  003312          JSR     PC,SETZ                 ;SET THE Z BIT'
016070                          SROMCLK                         ;NEXT WORD IS INSTRUCTION,
016070  004537  003100          JSR     R5,.SROMCLK
016074  100406                  100406                          ;START AT ROM PC=6
016076                          SROMCLK                         ;NEXT WORD IS INSTRUCTION,
016076  004537  003100          JSR     R5,.SROMCLK
016102  105525                  104125!<400*3>                  ;JUMP TO ROM PC OF 525
016104  004737  003330          JSR     PC,RAMDAT               ;R4=CRAM PC (LSB 8 BITS)
016110  000125                  125                             ;EXPECTED DATA
016112  120504                  CMPB    R5,R4                   ;IS ROM PC CORRECT?
016114  001406                  BEQ     6$                      ;BR IF YES
016116                          ERROR   5                       ;ERROR, CRAM PC IS WRONG
016122  104455                  TRAP    C$ERDF
016124  000005                  .WORD   5
016126  005055                  .WORD   EMO
016130  006556                  .WORD   ERR5
016132                  6$:     ESCAPE SEG
016132  104410                  TRAP    C$ESCAPE
016134  000002                  .WORD   10002$-.
016136                          ENDSEG
016136                  10002$:
016136  104405                  TRAP    C$ESEG
016140                  ENDTST
016140                  L10060:
016140  104401                  TRAP    C$ETST

016142                          BADHEAD
                                ;*************** TEST 15 ****************
                                ;*CRAM TEST OF JUMP(I) ON BRO SET MICRO-PROCESSOR INSTRUCTION.
                                ;*SET THE BRO BIT, PERFORM THE JUMP INSTRUCTION.
                                ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
                                ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
                                ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
                                ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
                                ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
                                ;*THEN PORT4 WILL CONTAIN A 37
016142                          BADHEAD
                                ;*************** TEST 15 ****************

016142                  BGNTST
016142                  T15::
016142                          MACEX2                          ;DON'T DO IF M8200.
                                ;DO NOT DO TEST IF M8200
016152  104432                  TRAP    C$EXIT
016154  000230                  .WORD   L10061-.
016156                          MYINT
016156  013701  002516          MOV     KMCSR,R1                        ;RECORD DEVICE ADDR.
                                                                ;R1 CONTAINS BASE M8200,4,7 ADDRESS
016162                          MSTCLR                          ;MASTER CLEAR M8200,4,7
016166  004737  003474          JSR     PC,MEMSET               ;SET MEM AND RAM
016172                  1$:
```

```
4395  016172
4396  016172  104404
4397  016174  004737  003220
4398  016200
4399  016200  004537  003100
4400  016204  100400
4401  016206
4402  016206  004537  003100
4403  016212  116377
4404  016214  004737  003330
4405  016220  000377
4406  016222  120504
4407  016224  001406
4408  016226
4409  016232  104455
4410  016234  000005
4411  016236  005055
4412  016240  006556
4413  016242
4414  016242  104410
4415  016244  000002
4416  016246
4417  016246
4418  016246  104405
4419  016250
4420  016250  104404
4421  016252  004737  003220
4422  016256
4423  016256  004537  003100
4424  016262  100403
4425  016264
4426  016264  004537  003100
4427  016270  102000
4428  016272  004737  003330
4429  016276  000000
4430  016300  120504
4431  016302  001406
4432  016304
4433  016310  104455
4434  016312  000005
4435  016314  005055
4436  016316  006556
4437  016320
4438  016320  104410
4439  016322  000002
4440  016324
4441  016324
4442  016324  104405
4443  016326
4444  016326  104404
4445  016330  004737  003220
4446  016334
4447  016334  004537  003100
4448  016340  100406
4449  016342
4450  016342  004537  003100
```

```
        BGNSEG
        TRAP    C$BSEG
        JSR     PC,SETBRO       ;SET THE BRO BIT'
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100400                  ;START AT ROM PC=0
        SROMCLK                 ;NEXT WORD IS INSTRUCION,
        JSR     R5,..SROMCLK
        114377!<400*4>          ;JUMP TO ROM PC OF 1777
        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
        377                     ;EXPECTED DATA
        CMPB    R5,R4           ;IS ROM PC CORRECT?
        BEQ     2$              ;BR IF YES
        ERROR   5               ;ERROR, CRAM PC IS WRONG
        TRAP    C$ERDF
        .WORD   5
        .WORD   EM0
        .WORD   ERR5
2$:     ESCAPE  SEG
        TRAP    C$ESCAPE
        .WORD   10000$-.
        ENDSEG
10000$:
        TRAP    C$ESEG
        BGNSEG
        TRAP    C$BSEG
        JSR     PC,SETBRO       ;SET THE BRO BIT'
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100403                  ;START AT ROM PC=3
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100000!<400*4>          ;JUMP TO ROM PC OF 0
        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
        0                       ;EXPECTED DATA
        CMPB    R5,R4           ;IS ROM PC CORRECT?
        BEQ     4$              ;BR IF YES
        ERROR   5               ;ERROR, CRAM PC IS WRONG
        TRAP    C$ERDF
        .WORD   5
        .WORD   EM0
        .WORD   ERR5
4$:     ESCAPE  SEG
        TRAP    C$ESCAPE
        .WORD   10001$-.
        ENDSEG
10001$:
        TRAP    C$ESEG
        BGNSEG
        TRAP    C$BSEG
        JSR     PC,SETBRO       ;SET THE BRO BIT'
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100406                  ;START AT ROM PC=6
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
4451  016346  106125
4452  016350  004737  003330          104125!<400*4>          ;JUMP TO ROM PC OF 525
4453  016354  000125                  JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4454  016356  120504                  125                     ;EXPECTED DATA
4455  016360  001406                  CMPB    R5,R4           ;IS ROM PC CORRECT?
4456  016362                          BEQ     6$              ;BR IF YES
4457  016366  104455                  ERROR   5               ;ERROR, CRAM PC IS WRONG
4458  016370  000005                  TRAP    C$ERDF
4459  016372  005055                  .WORD   5
4460  016374  006556                  .WORD   EM0
4461  016376                          .WORD   ERR5
4462  016376  104410          6$:     ESCAPE  SEG
4463  016400  000002                  TRAP    C$ESCAPE
4464  016402                          .WORD   10002$-.
4465  016402                          ENDSEG
4466  016402  104405          10002$:
4467  016404                          TRAP    C$ESEG
4468  016404                  ENDTST
4469  016404  104401          L10061:
4470                                  TRAP    C$ETST
4471  016406
4472                                  BADHEAD
4473                                  ;*************** TEST 16 ***************
4474                                  ;*CRAM TEST OF JUMP(I) ON BR1 SET MICRO-PROCESSOR INSTRUCTION.
4475                                  ;*SET THE BR1 BIT, PERFORM THE JUMP INSTRUCTION.
4476                                  ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
4477                                  ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
4478                                  ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
4479                                  ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
4480                                  ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
4481  016406                          ;*THEN PORT4 WILL CONTAIN A 37
4482                                  BADHEAD
4483                                  ;*************** TEST 16 ***************
4484  016406          BGNTST
4485  016406          T16::
4486  016406                          MACEX2                  ;DON'T DO IF M8200.
4487                                  ;DO NOT DO TEST IF M8200
4488  016416  104432                  TRAP    C$EXIT
4489  016420  000230                  .WORD   L10062-.
4490  016422                          MYINT
4491  016422  013701  002516          MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
4492                                                          ;R1 CONTAINS BASE M8200,4,7 ADDRESS
4493  016426                          MSTCLR                  ;MASTER CLEAR M8200,4,7
4494  016432  004737  003474          JSR     PC,MEMSET       ;SET MEM AND RAM
4495  016436          1$:
4496  016436                          BGNSEG
4497  016436  104404                  TRAP    C$BSEG
4498  016440  004737  003230          JSR     PC,SETBR1       ;SET THE BR1 BIT'
4499  016444                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4500  016444  004537  003100          JSR     R5,..SROMCLK
4501  016450  100400                  100400                  ;START AT ROM PC=0
4502  016452                          SROMCLK                 ;NEXT WORD IS INSTRUCION,
4503  016452  004537  003100          JSR     R5,..SROMCLK
4504  016456  116777                  114377!<400*5>          ;JUMP TO ROM PC OF 1777
4505  016460  004737  003330          JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4506  016464  000377                  377                     ;EXPECTED DATA
```

```
4507  016466  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
4508  016470  001406                BEQ     2$              ;BR IF YES
4509  016472                        ERROR   5               ;ERROR, CRAM PC IS WRONG
4510  016476  104455                TRAP    C$ERDF
4511  016500  000005                .WORD   5
4512  016502  005055                .WORD   EM0
4513  016504  006556                .WORD   ERR5
4514  016506                2$:     ESCAPE  SEG
4515  016506  104410                TRAP    C$ESCAPE
4516  016510  000002                .WORD   10000$-.
4517  016512                        ENDSEG
4518  016512                10000$:
4519  016512  104405                TRAP    C$ESEG
4520  016514                        BGNSEG
4521  016514  104404                TRAP    C$BSEG
4522  016516  004737  003230        JSR     PC,SETBR1               ;SET THE BR1 BIT'
4523  016522                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4524  016522  004537  003100        JSR     R5,..SROMCLK
4525  016526  100403                100403                  ;START AT ROM PC=3
4526  016530                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4527  016530  004537  003100        JSR     R5,..SROMCLK
4528  016534  102400                100000!<400*5>          ;JUMP TO ROM PC OF 0
4529  016536  004737  003330        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4530  016542  000000                0                       ;EXPECTED DATA
4531  016544  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
4532  016546  001406                BEQ     4$              ;BR IF YES
4533  016550                        ERROR   5               ;ERROR, CRAM PC IS WRONG
4534  016554  104455                TRAP    C$ERDF
4535  016556  000005                .WORD   5
4536  016560  005055                .WORD   EM0
4537  016562  006556                .WORD   ERR5
4538  016564                4$:     ESCAPE  SEG
4539  016564  104410                TRAP    C$ESCAPE
4540  016566  000002                .WORD   10001$-.
4541  016570                        ENDSEG
4542  016570                10001$:
4543  016570  104405                TRAP    C$ESEG
4544  016572                        BGNSEG
4545  016572  104404                TRAP    C$BSEG
4546  016574  004737  003230        JSR     PC,SETBR1               ;SET THE BR1 BIT'
4547  016600                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4548  016600  004537  003100        JSR     R5,..SROMCLK
4549  016604  100406                100406                  ;START AT ROM PC=6
4550  016606                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4551  016606  004537  003100        JSR     R5,..SROMCLK
4552  016612  106525                104125!<400*5>          ;JUMP TO ROM PC OF 525
4553  016614  004737  003330        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4554  016620  000125                125                     ;EXPECTED DATA
4555  016622  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
4556  016624  001406                BEQ     6$              ;BR IF YES
4557  016626                        ERROR   5               ;ERROR, CRAM PC IS WRONG
4558  016632  104455                TRAP    C$ERDF
4559  016634  000005                .WORD   5
4560  016636  005055                .WORD   EM0
4561  016640  006556                .WORD   ERR5
4562  016642                6$:     ESCAPE  SEG
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS

 4563  016642  104410                          TRAP    C$ESCAPE
 4564  016644  000002                          .WORD   10002$-.
 4565  016646                                  ENDSEG
 4566  016646                          10002$:
 4567  016646  104405                          TRAP    C$ESEG
 4568  016650                          ENDTST
 4569  016650                          L10062;
 4570  016650  104401                          TRAP    C$ETST
 4571
 4572  016652                                  BADHEAD
 4573                                          ;*************** TEST 17 ****************
 4574                                          ;*CRAM TEST OF JUMP(I) ON BR4 SET MICRO-PROCESSOR INSTRUCTION.
 4575                                          ;*SET THE BR4 BIT, PERFORM THE JUMP INSTRUCTION.
 4576                                          ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
 4577                                          ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
 4578                                          ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
 4579                                          ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
 4580                                          ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
 4581                                          ;*THEN PORT4 WILL CONTAIN A 37
 4582  016652                                  BADHEAD
 4583                                          ;*************** TEST 17 ****************
 4584
 4585  016652                          BGNTST
 4586  016652                          T17::
 4587  016652                                  MACEX2                  ;DON'T DO IF M8200.
 4588                                          ;DO NOT DO TEST IF M8200
 4589  016662  104432                          TRAP    C$EXIT
 4590  016664  000230                          .WORD   L10063-.
 4591  016666                                  MYINT
 4592  016666  013701  002516                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
 4593  016672                                  MSTCLR                  ;MASTER CLEAR M8200,4,7
 4594  016676  004737  003474                  JSR     PC,MEMSET       ;SET MEM AND RAM
 4595  016702                          1$:
 4596  016702                                  BGNSEG
 4597  016702  104404                          TRAP    C$BSEG
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
4598  016704  004737  003240          JSR     PC,SETBR4       ;SET THE BR4 BIT'
4599  016710                          SROMCLK                 ;NEXT WORD IS INSTRUCTION.
4600  016710  004537  003100          JSR     R5,..SROMCLK
4601  016714  100400                  100400                  ;START AT ROM PC=0
4602  016716                          SROMCLK                 ;NEXT WORD IS INSTRUCION.
4603  016716  004537  003100          JSR     R5,..SROMCLK
```

```
 4604  016722  117377                  114377!<400*6>          ;JUMP TO ROM PC OF 1777
 4605  016724  004737  003330          JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
```

```
 4606  016730  000377                    377                 ;EXPECTED DATA
 4607  016732  120504                    CMPB    R5,R4       ;IS ROM PC CORRECT?
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS
  4608  016734  001406                       BEQ     2$              ;BR IF YES
```

```
4609  016736
4610  016742  104455          ERROR   5               ;ERROR, CRAM PC IS WRONG
4611  016744  000005          TRAP    CSERDF
4612  016746  005055          .WORD   5
4613  016750  006556          .WORD   EM0
                              .WORD   ERR5
```

```
4614  016752
4615  016752  104410
4616  016754  000002
4617  016756
4618  016756
4619  016756  104405
4620  016760
4621  016760  104404
4622  016762  004737  003240
4623  016766
4624  016766  004537  003100
4625  016772  100403
4626  016774
4627  016774  004537  003100
4628  017000  103000
4629  017002  004737  003330
4630  017006  000000
4631  017010  120504
4632  017012  001406
4633  017014
4634  017020  104455
4635  017022  000005
4636  017024  005055
4637  017026  006556
4638  017030
4639  017030  104410
4640  017032  000002
4641  017034
4642  017034
4643  017034  104405
4644  017036
4645  017036  104404
4646  017040  004737  003240
4647  017044
4648  017044  004537  003100
4649  017050  100406
4650  017052
4651  017052  004537  003100
4652  017056  107125
4653  017060  004737  003330
4654  017064  000125
4655  017066  120504
4656  017070  001406
4657  017072
4658  017076  104455
4659  017100  000005
4660  017102  005055
4661  017104  006556
4662  017106
4663  017106  104410
4664  017110  000002
4665  017112
4666  017112
4667  017112  104405
4668  017114
4669  017114
```

```
2$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10000$-.
        ENDSEG
10000$:
        TRAP    C$ESEG
        BGNSEG
        TRAP    C$BSEG
        JSR     PC,SETBR4               ;SET THE BR4 BIT'
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100403                  ;START AT ROM PC=3
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100000!<400*6>          ;JUMP TO ROM PC OF 0
        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
        0                       ;EXPECTED DATA
        CMPB    R5,R4           ;IS ROM PC CORRECT?
        BEQ     4$              ;BR IF YES
        ERROR   5               ;ERROR, CRAM PC IS WRONG
        TRAP    C$ERDF
        .WORD   5
        .WORD   EM0
        .WORD   ERR5
4$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10001$-.
        ENDSEG
10001$:
        TRAP    C$ESEG
        BGNSEG
        TRAP    C$BSEG
        JSR     PC,SETBR4               ;SET THE BR4 BIT'
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100406                  ;START AT ROM PC=6
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        104125!<400*6>          ;JUMP TO ROM PC OF 525
        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
        125                     ;EXPECTED DATA
        CMPB    R5,R4           ;IS ROM PC CORRECT?
        BEQ     6$              ;BR IF YES
        ERROR   5               ;ERROR, CRAM PC IS WRONG
        TRAP    C$ERDF
        .WORD   5
        .WORD   EM0
        .WORD   ERR5
6$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10002$-.
        ENDSEG
10002$:
        TRAP    C$ESEG

ENDTST
L10063:
```

```
 4670  017114  104401                    TRAP    C$ETST
 4671
 4672  017116                            BADHEAD
 4673                                    ;*************** TEST 18 ***************
 4674                                    ;*CRAM TEST OF JUMP(I) ON BR7 SET MICRO-PROCESSOR INSTRUCTION.
 4675                                    ;*SET THE BR7 BIT, PERFORM THE JUMP INSTRUCTION.
 4676                                    ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
 4677                                    ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
 4678                                    ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
 4679                                    ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
 4680                                    ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
 4681                                    ;*THEN PORT4 WILL CONTAIN A 37
 4682  017116                            BADHEAD
 4683                                    ;*************** TEST 18 ***************
 4684
 4685  017116                    BGNTST
 4686  017116                    T18::
 4687  017116                            MACEX2                  ;DON'T DO IF M8200.
 4688                                    ;DO NOT DO TEST IF M8200
 4689  017126  104432                    TRAP    C$EXIT
 4690  017130  000230                    .WORD   L10064-.
 4691  017132                    MYINT
 4692  017132  013701  002516            MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
 4693                                                            ;R1 CONTAINS BASE M8200,4,7 ADDRESS
 4694  017136                            MSTCLR                  ;MASTER CLEAR M8200,4,7
 4695  017142  004737  003474            JSR     PC,MEMSET       ;SET MEM AND RAM
 4696  017146                    1$:     BGNSEG
 4697  017146  104404                    TRAP    C$BSEG
 4698  017150  004737  003250            JSR     PC,SETBR7       ;SET THE BR7 BIT'
 4699  017154                            SROMCLK                 ;NEXT WORD IS INSTRUCTION,
 4700  017154  004537  003100            JSR     R5,..SROMCLK
 4701  017160  100400                    100400                  ;START AT ROM PC=0
 4702  017162                            SROMCLK                 ;NEXT WORD IS INSTRUCION,
 4703  017162  004537  003100            JSR     R5,..SROMCLK
 4704  017166  117777                    114377!<400*7>          ;JUMP TO ROM PC OF 1777
 4705  017170  004737  003330            JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
 4706  017174  000377                    377                     ;EXPECTED DATA
 4707  017176  120504                    CMPB    R5,R4           ;IS ROM PC CORRECT?
 4708  017200  001406                    BEQ     2$              ;BR IF YES
 4709  017202                            ERROR   5               ;ERROR, CRAM PC IS WRONG
 4710  017206  104455                    TRAP    C$ERDF
 4711  017210  000005                    .WORD   5
 4712  017212  005055                    .WORD   EM0
 4713  017214  006556                    .WORD   ERR5
 4714  017216                    2$:     ESCAPE  SEG
 4715  017216  104410                    TRAP    C$ESCAPE
 4716  017220  000002                    .WORD   10000$-.
 4717  017222                            ENDSEG
 4718  017222                    10000$:
 4719  017222  104405                    TRAP    C$ESEG
 4720  017224                            BGNSEG
 4721  017224  104404                    TRAP    C$BSEG
 4722  017226  004737  003250            JSR     PC,SETBR7               ;SET THE BR7 BIT'
 4723  017232                            SROMCLK         ;NEXT WORD IS INSTRUCTION,
 4724  017232  004537  003100            JSR     R5,..SROMCLK
 4725  017236  100403                    100403                  ;START AT ROM PC=3
```

```
4726  017240                        SROMCLK                 ;NEXT WORD IS INSTRUCTION.
4727  017240  004537  003100        JSR     R5,.SROMCLK
4728  017244  103400                100000!<400*7>          ;JUMP TO ROM PC OF 0
4729  017246  004737  003330        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4730  017252  000000                0                       ;EXPECTED DATA
4731  017254  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
4732  017256  001406                BEQ     4$              ;BR IF YES
4733  017260                        ERROR   5               ;ERROR, CRAM PC IS WRONG
4734  017264  104455                TRAP    C$ERDF
4735  017266  000005                .WORD   5
4736  017270  005055                .WORD   EM0
4737  017272  006556                .WORD   ERR5
4738  017274                  4$:   ESCAPE SEG
4739  017274  104410                TRAP    C$ESCAPE
4740  017276  000002                .WORD   10001$-.
4741  017300                        ENDSEG
4742  017300                  10001$:
4743  017300  104405                TRAP    C$ESEG
4744  017302                        BGNSEG
4745  017302  104404                TRAP    C$BSEG
4746  017304  004737  003250        JSR     PC,SETBR7               ;SET THE BR7 BIT'
4747  017310                        SROMCLK                 ;NEXT WORD IS INSTRUCTION.
4748  017310  004537  003100        JSR     R5,.SROMCLK
4749  017314  100406                100406                  ;START AT ROM PC=6
4750  017316                        SROMCLK                 ;NEXT WORD IS INSTRUCTION.
4751  017316  004537  003100        JSR     R5,.SROMCLK
4752  017322  107525                104125!<400*7>          ;JUMP TO ROM PC OF 525
4753  017324  004737  003330        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4754  017330  000125                125                     ;EXPECTED DATA
4755  017332  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
4756  017334  001406                BEQ     6$              ;BR IF YES
4757  017336                        ERROR   5               ;ERROR, CRAM PC IS WRONG
4758  017342  104455                TRAP    C$ERDF
4759  017344  000005                .WORD   5
4760  017346  005055                .WORD   EM0
4761  017350  006556                .WORD   ERR5
4762  017352                  6$:   ESCAPE SEG
4763  017352  104410                TRAP    C$ESCAPE
4764  017354  000002                .WORD   10002$-.
4765  017356                        ENDSEG
4766  017356                  10002$:
4767  017356  104405                TRAP    C$ESEG
4768  017360                  ENDTST
4769  017360                  L10064:
4770  017360  104401                TRAP    C$ETST
4771
4772  017362                        BADHEAD
4773                                ;*************** TEST 19 ***************
4774                                ;*CRAM TEST OF JUMP(I) ON C BIT CLEAR MICRO-PROCESSOR INSTRUCTION.
4775                                ;*SET THE C BIT, PERFORM THE JUMP INSTRUCTION.
4776                                ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
4777                                ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
4778                                ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
4779                                ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
4780                                ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
4781                                ;*THEN PORT4 WILL CONTAIN A 37
```

```
4782  017362                                BADHEAD
4783                                        ;*************** TEST 19 ****************
4784
4785  017362                          BGNTST
4786  017362                          T19::
4787  017362                                MACEX2                  ;DON'T DO IF M8200.
4788                                        ;DO NOT DO TEST IF M8200
4789  017372  104432                        TRAP    C$EXIT
4790  017374  000244                        .WORD   L10065-.
4791  017376                                MYINT
4792  017376  013701  002516                MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
4793  017402                                MSTCLR                  ;MASTER CLEAR M8200,4,7
4794  017406  004737  003474                JSR     PC,MEMSET       ;SET MEM AND RAM
4795  017412                          1$:   BGNSEG
4796  017412  104404                        TRAP    C$BSEG
4797  017414  004737  003260                JSR     PC,SETC
4798  017420  004737  003166                JSR     PC,CLRALL
4799  017424                                SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4800  017424  004537  003100                JSR     R5,..SROMCLK
4801  017430  100400                        100400                  ;START AT ROM PC=0
4802  017432                                SROMCLK                 ;NEXT WORD IS INSTRUCION,
4803  017432  004537  003100                JSR     R5,..SROMCLK
4804  017436  115377                        114377!<400*2>          ;JUMP TO ROM PC OF 1777
4805  017440  004737  003330                JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4806  017444  000001                        1                       ;EXPECTED DATA
4807  017446  120504                        CMPB    R5,R4           ;IS ROM PC CORRECT?
4808  017450  001406                        BEQ     2$              ;BR IF YES
4809  017452                                ERROR   5               ;ERROR, CRAM PC IS WRONG
4810  017456  104455                        TRAP    C$ERDF
4811  017460  000005                        .WORD   5
4812  017462  005055                        .WORD   EM0
4813  017464  006556                        .WORD   ERR5
4814  017466                          2$:   ESCAPE SEG
4815  017466  104410                        TRAP    C$ESCAPE
4816  017470  000002                        .WORD   10000$-.
4817  017472                                ENDSEG
4818  017472                          10000$:
4819  017472  104405                        TRAP    C$ESEG
4820  017474                                BGNSEG
4821  017474  104404                        TRAP    C$BSEG
4822  017476                                SKIP06  6$
4823                                        ;GOTO 6$ IF M8206
4824  017506  004737  003166                JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
4825  017512                                SROMCLK                 ;NEXT WORD OF INSTRUCTION
4826  017512  004537  003100                JSR     R5,..SROMCLK
4827  017516  100403                        100403                  ;START AT ROM PC=3
4828  017520                                SROMCLK                 ;NEXT WORD OF INSTRUCTION
4829  017520  004537  003100                JSR     R5,..SROMCLK
4830  017524  101000                        100000!<400*2>          ;JUMP TO ROM PC OF 0
4831  017526  004737  003330                JSR     PC,RAMDAT       ;R4=CRAM PC(LSB 8 BITS)
4832  017532  000004                        4                       ;EXPECTED DATA
4833  017534  120504                        CMPB    R5,R4           ;IS ROM PC CORRECT?
4834  017536  001406                        BEQ     4$              ;BR IF YES
4835  017540                                ERROR   5               ;ERROR, CRAM PC IS WRONG
4836  017544  104455                        TRAP    C$ERDF
4837  017546  000005                        .WORD   5
```

```
4838  017550  005055                        .WORD   EM0
4839  017552  006556                        .WORD   ERR5
4840  017554                        4$:     ESCAPE SEG
4841  017554  104410                        TRAP    C$ESCAPE
4842  017556  000002                        .WORD   10001$-.
4843  017560                                ENDSEG
4844  017560                        10001$:
4845  017560  104405                        TRAP    C$ESEG
4846  017562                                BGNSEG
4847  017562  104404                        TRAP    C$BSEG
4848  017564  004737  003166                JSR     PC,CLRALL               ;CLEAR ALL CONDITIONS
4849  017570                                SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4850  017570  004537  003100                JSR     R5,..SROMCLK
4851  017574  100406                        100406                  ;START AT ROM PC=6
4852  017576                                SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4853  017576  004537  003100                JSR     R5,..SROMCLK
4854  017602  105125                        104125!<400*2>          ;JUMP TO ROM PC OF 525
4855  017604  004737  003330                JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4856  017610  000007                        7                       ;EXPECTED DATA
4857  017612  120504                        CMPB    R5,R4           ;IS ROM PC CORRECT?
4858  017614  001406                        BEQ     6$              ;BR IF YES
4859  017616                                ERROR   5               ;ERROR, CRAM PC IS WRONG
4860  017622  104455                        TRAP    C$ERDF
4861  017624  000005                        .WORD   5
4862  017626  005055                        .WORD   EM0
4863  017630  006556                        .WORD   ERR5
4864  017632                        6$:     ESCAPE SEG
4865  017632  104410                        TRAP    C$ESCAPE
4866  017634  000002                        .WORD   10002$-.
4867  017636                                ENDSEG
4868  017636                        10002$:
4869  017636  104405                        TRAP    C$ESEG
4870  017640                        ENDTST
4871  017640                        L10065:
4872  017640  104401                        TRAP    C$ETST
4873
4874  017642                                BADHEAD
4875                                        ;*************** TEST 20 ****************
4876                                        ;*CRAM TEST OF JUMP(I) ON Z BIT CLEAR MICRO-PROCESSOR INSTRUCTION.
4877                                        ;*CLEAR THE Z BIT, PERFORM THE JUMP INSTRUCTION.
4878                                        ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
4879                                        ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
4880                                        ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
4881                                        ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
4882                                        ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
4883                                        ;*THEN PORT4 WILL CONTAIN A 37
4884  017642                                BADHEAD
4885                                        ;*************** TEST 20 ****************
4886
4887  017642                        BGNTST
4888  017642                        T20::
4889  017642                                MACEX2                  ;DON'T DO IF M8200.
4890                                        ;DO NOT DO TEST IF M8200
4891  017652  104432                        TRAP    C$EXIT
4892  017654  000244                        .WORD   L10066-.
4893  017656                                MYINT
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
4894  017656  013701  002516          MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
4895  017662                          MSTCLR                          ;MASTER CLEAR M8200,4,7
4896  017666  004737  003474          JSR     PC,MEMSET               ;SET MEM AND RAM
4897  017672                  1$:     BGNSEG
4898  017672  104404                  TRAP    C$BSEG
4899  017674  004737  003312          JSR     PC,SETZ
4900  017700  004737  003166          JSR     PC,CLRALL               ; CLEAR CONDITION CODES ;*** B0
4901  017704                          SROMCLK                         ;NEXT WORD IS INSTRUCTION,
4902  017704  004537  003100          JSR     R5,..SROMCLK
4903  017710  100400                  100400                          ;START AT ROM PC=0
4904  017712                          SROMCLK                         ;NEXT WORD IS INSTRUCION,
4905  017712  004537  003100          JSR     R5,..SROMCLK
4906  017716  115777                  114377!<400*3>                  ;JUMP TO ROM PC OF 1777
4907  017720  004737  003330          JSR     PC,RAMDAT               ;R4=CRAM PC (LSB 8 BITS)
4908  017724  000001                  1                               ;EXPECTED DATA
4909  017726  120504                  CMPB    R5,R4                   ;IS ROM PC CORRECT?
4910  017730  001406                  BEQ     2$                      ;BR IF YES
4911  017732                          ERROR   5                       ;ERROR, CRAM PC IS WRONG
4912  017736  104455                  TRAP    C$ERDF
4913  017740  000005                  .WORD   5
4914  017742  005055                  .WORD   EM0
4915  017744  006556                  .WORD   ERR5
4916  017746                  2$:     ESCAPE SEG
4917  017746  104410                  TRAP    C$ESCAPE
4918  017750  000002                  .WORD   10000$-.
4919  017752                          ENDSEG
4920  017752                  10000$:
4921  017752  104405                  TRAP    C$ESEG
4922  017754                          BGNSEG
4923  017754  104404                  TRAP    C$BSEG
4924  017756                          SKIP06  6$
4925                                  ;GOTO 6$ IF M8206
4926  017766  004737  003166          JSR     PC,CLRALL               ;CLEAR ALL CONDITIONS
4927  017772                          SROMCLK                         ;NEXT WORD IS INSTRUCTION,
4928  017772  004537  003100          JSR     R5,..SROMCLK
4929  017776  100403                  100403                          ;START AT ROM PC=3
4930  020000                          SROMCLK                         ;NEXT WORD IS INSTRUCION,
4931  020000  004537  003100          JSR     R5,..SROMCLK
4932  020004  101400                  100000!<400*3>                  ;JUMP TO ROM PC OF 0
4933  020006  004737  003330          JSR     PC,RAMDAT               ;R4=CRAM PC (LSB 8 BITS)
4934  020012  000004                  4                               ;EXPECTED DATA
4935  020014  120504                  CMPB    R5,R4                   ;IS ROM PC CORRECT?
4936  020016  001406                  BEQ     4$                      ;BR IF YES
4937  020020                          ERROR   5                       ;ERROR, CRAM PC IS WRONG
4938  020024  104455                  TRAP    C$ERDF
4939  020026  000005                  .WORD   5
4940  020030  005055                  .WORD   EM0
4941  020032  006556                  .WORD   ERR5
4942  020034                  4$:     ESCAPE SEG
4943  020034  104410                  TRAP    C$ESCAPE
4944  020036  000002                  .WORD   10001$-.
4945  020040                          ENDSEG
4946  020040                  10001$:
4947  020040  104405                  TRAP    C$ESEG
4948  020042                          BGNSEG
4949  020042  104404                  TRAP    C$BSEG
```

```
4950  020044  004737  003166          JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
4951  020050                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4952  020050  004537  003100          JSR     R5,..SROMCLK
4953  020054  100406                  100406                  ;START AT ROM PC=6
4954  020056                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
4955  020056  004537  003100          JSR     R5,..SROMCLK
4956  020062  105525                  104125!<400*3>          ;JUMP TO ROM PC OF 525
4957  020064  004737  003330          JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
4958  020070  000007                  7                       ;EXPECTED DATA
4959  020072  120504                  CMPB    R5,R4           ;IS ROM PC CORRECT?
4960  020074  001406                  BEQ     6$              ;BR IF YES
4961  020076                          ERROR   5               ;ERROR, CRAM PC IS WRONG
4962  020102  104455                  TRAP    C$ERDF
4963  020104  000005                  .WORD   5
4964  020106  005055                  .WORD   EM0
4965  020110  006556                  .WORD   ERR5
4966  020112                  6$:     ESCAPE  SEG
4967  020112  104410                  TRAP    C$ESCAPE
4968  020114  000002                  .WORD   10002$-.
4969  020116                          ENDSEG
4970  020116                  10002$:
4971  020116  104405                  TRAP    C$ESEG
4972  020120                  ENDTST
4973  020120                  L10066:
4974  020120  104401                  TRAP    C$ETST
4975
4976  020122                          BADHEAD
4977                                  ;*************** TEST 21 ****************
4978                                  ;*CRAM TEST OF JUMP(I) ON BRO CLEAR MICRO-PROCESSOR INSTRUCTION.
4979                                  ;*CLEAR THE BRO BIT, PERFORM THE JUMP INSTRUCTION.
4980                                  ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
4981                                  ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
4982                                  ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
4983                                  ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
4984                                  ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
4985                                  ;*THEN PORT4 WILL CONTAIN A 37
4986  020122                          BADHEAD
4987                                  ;*************** TEST 21 ****************
4988
4989  020122                  BGNTST
4990  020122                  T21::
4991  020122                          MACEX2                  ;DON'T DO IF M8200.
4992                                  ;DO NOT DO TEST IF M8200
4993  020132  104432                  TRAP    C$EXIT
4994  020134  000240                  .WORD   L10067-.
4995  020136                          MYINT
4996  020136  013701  002516          MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
4997  020142                          MSTCLR                  ;MASTER CLEAR M8200,4,7
4998  020146  004737  003474          JSR     PC,MEMSET       ;SET MEM AND RAM
4999  020152                  1$:     BGNSEG
5000  020152  104404                  TRAP    C$BSEG
5001  020154  004737  003166          JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
5002  020160                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
5003  020160  004537  003100          JSR     R5,..SROMCLK
5004  020164  100400                  100400                  ;START AT ROM PC=0
5005  020166                          SROMCLK                 ;NEXT WORD IS INSTRUCION,
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
5006  020166  004537  003100          JSR     R5,.SROMCLK
5007  020172  116377                  114377!<400*4>          ;JUMP TO ROM PC OF 1777
5008  020174  004737  003330          JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
5009  020200  000001                  1                       ;EXPECTED DATA
5010  020202  120504                  CMPB    R5,R4           ;IS ROM PC CORRECT?
5011  020204  001406                  BEQ     2$              ;BR IF YES
5012  020206                          ERROR   5               ;ERROR, CRAM PC IS WRONG
5013  020212  104455                  TRAP    C$ERDF
5014  020214  000005                  .WORD   5
5015  020216  005055                  .WORD   EM0
5016  020220  006556                  .WORD   ERR5
5017  020222                  2$:     ESCAPE  SEG
5018  020222  104410                  TRAP    C$ESCAPE
5019  020224  000002                  .WORD   10000$-.
5020  020226                          ENDSEG
5021  020226                  10000$:
5022  020226  104405                  TRAP    C$ESEG
5023  020230                          BGNSEG
5024  020230  104404                  TRAP    C$BSEG
5025  020232                          SKIP06  6$
5026                                  ;GOTO 6$ IF M8206
5027  020242  004737  003166          JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
5028  020246                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
5029  020246  004537  003100          JSR     R5,.SROMCLK
5030  020252  100403                  100403                  ;START AT ROM PC=3
5031  020254                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
5032  020254  004537  003100          JSR     R5,.SROMCLK
5033  020260  102000                  100000!<400*4>          ;JUMP TO ROM PC OF 0
5034  020262  004737  003330          JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
5035  020266  000004                  4                       ;EXPECTED DATA
5036  020270  120504                  CMPB    R5,R4           ;IS ROM PC CORRECT?
5037  020272  001406                  BEQ     4$              ;BR IF YES
5038  020274                          ERROR   5               ;ERROR, CRAM PC IS WRONG
5039  020300  104455                  TRAP    C$ERDF
5040  020302  000005                  .WORD   5
5041  020304  005055                  .WORD   EM0
5042  020306  006556                  .WORD   ERR5
5043  020310                  4$:     ESCAPE  SEG
5044  020310  104410                  TRAP    C$ESCAPE
5045  020312  000002                  .WORD   10001$-.
5046  020314                          ENDSEG
5047  020314                  10001$:
5048  020314  104405                  TRAP    C$ESEG
5049  020316                          BGNSEG
5050  020316  104404                  TRAP    C$BSEG
5051  020320  004737  003166          JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
5052  020324                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
5053  020324  004537  003100          JSR     R5,.SROMCLK
5054  020330  100406                  100406                  ;START AT ROM PC=6
5055  020332                          SROMCLK                 ;NEXT WORD IS INSTRUCTION,
5056  020332  004537  003100          JSR     R5,.SROMCLK
5057  020336  106125                  104125!<400*4>          ;JUMP TO ROM PC OF 525
5058  020340  004737  003330          JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
5059  020344  000007                  7                       ;EXPECTED DATA
5060  020346  120504                  CMPB    R5,R4           ;IS ROM PC CORRECT?
5061  020350  001406                  BEQ     6$              ;BR IF YES
```

```
5062  020352                        ERROR   5               ;ERROR, CRAM PC IS WRONG
5063  020356  104455                TRAP    C$ERDF
5064  020360  000005                .WORD   5
5065  020362  005055                .WORD   EM0
5066  020364  006556                .WORD   ERR5
5067  020366                  6$:   ESCAPE SEG
5068  020366  104410                TRAP    C$ESCAPE
5069  020370  000002                .WORD   10002$-.
5070  020372                        ENDSEG
5071  020372                  10002$:
5072  020372  104405                TRAP    C$ESEG
5073  020374                  ENDTST
5074  020374                  L10067:
5075  020374  104401                TRAP    C$ETST
5076
5077  020376                        BADHEAD
5078                                ;*************** TEST 22 ***************
5079                                ;*CRAM TEST OF JUMP(I) ON BR1 CLEAR MICRO-PROCESSOR INSTRUCTION.
5080                                ;*CLEAR THE BR1 BIT, PERFORM THE JUMP INSTRUCTION.
5081                                ;*VERIFY THE JUMP DID OCCUR BY CLOCKING THE INSTRUCTION
5082                                ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
5083                                ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
5084                                ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
5085                                ;*THE JUMP WAS SUCCESSFUL, IF THE JUMP WAS UNSUCCESSFUL
5086                                ;*THEN PORT4 WILL CONTAIN A 37
5087  020376                        BADHEAD
5088                                ;*************** TEST 22 ***************
5089
5090  020376                  BGNTST
5091  020376                  T22::
5092  020376                        MACEX2                  ;DON'T DO IF M8200.
5093                                ;DO NOT DO TEST IF M8200
5094  020406  104432                TRAP    C$EXIT
5095  020410  000240                .WORD   L10070-.
5096  020412                        MYINT
5097  020412  013701  002516        MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
5098  020416                        MSTCLR                  ;MASTER CLEAR M8200,4,7
5099  020422  004737  003474        JSR     PC,MEMSET       ;SET MEM AND RAM
5100  020426                  1$:   BGNSEG
5101  020426  104404                TRAP    C$BSEG
5102  020430  004737  003166        JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
5103  020434                        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
5104  020434  004537  003100        JSR     R5,..SROMCLK
5105  020440  100400                100400                  ;START AT ROM PC=0
5106  020442                        SROMCLK                 ;NEXT WORD IS INSTRUCION,
5107  020442  004537  003100        JSR     R5,..SROMCLK
5108  020446  116777                114377:<400*5>          ;JUMP TO ROM PC OF 1777
5109  020450  004737  003330        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
5110  020454  000001                1                       ;EXPECTED DATA
5111  020456  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
5112  020460  001406                BEQ     2$              ;BR IF YES
5113  020462                        ERROR   5               ;ERROR, CRAM PC IS WRONG
5114  020466  104455                TRAP    C$ERDF
5115  020470  000005                .WORD   5
5116  020472  005055                .WORD   EM0
5117  020474  006556                .WORD   ERR5
```

```
5118  020476
5119  020476  104410
5120  020500  000002
5121  020502
5122  020502
5123  020502  104405
5124  020504
5125  020504  104404
5126  020506
5127
5128  020516  004737  003166
5129  020522
5130  020522  004537  003100
5131  020526  100403
5132  020530
5133  020530  004537  003100
5134  020534  102400
5135  020536  004737  003330
5136  020542  000004
5137  020544  120504
5138  020546  001406
5139  020550
5140  020554  104455
5141  020556  000005
5142  020560  005055
5143  020562  006556
5144  020564
5145  020564  104410
5146  020566  000002
5147  020570
5148  020570
5149  020570  104405
5150  020572
5151  020572  104404
5152  020574  004737  003166
5153  020600
5154  020600  004537  003100
5155  020604  100406
5156  020606
5157  020606  004537  003100
5158  020612  106525
5159  020614  004737  003330
5160  020620  000007
5161  020622  120504
5162  020624  001406
5163  020626
5164  020632  104455
5165  020634  000005
5166  020636  005055
5167  020640  006556
5168  020642
5169  020642  104410
5170  020644  000002
5171  020646
5172  020646
5173  020646  104405
```

```
2$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10000$-.
        ENDSEG
10000$:
        TRAP    C$ESEG
        BGNSEG
        TRAP    C$BSEG
        SKIP06  6$
        ;GOTO 6$ IF M8206
        JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100403                  ;START AT ROM PC=3
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100000!<400*5>          ;JUMP TO ROM PC OF 0
        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
        4                       ;EXPECTED DATA
        CMPB    R5,R4           ;IS ROM PC CORRECT?
        BEQ     4$              ;BR IF YES
        ERROR   5               ;ERROR, CRAM PC IS WRONG
        TRAP    C$ERDF
        .WORD   5
        .WORD   EM0
        .WORD   ERR5
4$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10001$-.
        ENDSEG
10001$:
        TRAP    C$ESEG
        BGNSEG
        TRAP    C$BSEG
        JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        100406                  ;START AT ROM PC=6
        SROMCLK                 ;NEXT WORD IS INSTRUCTION,
        JSR     R5,..SROMCLK
        104125!<400*5>          ;JUMP TO ROM PC OF 525
        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
        7                       ;EXPECTED DATA
        CMPB    R5,R4           ;IS ROM PC CORRECT?
        BEQ     6$              ;BR IF YES
        ERROR   5               ;ERROR, CRAM PC IS WRONG
        TRAP    C$ERDF
        .WORD   5
        .WORD   EM0
        .WORD   ERR5
6$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10002$-.
        ENDSEG
10002$:
        TRAP    C$ESEG
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
5174  020650                    ENDTST
5175  020650                    L10070:
5176  020650  104401                    TRAP    CSETST
5177
5178  020652                            BADHEAD
5179                                    ;*************** TEST 23 ***************
5180                                    ;*CRAM TEST OF JUMP(I) ON BR4 CLEAR MICRO-PROCESSOR INSTRUCTION.
5181                                    ;*CLEAR THE BR4 BIT, PERFORM THE JUMP INSTRUCTION.
5182                                    ;*VERIFY THE JUMP DID NOT OCCUR BY CLOCKING THE INSTRUCTION
5183                                    ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
5184                                    ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
5185                                    ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
5186  020652  020652                    .
5187                                    ;*THE CRAM PC IS CORRECT, IF THE CRAM PC IS NOT RIGHT.
5188                                    ;*THEN PORT4 CONTAINS A 37
5189  020654                            BADHEAD
5190                                    ;*************** TEST 23 ***************
5191
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
5192  020654                    BGNTST
5193  020654                    T23::
5194  020654                            MACEX2                    ;DON'T DO IF M8200.
5195                                    ;DO NOT DO TEST IF M8200
5196  020664  104432                    TRAP    C$EXIT
5197  020666  000240                    .WORD   L10071-.
5198  020670                            MYINT
5199  020670  013701  002516            MOV     KMCSR,R1                  ;RECORD DEVICE ADDR.
5200  020674                            MSTCLR                    ;MASTER CLEAR M8200,4,7
5201  020700  004737  003474            JSR     PC,MEMSET         ;SET MEM AND RAM
5202  020704                    1$:     BGNSEG
5203  020704  104404                    TRAP    C$BSEG
5204  020706  004737  003166            JSR     PC,CLRALL         ;CLEAR ALL CONDITIONS
5205  020712                            SROMCLK                   ;NEXT WORD IS INSTRUCTION.
5206  020712  004537  003100            JSR     R5,..SROMCLK
5207  020716  100400                    100400                    ;START AT ROM PC=0
5208  020720                            SROMCLK                   ;NEXT WORD IS INSTRUCTION.
5209  020720  004537  003100            JSR     R5,..SROMCLK
5210  020724  117377                    114377!<400*6>            ;JUMP TO ROM PC OF 1777
5211  020726  004737  003330            JSR     PC,RAMDAT         ;R4=CRAM PC (LSB 8 BITS)
```

```
5212  020732  000001                        1                       ;EXPECTED DATA
5213  020734  120504                        CMPB    R5,R4           ;IS ROM PC CORRECT?
5214  020736  001406                        BEQ     2$              ;BR IF YES
5215  020740                                ERROR   5               ;ERROR, CRAM PC IS WRONG
5216  020744  104455                        TRAP    C$ERDF
5217  020746  000005                        .WORD   5
5218  020750  005055                        .WORD   EM0
5219  020752  006556                        .WORD   ERR5
5220  020754                        2$:     ESCAPE SEG
5221  020754  104410                        TRAP    C$ESCAPE
5222  020756  000002                        .WORD   10000$-.
5223  020760                                ENDSEG
5224  020760                        10000$:
5225  020760  104405                        TRAP    C$ESEG
5226  020762                                BGNSEG
5227  020762  104404                        TRAP    C$BSEG
5228  020764                                SKIP06  6$
5229                                        ;GOTO 6$ IF M8206
5230  020774  004737  003166                JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
5231  021000                                SROMCLK                 ;NEXT WORD IS INSTRUCTION.
5232  021000  004537  003100                JSR     R5,..SROMCLK
5233  021004  100403                        100403                  ;START AT ROM PC=3
5234  021006                                SROMCLK                 ;NEXT WORD IS INSTRUCTION.
5235  021006  004537  003100                JSR     R5,..SROMCLK
5236  021012  103000                        100000!<400*6>  ;JUMP TO ROM PC OF 0
5237  021014  004737  003330                JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
5238  021020  000004                        4                       ;EXPECTED DATA
```

```
5239  021022  120504                CMPB    R5,R4           ;IS ROM PC CORRECT?
5240  021024  001406                BEQ     4$              ;BSR IF YES
```

```
5241  021026                        ERROR   5               ;ERROR, CRAM PC IS WRONG
5242  021032  104455                TRAP    C$ERDF
5243  021034  000005                .WORD   5
5244  021036  005055                .WORD   EM0
5245  021040  006556                .WORD   ERR5
```

```
5246  021042
5247  021042  104410
5248  021044  000002
5249  021046
5250  021046
5251  021046  104405
5252  021050
5253  021050  104404
5254  021052  004737  003166
5255  021056
5256  021056  004537  003100
5257  021062  100406
5258  021064
5259  021064  004537  003100
5260  021070  107125
5261  021072  004737  003330
5262  021076  000007
5263  021100  120504
5264  021102  001406
5265  021104
5266  021110  104455
5267  021112  000005
5268  021114  005055
5269  021116  006556
5270  021120
5271  021120  104410
5272  021122  000002
5273  021124
5274  021124
5275  021124  104405
5276  021126
5277  021126
5278  021126  104401
5279
5280  021130
5281
5282
5283
5284
5285
5286
5287
5288
5289
5290  021130
5291
5292
5293  021130
5294  021130
5295  021130
5296
5297  021140  104432
5298  021142  000240
5299  021144
5300  021144  013701  002516
5301  021150
```

```
4$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10001$-.
        ENDSEG
10001$:
        TRAP    C$ESEG
        BGNSEG
        TRAP    C$BSEG
        JSR     PC,CLRALL       ;CLEAR ALL CONDITIONS
        SROMCLK                 ;NEXT WORD IS INSTRUCTION.
        JSR     R5,..SROMCLK
        100406                  ;START AT ROM PC=6
        SROMCLK                 ;NEXT WORD IS INSTRUCTION.
        JSR     R5,..SROMCLK
        104125!<400*6>  ;JUMP TO ROM PC OF 525
        JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
        7                       ;EXPECTED DATA
        CMPB    R5,R4           ;IS ROM PC CORRECT?
        BEQ     6$              ;BR IF YES
        ERROR   5               ;ERROR, CRAM PC IS WRONG
        TRAP    C$ERDF
        .WORD   5
        .WORD   EM0
        .WORD   ERR5
6$:     ESCAPE SEG
        TRAP    C$ESCAPE
        .WORD   10002$-.
        ENDSEG
10002$:
        TRAP    C$ESEG
ENDTST
L10071:
        TRAP    C$ETST

        BADHEAD
        ;*************** TEST 24 ***************
        ;*CRAM TEST OF JUMP(I) ON BR7 CLEAR MICRO-PROCESSOR INSTRUCTION.
        ;*CLEAR THE BR7 BIT, PERFORM THE JUMP INSTRUCTION.
        ;*VERIFY THE JUMP DID NOT OCCUR BY CLOCKING THE INSTRUCTION
        ;*IN THE LOCATION IT IS AT. THIS INSTRUCTION LOADS THE
        ;*BR WITH THE LOWEST 8 BITS OF THE CRAM PC. AT THIS POINT
        ;*THE BR DATA IS MOVED TO PORT4. IF THIS DATA IS CORRECT
        ;*THE CRAM PC IS CORRECT, IF THE CRAM PC IS NOT RIGHT.
        ;*THEN PORT4 CONTAINS A 37
        BADHEAD
        ;*************** TEST 24 ***************

BGNTST
T24::
        MACEX2                  ;DON'T DO IF M8200.
        ;DO NOT DO TEST IF M8200
        TRAP    C$EXIT
        .WORD   L10072-.
MYINT
        MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
        MSTCLR                  ;MASTER CLEAR M8200,4,7
```

```
021154  004737  003474                  JSR     PC,MEMSET               ;SET MEM AND RAM
021160                          1$:     BGNSEG
021160  104404                          TRAP    C$BSEG
021162  004737  003166                  JSR     PC,CLRALL               ;CLEAR ALL CONDITIONS
021166                                  SROMCLK                         ;NEXT WORD IS INSTRUCTION,
021166  004537  003100                  JSR     R5,.SROMCLK
021172  100400                          100400          ;START AT ROM PC=0
021174                                  SROMCLK                         ;NEXT WORD IS INSTRUCTION,
021174  004537  003100                  JSR     R5,.SROMCLK
021200  117777                          114377!<400*7>                  ;JUMP TO ROM PC OF 1777
021202  004737  003330                  JSR     PC,RAMDAT               ;R4=CRAM PC (LSB 8 BITS)
021206  000001                          1                               ;EXPECTED DATA
021210  120504                          CMPB    R5,R4                   ;IS ROM PC CORRECT?
021212  001406                          BEQ     2$                      ;BR IF YES
021214                                  ERROR   5                       ;ERROR, CRAM PC IS WRONG
021220  104455                          TRAP    C$ERDF
021222  000005                          .WORD   5
021224  005055                          .WORD   EM0
021226  006556                          .WORD   ERR5
021230                          2$:     ESCAPE SEG
021230  104410                          TRAP    C$ESCAPE
021232  000002                          .WORD   10000$-.
021234                                  ENDSEG
021234                          10000$:
021234  104405                          TRAP    C$ESEG
021236                                  BGNSEG
021236  104404                          TRAP    C$BSEG
021240                                  SKIP06  6$
                                        ;GOTO 6$ IF M8206
021250  004737  003166                  JSR     PC,CLRALL               ;CLEAR ALL CONDITIONS
021254                                  SROMCLK                         ;NEXT WORD IS INSTRUCTION,
021254  004537  003100                  JSR     R5,.SROMCLK
021260  100403                          100403                          ;START AT ROM PC=3
021262                                  SROMCLK                         ;NEXT WORD IS INSTRUCTION,
021262  004537  003100                  JSR     R5,.SROMCLK
021266  103400                          100000!<400*7>  ;JUMP TO ROM PC OF 0
021270  004737  003330                  JSR     PC,RAMDAT               ;;R4=CRAM PC (LCB 8 BITS)
021274  000004                          4                               ;EXPECTED DATA
021276  120504                          CMPB    R5,R4                   ;IS ROM PC CORRECT?
021300  001406                          BEQ     4$                      ;BR IF YES
021302                                  ERROR   5                       ;ERROR, CRAM PC IS WRONG
021306  104455                          TRAP    C$ERDF
021310  000005                          .WORD   5
021312  005055                          .WORD   EM0
021314  006556                          .WORD   ERR5
021316                          4$:     ESCAPE SEG
021316  104410                          TRAP    C$ESCAPE
021320  000002                          .WORD   10001$-.
021322                                  ENDSEG
021322                          10001$:
021322  104405                          TRAP    C$ESEG
021324                                  BGNSEG
021324  104404                          TRAP    C$BSEG
021326  004737  003166                  JSR     PC,CLRALL               ;CLEAR ALL CONDITIONS
021332                                  SROMCLK                         ;NEXT WORD IS INSTRUCTION,
021332  004537  003100                  JSR     R5,.SROMCLK
```

```
5358  021336  100406                          100406                  ;START AT ROM PC=6
5359  021340                                  SROMCLK                 ;NEXT WORD IS INSTRUCTION,
5360  021340  004537  003100                  JSR     R5,.SROMCLK
5361  021344  107525                          104125!<400*7>          ;JUMP TO ROM PC OF 525
5362  021346  004737  003330                  JSR     PC,RAMDAT       ;R4=CRAM PC (LSB 8 BITS)
5363  021352  000007                          7                       ;EXPECTED DATA
5364  021354  120504                          CMPB    R5,R4           ;IS ROM PC CORRECT?
5365  021356  001406                          BEQ     6$              ;BR IF YES
5366  021360                                  ERROR   5               ;ERROR, CRAM PC IS WRONG
5367  021364  104455                          TRAP    C$ERDF
5368  021366  000005                          .WORD   5
5369  021370  005055                          .WORD   EM0
5370  021372  006556                          .WORD   ERR5
5371  021374                          6$:     ESCAPE SEG
5372  021374  104410                          TRAP    C$ESCAPE
5373  021376  000002                          .WORD   10002$-.
5374  021400                                  ENDSEG
5375  021400                          10002$:
5376  021400  104405                          TRAP    C$ESEG
5377  021402                          ENDTST
5378  021402                          L10072:
5379  021402  104401                          TRAP    C$ETST
5380
5381  021404                                  BADHEAD
5382                                          ;*************** TEST 25 ****************
5383                                          ;*
5384                                          ;*MAIN MEMORY PAGE DUAL ADDRESS TEST.
5385                                          ;*IN THIS TEST WE WILL VERIFY THAT PAGES DO
5386                                          ;*NOT DUAL ADDRESS.  THIS TEST IS DIFFERENT FROM THE
5387                                          ;*PREVIOUS DUAL ADDRESS TESTS IN THAT THE OTHER
5388                                          ;*TEST REALLY DIDN'T CHECK PAGE DUAL ADDRESSING
5389  021404                                  BADHEAD
5390                                          ;*************** TEST 25 ****************
5391
5392  021404                          BGNTST
5393  021404                          T25::
5394  021404                                  K4ONLY                  ;FOR 4K CPUS ONLY.
5395                                          ;DO NOT DO TEST IF M8200, OR M8204
5396  021414  104432                          TRAP    C$EXIT
5397  021416  000156                          .WORD   L10073-.
5398  021420                                  MYINT
5399  021420  013701  002516                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
5400  021424                                  MSTCLR
5401  021430  005002                          CLR R2                  ;R2 WILL BE PAGE #
5402  021432  042737  000037  021456  1$:     BIC #37,2$              ;CLEAR UNUSED BITS
5403  021440  050237  021456                  BIS R2,2$               ;ADD CURRENT PAGE MARKER.
5404  021444                                  ROMCLK                  ;SET ADDR D
5405  021444  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
5406  021450  010000                          10000
5407  021452                                  ROMCLK                  ;OF PAGE X
5408  021452  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
5409  021456  004000                  2$:     4000                    ;THIS LOCATION MODIFIED BY LOST
5410                                                                  ;FEW INSTRUCTIONS
5411  021460  010261  000004                  MOV R2,4(R1)            ;PUT PAGE # INTO PART 4
5412  021464                                  ROMCLK                  ;CLOCK PART 4 INTO MEMORY
5413  021464  004537  003044                  JSR     R5,.ROMCLK      ;CLOCK INSTRUCTION
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS

 5414  021470  122500                          122500                  ;WHOSE PAGE # IS IN R2
 5415  021472  005202                          INC R2                  ;UPDATE PAGE #
 5416  021474  032702  000020                  BIT #20,R2              ;DONE ALL PAGES?
 5417  021500  001754                          BEQ 1$                  ;NO-DO NEXT ONE
 5418
 5419                                                                  ;OK, ALL LOADS OF ALL PAGES
 5420                                                                  ;CONTAIN THIER PAGE NUMBER, NOW
 5421                                                                  ;WE'LL GO BACK THROUGH
 5422                                                                  ;THEM CHECKING THEM OUT.
 5423
 5424  021502  005002                          CLR R2                  ;R2 STILL HAS PAGE NUMBER
 5425
 5426  021504  042737  000037  021522  3$:     BIC #37,4$
 5427  021512  050237  021522                  BIS R2,4$
 5428  021516                                  ROMCLK                  ;LOAD PAGE NUMBER
 5429  021516  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
 5430  021522  004000                  4$:     4000
 5431  021524                                  ROMCLK                  ;MOVE MEM TO PART 4
 5432  021524  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
 5433  021530  041224                          041224
 5434  021532  116104  000004                  MOVB 4(R1),R4           ;'FOUND'
 5435  021536  110205                          MOVB R2,R5              ;'EXPECTED'
 5436  021540  120504                          CMPB R5,R4              ;ADDRESS PROBLEM?
 5437  021542  001406                          BEQ 5$
 5438
 5439  021544                                  ERROR 13                ;PAGE ADDRESSING ERROR IN MAIN
 5440  021550  104455                          TRAP    C$ERDF
 5441  021552  000015                          .WORD   13
 5442  021554  005055                          .WORD   EM0
 5443  021556  007542                          .WORD   ERR13
 5444                                                                  ;MEMORY.
 5445                                                                  ;MAR BITS 8 THROUGH 12 ARE REPRESENTED
 5446                                                                  ;BY R2 ('EXP'ED')BITS 0-4)
 5447
 5448  021560                          5$:     ESCAPE TST
 5449  021560  104410                          TRAP    C$ESCAPE
 5450  021562  000012                          .WORD   L10073-.
 5451  021564  005202                          INC R2                  ;UPDATE PAGE ADDRESS
 5452  021566  032702  000020                  BIT #20,R2              ;ALL DONE?
 5453  021572  001744                          BEQ 3$                  ;NO-CHECK NEXT PAGE.
 5454                                                                  ;YES-EXIT.
 5455  021574                          ENDTST
 5456  021574                          L10073:
 5457  021574  104401                          TRAP    C$ETST
 5458
 5459
 5460  021576                                  BADHEAD
 5461                                          ;*************** TEST 26 ****************
 5462                                          ;*
 5463                                          ;*JUMP FIELD,PAGE TEST
 5464                                          ;*
 5465                                          ;*IN THIS TEST WILL MAKE SURE A JUMP FIELD INSTRUCTION
 5466                                          ;*WORKS.  TO DO THIS, WE'LL PUT THE DESIRED PAGE,FIELD
 5467                                          ;*INORMATION IN IBUS*<13> THEN ISSUE A JUMP FIELD
 5468                                          ;*THEN WE'LL READ PC REG. AND VERIFY.
 5469  021576                                  BADHEAD
```

```
5470                                        ;*************** TEST 26 ***************
5471
5472  021576                        BGNTST
5473  021576                        T26::
5474  021576                              K4ONLY                  ;FOR 4K CPUS ONLY
5475                                      ;DO NOT DO TEST IF M8200, OR M8204
5476  021606  104432                      TRAP    C$EXIT
5477  021610  000132                      .WORD   L10074-.
5478  021612                              MYINT
5479  021612  013701  002516              MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
5480  021616                              MSTCLR
5481
5482  021622  005002                      CLR R2                          ;R2 TO CONTAIN FIELD #
5483
5484  021624  042737  000017  021642 1$:  BIC #17,2$                      ;CLEAR ANY JUNK
5485  021632  050237  021642              BIS R2,2$                       ;SET FIELD # INTO INSTR.
5486
5487  021636                              ROMCLK                          ;CLOCK FIELD BITS INTO BREG.
5488  021636  004537  003044              JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
5489  021642  000400              2$:     000400                          ;CONTAINS FIELD,PAGE BITS
5490  021644                              ROMCLK                          ;XFERR BREG INTO IBUS*<13>
5491  021644  004537  003044              JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
5492  021650  061233                      061233
5493  021652                              SROMCLK                         ;GET INSTRUCTION CLOCKED.
5494  021652  004537  003100              JSR     R5,..SROMCLK
5495  021656  100000                      100000                          ;BAS FORM FOR JUM FIELD INSTR.
5496
5497
5498  021660  142761  000002  000001      BICB #BIT1,1(R1)                        ;CLEAR ROMI
5499  021666                              ROMCLK                          ;CLOCK NEXT INSTR.
5500  021666  004537  003044              JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
5501  021672  121264                      121264                          ;MOVE IBUS*TO PORT 4
5502  021674  116104  000004              MOVB 4(R1),R4                   ;GET IT.
5503  021700  042704  177760              BIC     #^C<17>,R4
5504  021704  120402                      CMPB R4,R2                      ;FIELD OK?
5505  021706  001407                      BEQ 3$                          ;IF OK GO AHEAD
5506  021710  010205                      MOV     R2,R5
5507  021712                              ERROR 14                        ;CHANGE FIELD INSTRUCTION   ;*** BO
5508  021716  104455                      TRAP    C$ERDF
5509  021720  000016                      .WORD   14
5510  021722  005055                      .WORD   EM0
5511  021724  007664                      .WORD   ERR14
5512                                                                      ;FAILED.  FOR FIELD,PAGE INDICATES
5513                                                                      ;BY "EXPECTED" BITS 0,1,2,3 OF
5514                                                                      ;EXPECTED REPRESENT FIELD BITS.
5515  021726                        3$:     ESCAPE TST
5516  021726  104410                      TRAP    C$ESCAPE
5517  021730  000012                      .WORD   L10074-.
5518
5519
5520  021732  005202                      INC R2                          ;UPDATE TO NEXT FIELD
5521  021734  032702  000020              BIT #20,R2                      ;DONE ALL FIELDS?
5522  021740  001731                      BEQ     1$
5523
5524  021742                        ENDTST
5525  021742                        L10074:
```

```
5526  021742  104401                    TRAP    C$ETST
5527
5528  021744                            BADHEAD
5529                                    ;*************** TEST 27 ****************
5530                                    ;*
5531                                    ;*JUMP TEST, JUMP ALWAYS, JUMP CHANGE FIELD
5532                                    ;*
5533                                    ;*IN THIS TEST, WE WILL CHECK THE ABILITY OF THE
5534                                    ;*MICRO PROCESSOR TO JUMP (BRANCH & ALWAYS INSTRUCTION)
5535                                    ;*TO LOCATIONS, FIELDS FROM OTHER LOCATIONS FIELDS.
5536                                    ;*WE ALREADY KNOW THAT THE BRANCH INSTR WORKS FROM
5537                                    ;*OTHER TEST.                PROCEDURE:
5538                                    ;*      1. START ADDR 0, FIELD 0
5539                                    ;*      2. **CALCULATE NEW ADDRS, FIELD  VIA INC,
5540                                    ;*      3. CAUSE JUMP (BRANCH) TO NEW ADDRESS
5541                                    ;*      4. READ PC FROM IBUS*12 AND IBUS*13
5542                                    ;*      5. REPEAT STEP 2-4 256.TIMES
5543                                    ;*
5544                                    ;*      TO CALCULATE NEW ADDRESS:
5545                                    ;*      1. INC LOW BYTE OF ADDRESS FOR PC ADDRES 0-7
5546                                    ;*      2. INC LOW BYTE OF NADDRESS FOR PC ADDRESS 8-11
5547                                    ;*         BITS REPRESENTED AS BITS 0-3. WHEN  0-3 OVERFLOWS,
5548                                    ;*         RESTARTS AT ZERO.
5549                                    ;*         NET RESULT IS JUMPS FROM:
5550                                    ;*              FIELD,PAGE              LOC
5551                                    ;*                   0                  0
5552                                    ;*                   1                  1
5553                                    ;*                   2                  2
5554                                    ;*                   3                  3
5555                                    ;*                  10                  7
5556                                    ;*                  11                 11
5557                                    ;*                  :TO                 :
5558                                    ;*                  17                377
5559                                    ;*
5560  021744                            BADHEAD
5561                                    ;*************** TEST 27 ****************
5562
5563  021744                            BGNTST
5564  021744                    T27::
5565  021744                            MYINT
5566  021744  013701  002516            MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
5567  021750                            K4ONLY                          ;4K CPUS ONLY.
5568                                    ;DO NOT DO TEST IF M8200, OR M8204
5569  021760  104432                    TRAP    C$EXIT
5570  021762  000336                    .WORD   L10075-.
5571  021764                            MSTCLR
5572
5573  021770  012737  000000  002406    MOV     #0, FLAG        ;FLAG TO REPRESENT
5574                                                            ;FIELD,PAGE
5575                                                            ;TO VARIE STARTING PAGE,FIELD,
5576                                                            ;CHANGE #0 PORTION OF INSTR.
5577  021776  012702  000000            MOV     #0, R2          ;R2 TO CONTAIN JUMPED
5578                                                            ;TO CHANGE STARTING IMM ADDR.,
5579                                                            ;VARIE #0 PORTIONS OF INSTR.
5580  022002  012737  000000  002412    MOV     #0, FADR        ;ADDRESS
5581                                    ;LOOP HERE
```

```
5582  022010
5583  022010  042737  000017  022050  1$:     BIC #17,2$              ;CLEAR JUNK FROM FIELD
                                              ;PORTION OF CHANGE FIELD INSTR
5584                                          ;INORDER TO INC, DEC FIELD,PAGE
5585  022016  013700  002406                  MOV FLAG,R0
5586  022022  042700  177760                  BIC #^C<17>,R0
5587  022026  050037  022050                  BIS R0,2$               ;NOW POSITION IN INSTR.
5588  022032  042737  077777  022064          BIC #077777,3$                  ;NOW FOR IMMED. BR INSTR.
5589  022040  050237  022064                  BIS R2,3$               ;NOW ADD IMMEDIATE ADDR
5590
5591
5592
5593  022044                                  ROMCLK
5594  022044  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
5595  022050  000400          2$:             000400                  ;MOVE PAGE,FIELD # TO BREG.
5596  022052                                  ROMCLK
5597  022052  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
5598  022056  061233                          61233                   ;MOV    BREG TO PC HIGH REG.
5599  022060                                  SROMCLK
5600  022060  004537  003100                  JSR     R5,..SROMCLK
5601  022064  100000          3$:             100000                  ;NOW CLOCK IT IN BY JMP FIELD INSTR.
5602
5603  022066                                  ROMCLK                  ;READ PC REG HI
5604  022066  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
5605  022072  121265                          121265
5606  022074                                  ROMCLK                  ;READ PC REG LOW
5607  022074  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
5608  022100  121244                          121244
5609
5610  022102  016104  000004                  MOV 4(R1),R4            ;READ PC REG (NOW IN SEL 4)
5611  022106  042704  170000                  BIC     #170000,R4      ;STRIP FOR ONLY PAGE,FIELD BITS.
5612
5613  022112  013705  022050                  MOV 2$,R5               ;NOW FROM ADDR WE WANTED TO
5614  022116  000305                          SWAB R5                 ;JUMP TO
5615  022120  042705  170377                  BIC #170377,R5          ;CLEAR JUNK
5616  022124  050205                          BIS R2,R5               ;ADD IMMED ADDR
5617  022126                                  SKIP06  5$
5618                                          ;GOTO 5$ IF M8206
5619  022136  105205                          INCB    R5              ;UPDATE ADDR. EXPECTED SENCE THE READ
5620  022140                    5$:
5621                                          ;OF THE IBUS <13> INC THE PC.
5622
5623  022140  020504                          CMP R5,R4               ;JUMP GO OK?
5624  022142  001406                          BEQ 4$                  ;YEA, CONTINUES
5625  022144                                  ERROR 15                ;FAILED TO JUMP PROPERLY.
5626  022150  104455                          TRAP    C$ERDF
5627  022152  000017                          .WORD   15
5628  022154  005055                          .WORD   EM0
5629  022156  010006                          .WORD   ERR15
5630                                          ;"FROM ADDR" REPRESENTS
5631                                          ;THE ADDRESS WE STARTED AT
5632                                          ;"TO ADDR" REPRESENTS WHERE
5633                                          ;WE EXPECTED TO JUMP TO,
5634                                          ;"BAD ADDR" REPRESENTS WHERE
5635                                          ;WE WENT TO.
5636
5637                    .REM %
```

```
                                             15      12       11       10       9 8 7              0
                                             -------------------------------------------------------
                                             !UNUSED !  FIELD           ! PAGE     !IMMED. ADDR    !
                                             !       !   BITS           ! BITS     !               !
                                             -------------------------------------------------------

                                      ;THIS IS A PICTURE OF THE P.C. REG.
                                       BITS 0-7 ARE IN IBUS*<12>
                                       BITS 8-11 ARE IN IBUS*<13>
                                       THEY GOT CLOCK IN THERE VIA JUMPS TAKEN
                                       THE FIELD BITS
                                       ARE IN BIT POSITION 0,1 OF THE INSTRUCTION AT 2$.

                                       3$ WAS THE JUMP ALWAYS INSTRUCTION. THE IMMED. ADDR.
                                       WAS IN 0-7 OF THE JUMP INSTR.  THE PAGE BITS,
                                       PC REG BITS 8,9, WERE IN BITS 11,12 OF THE INSTR.
                                       JUMP INSTRUCTIONS HAVE BEEN CHECKED OUT
                                       BEFORE, SO THE IMPORTANT THING TO REMEMBER TO
                                       WATCH IS THE "FROM ADDR","TO ADDR"

                              %

022160                        4$:     ESCAPE TST
022160  104410                        TRAP    C$ESCAPE
022162  000136                        .WORD   L10075-.
022164  010437  002412                MOV R4,FADR
022170  005237  002406                INC FLAG                  ;UPDATE PAGE,FIELD
022174  105202                        INCB R2                   ;UPDATE JMMED. ADDR
022176  001304                        BNE 1$                    ;LOOP IF NOT DONE.

                                      ;*
                                      ;*CHECK HERE TO SEE IF MASTER CLEAR CLEARS P.C. REG
                                      ;*

022200                                SKIP06  40$
                                      ;GOTO 40$ IF M8206
022210  005737  002470                TST     RUNB
022214  001041                        BNE     40$
022216  005737  002472                TST     RUNINH
022222  001036                        BNE     40$
022224  052711  040000                BIS     #40000,(R1)       ;SET MASTER CLEAR
022230  105761  000001                TSTB    1(R1)
022234  042711  040000                BIC     #40000,(R1)

                                      ;TO RUN THIS SECTION OF CODE YOU MUST TURN SW7 OF SWITCH PACK #E28
                                      ;OFF SO THAT M8207 NOT SELFSTARTING.

022240                                ROMCLK                    ;WE MUST FIRST CLOCK
022240  004537  003044                JSR     R5,.ROMCLK        ;CLOCK INSTRUCTION
022244  121265                        121265                    ;THE PC LATCH REGS
022246                                ROMCLK                    ;BEFORE WE CAN READ THEM
022246  004537  003044                JSR     R5,.ROMCLK        ;CLOCK INSTRUCTION
022252  121244                        121244
022254                                ROMCLK                    ;REG PC REG HI, PUT IN PORT5
022254  004537  003044                JSR     R5,.ROMCLK        ;CLOCK INSTRUCTION
022260  121265                        121265
```

```
5694
5695  022262
5696  022262  004537  003044          ROMCLK                  ;REG PC REG LOW, PUT IN PORT4
5697  022262                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
5698  022266  121244                  121244
5699  022270  005005                  CLR     R5              ;EXPECT ZERO
5700  022272  016104  000004          MOV     4(R1),R4        ;READ PC REG FROM PORT 4&5
5701  022276  042704  170003          BIC     #170003,R4
5702  022302  001406                  BEQ     40$             ;IF CLEARED, EXIT
5703                                                          ;NOTE WE ALSO CLEARED BIT 1 OF THE
5704                                                          ;PC REG, BECAUSE AFTER THE MASTER
5705                                                          ;CLEAR, WE DID TWO INSTRUCTIONS TO
5706                                                          ;READ IT, THUS CAUSING THE PC REG
5707                                                          ;TO GET BUMPED.
5708  022304                          ERROR   45              ;MASTER CLEAR FAILED TO CLEAR
5709  022310  104455                  TRAP    C$ERDF
5710  022312  000055                  .WORD   45
5711  022314  005055                  .WORD   EMO
5712  022316  011124                  .WORD   ERR45
5713                                                          ;PC REG
5714  022320                  40$:
5715  022320                  ENDTST
5716  022320                  L10075:
5717  022320  104401                  TRAP    C$ETST
5718  022322                          BADHEAD
5719                                  ;*************** TEST 28 ***************
5720                                  ;*
5721                                  ;*JUMP TEST, JUMP ALWAYS, JUMP CHANGE FIELD
5722                                  ;*
5723                                  ;*IN THIS TEST, WE WILL CHECK THE ABILITY OF THE
5724                                  ;*MICRO PROCESSOR TO JUMP (BRANCH & ALWAYS INSTRUCTION)
5725                                  ;*TO LOCATIONS, FIELDS FROM OTHER LOCATIONS FIELDS.
5726                                  ;*WE ALREADY KNOW THAT THE BRANCH INSTR WORKS FROM
5727                                  ;*OTHER TEST.                   PROCEDURE:
5728                                  ;*      1. START ADDR 0, FIELD 0
5729                                  ;*      2. **CALCULATE NEW ADDRS, FIELD  VIA DEC,
5730                                  ;*      3. CAUSE JUMP (BRANCH) TO NEW ADDRESS
5731                                  ;*      4. READ PC FROM IBUS*12 AND IBUS*13
5732                                  ;*      5. REPEAT STEP 2-4 256.TIMES
5733                                  ;*
5734                                  ;*      TO CALCULATE NEW ADDRESS:
5735                                  ;*      1. DEC LOW BYTE OF ADDRESS FOR PC ADDRES 0-7
5736                                  ;*      2. DEC LOW BYTE OF NADDRESS FOR PC ADDRESS 8-11
5737                                  ;*         BITS REPRESENTED AS BITS 0-3. WHEN  0-3 OVERFLOWS,
5738                                  ;*         RESTARTS AT ZERO.
5739                                  ;*         NET RESULT IS JUMPS FROM:
5740                                  ;*              FIELD,PAGE              LOC
5741                                  ;*                  0                   0
5742                                  ;*                  17                  377
```

```
                                        ;*                      16              376
                                        ;*                      15              375
                                        ;*                      :TO             :
                                        ;*                      00              000
                                        ;*
                                        BADHEAD
                                        ;*************** TEST 28 ****************

                                        BGNTST
                                T28::
                                        MYINT
022322 013701 002516                    MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
022326                                  K4ONLY                  ;4K CPUS ONLY.
                                        ;DO NOT DO TEST IF M8200, OR M8204
022336 104432                           TRAP    C$EXIT
022340 000216                           .WORD   L10076-.
022342                                  MSTCLR

022346 012737 000000 002406             MOV     #0, FLAG        ;FLAG TO REPRESENT
                                                                ;FIELD,PAGE
                                                                ;TO VARIE STARTING PAGE,FIELD,
                                                                ;CHANGE #0 PORTION OF INSTR.
022354 012702 000000                    MOV     #0, R2          ;R2 TO CONTAIN JUMPED
                                                                ;TO CHANGE STARTING IMM ADDR.,
                                                                ;VARIE #0 PORTIONS OF INSTR.
022360 012737 000000 002412             MOV     #0, FADR        ;ADDRESS
                                        ;LOOP HERE
022366                          1$:
022366 042737 000017 022426             BIC #17,2$              ;CLEAR JUNK FROM FIELD
                                                                ;PORTION OF CHANGE FIELD INSTR
022374 013700 002406                    MOV FLAG,R0             ;INORDER TO INC, DEC FIELD,PAGE
022400 042700 177760                    BIC #^C<17>,R0
022404 050037 022426                    BIS R0,2$               ;NOW POSITION IN INSTR.
022410 042737 077777 022442             BIC #077777,3$                  ;NOW FOR IMMED. BR INSTR.
022416 050237 022442                    BIS R2,3$               ;NOW ADD IMMEDIATE ADDR


022422                                  ROMCLK
022422 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
022426 000400                   2$:     000400                  ;MOVE PAGE,FIELD # TO BREG.
022430                                  ROMCLK
022430 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
022434 061233                           61233                   ;MOV    BREG TO PC HIGH REG.
022436                                  SROMCLK
022436 004537 003100                    JSR     R5,..SROMCLK
022442 100000                   3$:     100000                  ;NOW CLOCK IT IN BY JMP FIELD INSTR.

022444                                  ROMCLK                  ;READ PC REG HI
022444 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
022450 121265                           121265
022452                                  ROMCLK                  ;READ PC REG LOW
022452 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
022456 121244                           121244

022460 016104 000004                    MOV 4(R1),R4            ;READ PC REG (NOW IN SEL 4)
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

```
5799  022464  042704  170000          BIC     #170000,R4      ;STRIP FOR ONLY PAGE,FIELD BITS.
5800
5801  022470  013705  022426          MOV 2$,R5               ;NOW FROM ADDR WE WANTED TO
5802  022474  000305                  SWAB R5                 ;JUMP TO
5803  022476  042705  170377          BIC #170377,R5          ;CLEAR JUNK
5804  022502  050205                  BIS R2,R5               ;ADD IMMED ADDR
5805  022504                          SKIP06  5$
5806                                  ;GOTO 5$ IF M8206
5807  022514  105205                  INCB    R5              ;UPDATE ADDR. EXPECTED SENCE THE READ
5808  022516                  5$:
5809                                                          ;OF THE IBUS <13> INC THE PC.
5810
5811  022516  020504                  CMP R5,R4               ;JUMP GO OK?
5812  022520  001406                  BEQ 4$                  ;YEA, CONTINUES
5813  022522                          ERROR 15                ;FAILED TO JUMP PROPERLY.
5814  022526  104455                  TRAP    C$ERDF
5815  022530  000017                  .WORD   15
5816  022532  005055                  .WORD   EM0
5817  022534  010006                  .WORD   ERR15
5818
5819                                                          ;"FROM ADDR" REPRESENTS
5820                                                          ;THE ADDRESS WE STARTED AT
5821                                                          ;"TO ADDR" REPRESENTS WHERE
5822                                                          ;WE EXPECTED TO JUMP TO,
5823                                                          ;"BAD ADDR" REPRESENTS WHERE
5824                                                          ;WE WENT TO.
5825
5826                  .REM %
5827                                          15      12      11      10      9 8 7           0
5828                                          ----------------------------------------------
5829                                          !UNUSED ! FIELD         ! PAGE    !IMMED. ADDR  !
5830                                          !       !  BITS         ! BITS    !             !
5831                                          ----------------------------------------------
5832
5833                                          ;THIS IS A PICTURE OF THE P.C. REG.
5834                                           BITS 0-7 ARE IN IBUS*<12>
5835                                           BITS 8-11 ARE IN IBUS*<13>
5836                                           THEY GOT CLOCK IN THERE VIA JUMPS TAKEN
5837                                           THE FIELD BITS
5838                                           ARE IN BIT POSITION 0,1 OF THE INSTRUCTION AT 2$.
5839
5840                                           3$ WAS THE JUMP ALWAYS INSTRUCTION. THE IMMED. ADDR.
5841                                           WAS IN 0-7 OF THE JUMP INSTR.  THE PAGE BITS,
5842                                           PC REG BITS 8,9, WERE IN BITS 11,12 OF THE INSTR.
5843                                           JUMP INSTRUCTIONS HAVE BEEN CHECKED OUT
5844                                           BEFORE, SO THE IMPORTANT THING TO REMEMBER TO
5845                                           WATCH IS THE "FROM ADDR","TO ADDR"
5846
5847                  %
5848  022536                  4$:     ESCAPE TST
5849  022536  104410                  TRAP    C$ESCAPE
5850  022540  000016                  .WORD   L10076-.
5851  022542  010437  002412          MOV R4,FADR
5852  022546  005337  002406          DEC FLAG                ;UPDATE PAGE,FIELD
5853  022552  105302                  DECB R2                 ;UPDATE JMMED. ADDR
5854  022554  001304                  BNE 1$                  ;LOOP IF NOT DONE.
```

```
                                        ENDTST
                                        L10076:
                                                TRAP    C$ETST
                                                BADHEAD
                                                ;*************** TEST 29 ***************
                                                ;*
                                                ;* IN THIS TEST WE'LL VERIFY THAT THE Z BIT CAN BE READ FROM
                                                ;* IBUS*<13>. WE ALLREADY KNOW THAT THE Z BIT WORKS PROPERLY,
                                                ;* ALL WE WANT TO KNOW HERE IS THAT IT CAN BE READ.
                                                ;*
                                                BADHEAD
                                                ;*************** TEST 29 ***************

                                                BGNTST
                                        T29::
                                                K4ONLY                          ;M8206 &M8207 ONLY!
                                                ;DO NOT DO TEST IF M8200, OR M8204
                                                TRAP    C$EXIT
                                                .WORD   L10077-.
                                                MSTCLR
                                                MYINT
022600 013701 002516                            MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
022604 004737 003166                            JSR     PC,CLRALL               ;CLR CONDITION CODES.
                                                ROMCLK                          ;NOW READ IBUS*<15>PUT IN PORT 4
022610 004537 003044                            JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
022614 121264                                   121264
022616 116104 000004                            MOVB    4(R1),R4                ;READ IT FROM PORT 4
022622 042704 177477                            BIC     #177477,R4              ;STRIP ANY JUNK,C&Z BITS 6,7
022626 012705 000000                            MOV     #0,R5                   ;EXPECT IT CLEAR
022632 120405                                   CMPB    R4,R5                   ;OK?
022634 001410                                   BEQ     1$
022636                                          ERROR   16                      ;FAILURE OF Z&C TO BE CLEAR.
022642 104455                                   TRAP    C$ERDF
022644 000020                                   .WORD   16
022646 005055                                   .WORD   EM0
022650 010134                                   .WORD   ERR16

022652                                          ESCAPE TST
022652 104410                                   TRAP    C$ESCAPE
022654 000116                                   .WORD   L10077-.
022656 004737 003312                    1$:     JSR     PC,SETZ                 ;SET Z BIT.
022662                                          ROMCLK                          ;NOW GO BACK AND CHECK Z BIT SET.
022662 004537 003044                            JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
022666 121264                                   121264

022670 016104 000004                            MOV     4(R1),R4                ;GET INFO.
022674 042704 177477                            BIC     #^C<300>,R4             ;STRIP FOR C&Z BITS.
022700 012705 000200                            MOV     #200,R5                 ;EXPECT ONLY Z BIT SET.
022704 120405                                   CMPB    R4,R5                   ;SET OK?
022706 001410                                   BEQ     2$
022710                                          ERROR   16                      ;Z BIT FAILED TO SET PROPERLY.
022714 104455                                   TRAP    C$ERDF
022716 000020                                   .WORD   16
022720 005055                                   .WORD   EM0
```

```
5911  022722  010134                        .WORD   ERR16
5912
5913  022724                                ESCAPE  TST
5914  022724  104410                        TRAP    C$ESCAPE
5915  022726  000044                        .WORD   L10077-.
5916  022730  004737  003166        2$:     JSR     PC,CLRALL               ;NOW TRY TO CLEAR Z BIT.
5917  022734                                ROMCLK
5918  022734  004537  003044                JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
5919  022740  121264                        121264
5920  022742  016104  000004                MOV     4(R1),R4
5921  022746  042704  177477                BIC     #^C<300>,R4             ;STRIP FOR C&Z BITS
5922  022752  001407                        BEQ     3$                      ;IF ZERO,WE'RE OK
5923  022754  005005                        CLR     R5                      ;ELSE REPORT ERROR
5924  022756                                ERROR   16                      ;Z BIT FAILED TO CLEAR PROPERLY.
5925  022762  104455                        TRAP    C$ERDF
5926  022764  000020                        .WORD   16
5927  022766  005055                        .WORD   EM0
5928  022770  010134                        .WORD   ERR16
5929  022772                        3$:
5930  022772                                ENDTST
5931  022772                        L10077:
5932  022772  104401                        TRAP    C$ETST
5933                                ;FINDFAST
5934  022774                                BADHEAD
5935                                        ;*************** TEST 30 ***************
5936                                        ;*
5937                                        ;* IN THIS TEST WE'LL VERIFY THAT THE C BIT CAN BE READ FROM
5938                                        ;* IBUS*<13>. WE ALLREADY KNOW THAT THE C BIT WORKS PROPERLY,
5939                                        ;* ALL WE WANT TO KNOW HERE IS THAT IT CAN BE READ.
5940                                        ;*
5941  022774                                BADHEAD
5942                                        ;*************** TEST 30 ***************
5943
5944  022774                                BGNTST
5945  022774                        T30::
5946  022774                                K4ONLY                          ;M8206 &M8207 ONLY!
5947                                        ;DO NOT DO TEST IF M8200, OR M8204
5948  023004  104432                        TRAP    C$EXIT
5949  023006  000200                        .WORD   L10100-.
5950  023010                                MSTCLR
5951  023014                                MYINT
5952  023014  013701  002516                MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
5953  023020  004737  003166                JSR     PC,CLRALL               ;CLR CONDITION CODES.
5954  023024                                ROMCLK                          ;NOW READ IBUS*<13>PUT IN PORT 4
5955  023024  004537  003044                JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
5956  023030  121264                        121264
5957  023032  116104  000004                MOVB    4(R1),R4                ;READ IT FROM PORT 4
5958  023036  042704  177477                BIC     #177477,R4              ;STRIP ANY JUNK,C&Z BITS 6,7
5959  023042  012705  000000                MOV     #0,R5                   ;EXPECT IT CLEAR
5960  023046  120405                        CMPB    R4,R5                   ;OK?
5961  023050  001410                        BEQ     1$
5962  023052                                ERROR   16                      ;FAILURE OF Z&C TO BE CLEAR.
5963  023056  104455                        TRAP    C$ERDF
5964  023060  000020                        .WORD   16
5965  023062  005055                        .WORD   EM0
5966  023064  010134                        .WORD   ERR16
```

```
                                                ESCAPE TST
                                                TRAP    C$ESCAPE
                                                .WORD   L10100-.
                                1$:             JSR     PC,SETC                 ;SET C BIT.
                                                ROMCLK                          ;NOW GO BACK AND CHECK C BIT SET.
                                                JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
                                                121264
                                                MOV     4(R1),R4                ;GET INFO.
                                                BIC     #^C<300>,R4             ;STRIP FOR C&Z BITS.
                                                MOV     #100,R5                 ;EXPECT ONLY C BIT SET.
                                                CMPB    R4,R5                   ;SET OK?
                                                BEQ     2$
                                                ERROR   16                      ;C BIT FAILED TO SET PROPERLY.
                                                TRAP    C$ERDF
                                                .WORD   16
                                                .WORD   EM0
                                                .WORD   ERR16

                                                ESCAPE  TST
                                                TRAP    C$ESCAPE
                                                .WORD   L10100-.
                                2$:             JSR     PC,CLRALL               ;NOW TRY TO CLEAR C BIT.
                                                ROMCLK
                                                JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
                                                121264
                                                MOV     4(R1),R4
                                                BIC     #^C<300>,R4             ;STRIP FOR C&Z BITS
                                                BEQ     3$                      ;IF ZERO,WE'RE OK
                                                CLR     R5                      ;ELSE REPORT ERROR
                                                ERROR   16                      ;C BIT FAILED TO CLEAR PROPERLY.
                                                TRAP    C$ERDF
                                                .WORD   16
                                                .WORD   EM0
                                                .WORD   ERR16
                                3$:
                                                ENDTST
                                L10100:
                                                TRAP    C$ETST
                                                BADHEAD
                                                ;*************** TEST 31 ***************
                                                ;*TEST OF PROGRAM CLOCK BIT
                                                ;*DO A MASTER CLEAR, VERIFY THAT PROGRAM CLOCK IS SET
                                                ;*WRITE PROGRAM CLOCK BIT TO A ONE, VERIFY THAT IT CLEARS,
                                                ;*AND THEN SETS SOME TIME LATER
                                                BADHEAD
                                                ;*************** TEST 31 ***************

                                BGNTST
                                T31::
                                                MYINT
                                                MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
                                                MSTCLR                  ;MASTER CLEAR M8200,4,7
                                                CLR     TEMP            ;PREPARE FOR
                                                CLR     STMP0           ;DELAY
                        000004  1$:             MOV     #20,4(R1)       ;LOAD PORT 4
```

```
6023  023236  152761  000002  000001          BISB    #BIT1,1(R1)       ;SET ROMI
6024  023244  012761  121111  000006          MOV     #121111,6(R1)     ;SEL6  INSTRUCTION
6025  023252  152761  000003  000001          BISB    #BIT1!BIT0,1(R1);SET CLOCK BIT
6026  023260  012761  121224  000006          MOV     #121224,6(R1)     ;LOAD NEXT INSTRUCTION
6027  023266  152761  000003  000001          BISB    #BIT1!BIT0,1(R1);READ CLOCK BIT
6028  023274  142761  030001  000001          BICB    #BIT!BIT0,1(R1);CLEAR MAINT BITS
6029  023302  016104  000004                  MOV     4(R1),R4          ;PUT "FOUND" IN R4
6030  023306  005037  002452                  CLR     $GDDAT            ;PUT "EXPECTED" IN $GDDAT
6031  023312  123704  002452                  CMPB    $GDDAT,R4                 ;IS PGM CLOCK CLEAR?
6032  023316  001406                          BEQ     2$
6033  023320  013702  002452                  MOV     $GDDAT,R2
6034  023324                                  ERRDF   50,EMB50                  ;ERROR, PGM CLOCK IS NOT CLEAR
6035  023324  104455                          TRAP    C$ERDF
6036  023326  000062                          .WORD   50
6037  023330  005545                          .WORD   EMB50
6038  023332  000000                          .WORD   0
6039  023334                          2$:
6040  023334                                  ROMCLK                    ;NEXT WORD IS INSTRUCTION,
6041  023334  004537  003044                  JSR     R5,..ROMCLK       ;CLOCK INSTRUCTION
6042  023340  121224                          121224                    ;PORT4 LU11
6043  023342  122761  000020  000004          CMPB    #20,4(R1)         ;IS PGM CLOCK SET?
6044  023350  001420                          BEQ     3$                ;BR IF YES
6045  023352  005237  002440                  INC     TEMP              ;INCREMENT DELAY
6046  023356  005537  002444                  ADC     $TMP0             ;INCREMENT DELAY
6047  023362  022737  000006  002444          CMP     #6,$TMP0          ;IS DELAY DONE
6048  023370  001361                          BNE     2$                ;BR IF NO
6049  023372  012702  000006                  MOV     #6,R2
6050  023376  013704  002444                  MOV     $TMP0,R4
6051  023402                                  ERRDF   51,EMB51                  ;ERROR PGM CLOCK NOT SET
6052  023402  104455                          TRAP    C$ERDF
6053  023404  000063                          .WORD   51
6054  023406  005577                          .WORD   EMB51
6055  023410  000000                          .WORD   0
6056  023412                          3$:
6057
6058  023412  122737  000007  002414          CMPB    #7,WTYPE          ; ONLY DO NEXT TEST IF M8207
6059  023420  001013                          BNE     4$                ; EXIT IF NOT.
6060
6061  023422  005737  002444                  TST     $TMP0             ; IF ANY LARGE COUNT, WE'RE OK
6062  023426  001010                          BNE     4$                ; THEN EXIT
6063
6064  023430  042737  000007  002440          BIC     #7,TEMP           ; CLEAR OUT ANY SMALL COUNT
6065  023436  001004                          BNE     4$                ; IF LARGE COUNT LEFT OVER, WE'RE OK.
6066
6067  023440                                  ERRDF   100,EMB1          ; ERROR
6068  023440  104455                          TRAP    C$ERDF
6069  023442  000144                          .WORD   100
6070  023444  005410                          .WORD   EMB1
6071  023446  000000                          .WORD   0
6072                                                                    ; TIME TTOOO SHORT FOR CLOCK.  MUST BE
6073                                                                    ; DEFECTIVE CAPACITOR IN TIMEING CIRCUIT.
6074  023450                          4$:
6075
6076  023450                          ENDTST
6077  023450                          L10101:
6078  023450  104401                          TRAP    CSETST
```

```
                                        BADHEAD
                                        ;*************** TEST 32 ***************
                                        ;*FORCE POWER FAIL TEST
                                        ;*SET FORCE POWER FAIL BIT VERIFY THAT PROCESSOR TRAPS TO 24
                                        ;*GOING DOWN AND COMING UP. VERIFY ALSO THAT BUS INIT WAS
                                        ;*BLOCKED FROM GETTING TO THE M8200,4,7 DURING THE POWER FAIL
                                        ;*THIS TEST WILL TAKE LONGER THAN 2 SECONDS TO RUN. THIS TEST
                                        ;*SHOULD NOT BE RUN IF YOU HAVE VOLATILABLE MEMORY IN YOUR SYSTEM.
                                        BADHEAD
                                        ;*************** TEST 32 ***************

                                BGNTST
                                T32::
                                        BRESET                  ;STALL FOR TIME
023452  104433                          TRAP    C$RESET
                                        MYINT
023454  013701  002516                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
                                                                ;R1 CONTAINS BASE M8200,4,7 ADDRESS
023460                                  MSTCLR                  ;MASTER CLEAR M8200,4,7
023464  005037  002440                  CLR     TEMP            ;PREPARE FOR DELAY
023470  013737  000024  002444          MOV     @#24,$TMP0      ;SAVE POWER FAIL ADDRESS
023476  013746  000024                  MOV     @#24,-(SP)      ;STORE POWER FAIL ADDRESS
023502  012737  023564  000024          MOV     #1$,@#24        ;SET U FOPR FORCE POWER FAIL
023510  012761  000002  000004          MOV     #2,4(R1)        ;LOAD PORT4
023516  012711  001000                  MOV     #BIT9,(R1)      ;SET ROMI
023522  012761  121111  000006          MOV     #121111,6(R1)   ;LOAD INSTRUCTION
023530  012711  005400                  MOV     #BIT9!BIT8!BIT11,(R1)   ;CLOCK INSTRUCTION
023534  005237  002440          5$:     INC     TEMP            ;WAIT FOR POWER FAIL
023540  001375                          BNE     5$              ;BR IF DELAY NOT DONE
023542                                  MSTCLR
023546                                  ERROR   17      ;ERROR, NO POWER FAIL
023552  104455                          TRAP    C$ERDF
023554  000021                          .WORD   17
023556  005055                          .WORD   EM0
023560  010256                          .WORD   ERR17
023562  000445                          BR      4$
023564  012737  023602  000024  1$:     MOV     #3$,@#24        ;POWER UP ADDRESS
023572  010637  023600                  MOV     SP,2$           ;STORE STACK
023576  000000                          HALT                    ;WAIT FOR POWER UP SEQUENCE
023600  000000                  2$:     0
023602  013706  023600          3$:     MOV     2$,SP           ;RESTORE STACK
023606  012737  024002  000024          MOV     #10$,@#24       ;PUT IN CASE OF FALSE POWER-UP.
023614  005037  024000                  CLR     11$
023620  005237  024000          12$:    INC     11$     ;STALL ON POWER UP.
023624  001375                          BNE     12$             ;WAIT HERE IF BAD,WILL POWER OUT OF HERE.
                                                                ;ELSE PROCEED.

023626                                  POPSP2                  ;POP STACK TWICE2
023630  013701  002516                  MOV     KMCSR,R1
023634  012637  000024                  MOV     (SP)+,@#24      ;RESTORE TRUE POWER FAIL ADDRESS
023640  023737  002444  000024          CMP     $TMP0,@#24      ;IS IT CORRECT?
023646  001413                          BEQ     4$              ;BR IF YES
023650                                  ERROR   17              ;ERROR, STACK IS INCORRECT
023654  104455                          TRAP    C$ERDF
023656  000021                          .WORD   17
```

```
6135  023660  005055                        .WORD   EM0
6136  023662  010256                        .WORD   ERR17
6137  023664  013737  002444  000024        MOV     $TMP0,@#24      ;RESTORE TRUE POWER FAIL ADDRESS
6138  023672  013706  002344                MOV     PSTACK,SP       ;RESTORE STACK
6139  023676  032711  004000          4$:   BIT     #BIT11,(R1)     ;BIT11 STILL SET?
6140  023702  001016                        BNE     7$
6141  023704  005737  002470                TST     RUNB
6142  023710  001013                        BNE     7$
6143  023712  011104                        MOV     (R1),R4
6144  023714  012705  004000                MOV     #BIT11,R5
6145  023720                                ERROR   35              ;OAC FAILED
6146  023724  104455                        TRAP    C$ERDF
6147  023726  000043                        .WORD   35
6148  023730  005055                        .WORD   EM0
6149  023732  010472                        .WORD   ERR35
6150                                                                ;TO PREVENT
6151                                                                ;INIT FROM
6152                                                                ;CLEARING CSR
6153  023734                          EXIT  TST
6154  023734  104432                        TRAP    C$EXIT
6155  023736  000104                        .WORD   L10102-.
6156  023740  012711  003000          7$:   MOV     #BIT9!BIT10,(R1) ;SEL6 = MAINT IR
6157  023744  012705  121111                MOV     #121111,R5      ;R5 = EXPECTED
6158  023750  016104  000006                MOV     6(R1),R4        ;R4 = FOUND
6159  023754  020504                        CMP     R5,R4           ;MAINT IR SHOULD = 12111
6160  023756  001431                        BEQ     6$              ;BR IF OK
6161  023760                                MSTCLR
6162  023764                                ERROR   35              ;IF = 0 THEN BUS INIT WAS
6163  023770  104455                        TRAP    C$ERDF
6164  023772  000043                        .WORD   35
6165  023774  005055                        .WORD   EM0
6166  023776  010472                        .WORD   ERR35
6167                                                                ;NOT BLOCKED FROM CLEARING
6168                                                                ;THE M8200,4,7
6169
6170  024000  000000          11$:  .WORD   0                       ;TEMP COUNT FOR STALL ON POWER UP.
6171
6172  024002  052711  040000  10$:  BIS     #BIT14,(R1)             ;CLR THE THING SO IT CAN'T ASSIRT AC LOW
6173                                                                ;AGAIN!
6174  024006                                MSTCLR
6175  024012                                ERROR   17              ;ERROR  GLIP GAVE US SECOUND UNEXPECTED
6176  024016  104455                        TRAP    C$ERDF
6177  024020  000021                        .WORD   17
6178  024022  005055                        .WORD   EM0
6179  024024  010256                        .WORD   ERR17
6180                                                                ;ASSERTION OF AC LOW ON UNIBUS.
6181                                                                ;FATEL TYPE OF ERROR.
6182  024026  062706  000004                ADD     #4,SP           ;RESTORE STACK.
6183  024032  012637  000024                MOV     (SP)+,@#24
6184  024036                                MSTCLR
6185  024042                          6$:
6186  024042                          ENDTST
6187  024042                          L10102:
6188  024042  104401                        TRAP    C$ETST
6189
6190  024044                                BADHEAD
```

```
                                        ;*************** TEST 33 ****************
                                        ;*MICRO-PROCESSOR NOISE TEST
                                        ;*WRITE ALL ZERO'S THEN ALL ONE'S THEN A DATA PATTTERN
                                        ;*TO THE IBUS* AND IBUS REGISTERS AND TO THE SP AND MAIN MEM
                                        ;*THEN GO BACK AND READ THE DATA PATERNS TO VERIFY THAT
                                        ;*READING AND WRITING OF OTHER LOCATIONS AND REGISTERS
                                        ;*DID NOT CHANGE THE DATA.
024044                                  BADHEAD
                                        ;*************** TEST 33 ****************

024044                          BGNTST
024044                          T33::
024044                                  MYINT
024044  013701  002516                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
024050                                  MSTCLR                          ;MASTER CLEAR M8200,4,7
024054  005002                          CLR     R2                      ;R2 IS INDEX REGISTER
024056  042737  000017  024104  1$:     BIC     #17,2$                  ;CLEAR ADDRESS FIELD
024064  156237  025100  024104          BISB    30$(R2),2$              ;ADD IBUS* REG ADDRESS TO INSTRUCTION
024072  116261  025106  000004          MOVB    31$(R2),4(R1)           ;LOAD PORT4
024100                                  ROMCLK                          ;NEXT WORD IS INSTRUCTION,
024100  004537  003044                  JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
024104  121100                  2$:     121100                          ;WRITE IBUS* REGISTER
024106  005202                          INC     R2                      ;INC INDEX REGISTER
024110  022702  000005                  CMP     #5,R2                   ;DONE YET?
024114  001360                          BNE     1$                      ;BR IF NO
024116  005002                          CLR     R2                      ;R2 IS IBUS REGISTER ADDRESS
024120  042737  000017  024166  3$:     BIC     #17,4$                  ;CLEAR ADDRESS FIELD OF INSTRUCTIONS
024126  042737  000017  024202          BIC     #17,5$
024134  042737  000017  024214          BIC     #17,6$
024142  050237  024166                  BIS     R2,4$                   ;ADD IBUS REG ADDRESS TO INSTRUCTION
024146  050237  024202                  BIS     R2,5$
024152  050237  024214                  BIS     R2,6$
024156  105061  000004                  CLRB    4(R1)                   ;CLEAR PORT4
024162                                  ROMCLK                          ;NEXT WORD IS INSTRUCTION,
024162  004537  003044                  JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
024166  122100                  4$:     122100                          ;WRITE 0 TO IBUS REG
024170  112761  000377  000004          MOVB    #377,4(R1)              ;LOAD PORT4
024176                                  ROMCLK                          ;NEXT WORD IS INSTRUCTION,
024176  004537  003044                  JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
024202  122100                  5$:     122100                          ;WRITE ALL ONES TO  IBUS REG
024204  110261  000004                  MOVB    R2,4(R1)                ;LOAD PORT4
024210                                  ROMCLK                          ;NEXT WORD IS INSTRUCTION,
024210  004537  003044                  JSR     R5,..ROMCLK             ;CLOCK INSTRUCTION
024214  122100                  6$:     122100                          ;WRITE ITS OWN ADDRESS TO IBUS REG
024216  005202                          INC     R2                      ;NEXT ADDRESS
024220  022702  000010                  CMP     #10,R2                  ;DONE YET?
024224  001335                          BNE     3$                      ;BR IF NO
024226  005002                          CLR     R2                      ;START AT SP ADDRESS 0
024230  042737  000017  024276  7$:     BIC     #17,8$                  ;CLEAR ADDRESS FIELD
024236  042737  000017  024312          BIC     #17,9$
024244  042737  000017  024324          BIC     #17,10$
024252  050237  024276                  BIS     R2,8$                   ;ADD ADDRESS TO INSTRUCTION
024256  050237  024312                  BIS     R2,9$
024262  050237  024324                  BIS     R2,10$
024266  105061  000004                  CLRB    4(R1)                   ;CLEAR PORT4
024272                                  ROMCLK                          ;NEXT WORD IS INSTRUCTION,
```

```
6247  024272  004537  003044                 8$:  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6248  024276  123100                               123100                  ;WRITE ZERO TO SP
6249  024300  112761  000377  000004               MOVB    #377,4(R1)      ;LOAD PORT4
6250  024306                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6251  024306  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6252  024312  123100                               123100                  ;WRITE ALL ONES TO SP
6253  024314  110261  000004                 9$:   MOVB    R2,4(R1)        ;LOAD PORT4
6254  024320                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6255  024320  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6256  024324  123100                               123100                  ;WRITE SP ADDRESS TO ITSELF
6257  024326  005202                 10$:  INC     R2                      ;NEXT SP ADDRESS
6258  024330  022702  000020                       CMP     #20,R2          ;DONE YET?
6259  024334  001335                               BNE     7$              ;BR IF NO
6260  024336  005002                               CLR     R2              ;R2 = ,AOM ,E, ADDRESS
6261  024340                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6262  024340  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6263  024344  010000                               010000                  ;MAR _ 0
6264  024346                                       ROMCLK
6265  024346  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6266  024352  004000                               4000
6267  024354  105061  000004                 11$:  CLRB    4(R1)           ;CLEAR PORT4
6268  024360                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6269  024360  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6270  024364  122500                               122500                  ;WRITE ZEROS TO MEM
6271  024366  112761  000377  000004               MOVB    #377,4(R1)      ;LOAD PORT4
6272  024374                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6273  024374  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6274  024400  122500                               122500                  ;WRITE ONES TO MEM
6275  024402  110261  000004                       MOVB    R2,4(R1)        ;LOAD PORT4
6276  024406                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6277  024406  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6278  024412  136500                               136500                  ;WRITE TO MEM IT OWN ADDRESS
6279  024414  005202                               INC     R2              ;NEXT MEM ADDRESS
6280  024416  022702  001000                       CMP     #1000,R2        ;DONE YET?
6281  024422  001354                               BNE     11$             ;BR IF NO
6282
6283                                               ;NOW GO BACK AND READ EVERYTHIN
6284
6285  024424                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6286  024424  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6287  024430  010000                               010000                  ;MAR 0
6288  024432                                       ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6289  024432  004537  003044                       JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6290  024436  004000                               4000                    ;MAR HI _ 0 (M8200,4,7 ONLY)
6291                                               ;WOULD BE CRAM CODE
6292  024440  005737  002432                       TST     TYPE
6293  024444  001452                               BEQ     40$
6294  024446  005005                               CLR     R5              ;R5 IS INDEX REGISTER
6295  024450  042737  000360  024512  12$:  BIC     #360,13$        ;CLEAR ADDRESS FIELD
6296  024456  116502  025100                       MOVB    30$(R5),R2      ;R2 = IBUS* ADDRESS
6297  024462  010203                               MOV     R2,R3           ;PUT IBUS* ADDRESS IN R3
6298  024464  006303                               ASL     R3              ;SHIFT ADDRESS TO BITS 4-7
6299  024466  006303                               ASL     R3
6300  024470  006303                               ASL     R3
6301  024472  006303                               ASL     R3
6302  024474  050337  024512                       BIS     R3,13$          ;ADD ADDRESS TO INSTRUCTION
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS

 6303  024500  116537  025106  002452          MOVB    31$(R5),$GDDAT  ;$GDDAT = "EXPECTED"
 6304  024506                                  ROMCLK                  ;NEXT WORD IS INSTRUCTION,
 6305  024506  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
 6306  024512  121004                  13$:    121004                  ;PORT4 - IBUS* REGISTER
 6307  024514  016104  000004                  MOV     4(R1),R4        ;R4 = "FOUND"
 6308  024520  123704  002452                  CMPB    $GDDAT,R4               ;IBUS* CONTENTS OK?
 6309  024524  001416                          BEQ     20$             ;BR IF YES
 6310  024526  010237  002434                  MOV     R2,MR0
 6311  024532  105037  002453                  CLRB    $GDDAT+1
 6312  024536  013705  002452                  MOV     $GDDAT,R5
 6313  024542                                  ERROR   29              ;IBUS* DATA ERROR
 6314  024546  104455                          TRAP    C$ERDF
 6315  024550  000035                          .WORD   29
 6316  024552  005055                          .WORD   EM0
 6317  024554  010350                          .WORD   ERR29
 6318  024556                                  ESCAPE  TST
 6319  024556  104410                          TRAP    C$ESCAPE
 6320  024560  000334                          .WORD   L10103-.
 6321  024562  005205                  20$:    INC     R5              ;INC COUNTER
 6322  024564  022705  000005                  CMP     #5,R5           ;DONE YET?
 6323  024570  001327                          BNE     12$             ;BR IF NO
 6324
 6325  024572                          40$:
 6326                                  ;END CRAM,GENERAL TESTS
 6327
 6328  024572  005002                          CLR     R2              ;R2 = IBUS REG ADDRESS
 6329  024574  042737  000360  024630  14$:    BIC     #360,15$        ;CLEAR ADDRESS FIELD OF INSTRUCTION
 6330  024602  010203                          MOV     R2,R3           ;R3 = IBUS ADDRESS
 6331  024604  006303                          ASL     R3              ;SHIFT ADDRESS TO BITS 4-7
 6332  024606  006303                          ASL     R3
 6333  024610  006303                          ASL     R3
 6334  024612  006303                          ASL     R3
 6335  024614  050337  024630                  BIS     R3,15$          ;ADD ADDRESS TO INSTRUCTION
 6336  024620  010237  002452                  MOV     R2,$GDDAT               ;$GDDAT = "EXPECTED"
 6337  024624                                  ROMCLK                  ;NEXT WORD IS INSTRUCTION,
 6338  024624  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
 6339  024630  021004                  15$:    021004                  ;PORT4 - IBUS REG
 6340  024632  016104  000004                  MOV     4(R1),R4        ;IBUS = "FOUND"
 6341  024636  123704  002452                  CMPB    $GDDAT,R4               ;IBUS CONTENTS OK?
 6342  024642  001410                          BEQ     21$             ;BR IF YES
 6343  024644  013705  002452                  MOV     $GDDAT,R5
 6344  024650                                  ERROR   29              ;IBUS DATA ERROR
 6345  024654  104455                          TRAP    C$ERDF
 6346  024656  000035                          .WORD   29
 6347  024660  005055                          .WORD   EM0
 6348  024662  010350                          .WORD   ERR29
 6349  024664  005202                  21$:    INC     R2              ;NEXT IBUS REGISTER
 6350  024666  022702  000010                  CMP     #10,R2          ;DONE YET?
 6351  024672  001340                          BNE     14$             ;BR IF NO
 6352  024674  005002                          CLR     R2              ;R2 = SP ADDRESS
 6353  024676  042737  000017  024714  16$:    BIC     #17,17$         ;CLEAR ADDRESS FIELD OF INSTRUCTION
 6354  024704  050237  024714                  BIS     R2,17$          ;ADD ADDRESS TO INSTRUCTION
 6355  024710                                  ROMCLK                  ;NEXT WORD IS INSTRUCTION,
 6356  024710  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
 6357  024714  040600                  17$:    040600                  ;BR - SP
 6358  024716  010237  002452                  MOV     R2,$GDDAT               ;$GDDAT = "EXPECTED"
```

```
6359  024722                        ROMCLK                  ;NEXT WORD IS INSTRUCTION, ROMCLK PC-5304
6360  024722  004537  003044        JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6361  024726  061224                061224                  ;PORT4   BR
6362  024730  016104  000004        MOV     4(R1),R4        ;R4 = 'FOUND'
6363  024734  123704  002452        CMPB    $GDDAT,R4               ;SP CONTENTS OK?
6364  024740  001412                BEQ     22$             ;BR IF YES
6365  024742  013705  002452        MOV     $GDDAT,R5
6366  024746                        ERROR   7               ;SP DATA ERROR
6367  024752  104455                TRAP    C$ERDF
6368  024754  000007                .WORD   7
6369  024756  005055                .WORD   EM0
6370  024760  007026                .WORD   ERR7
6371  024762                        ESCAPE  TST
6372  024762  104410                TRAP    C$ESCAPE
6373  024764  000130                .WORD   L10103-.
6374  024766  005202        22$:    INC     R2              ;NEXT SP LOCATION
6375  024770  022702  000020        CMP     #20,R2          ;DONE YET?
6376  024774  001340                BNE     16$             ;BR IF NO
6377  024776  005002                CLR     R2              ;R2 = MEMORY ADDRESS
6378  025000                        ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6379  025000  004537  003044        JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6380  025004  010000                010000                  ;MAR    0
6381  025006                        ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6382  025006  004537  003044        JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6383  025012  004000                4000                    ;MAR HI   0 (M8200,4,7 OR FAMILY ONLY)
6384  025014  010237  002452  18$:  MOV     R2,$GDDAT               ;$GDDAT = 'EXPECTED'
6385  025020                        ROMCLK                  ;NEXT WORD IS INSTRUCTION,
6386  025020  004537  003044        JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6387  025024  055224                055224                  ;PORT4   MAIN MEM
6388  025026  016104  000004        MOV     4(R1),R4        ;R4 = 'FOUND'
6389  025032  123704  002452        CMPB    $GDDAT,R4               ;MAIN MEM CONTENTS OK?
6390  025036  001412                BEQ     23$             ;BR IF YES
6391  025040  013705  002452        MOV     $GDDAT,R5
6392  025044                        ERROR   6               ;MAIN MEM DATA ERROR
6393  025050  104455                TRAP    C$ERDF
6394  025052  000006                .WORD   6
6395  025054  005055                .WORD   EM0
6396  025056  006700                .WORD   ERR6
6397  025060                        ESCAPE  TST
6398  025060  104410                TRAP    C$ESCAPE
6399  025062  000032                .WORD   L10103-.
6400  025064  005202        23$:    INC     R2              ;NEXT MEM ADDRESS
6401  025066  022702  001000        CMP     #1000,R2                ;DONE YET?
6402  025072  001350                BNE     18$             ;BR IF NO
6403  025074                        EXIT    TST
6404  025074  104432                TRAP    C$EXIT
6405  025076  000016                .WORD   L10103-.
6406  025100     000     002  003  30$:    .BYTE   0,2,3,5,10
6407  025103     005     010
6408
6409          025106                .EVEN
6410  025106     001     003  004  31$:    .BYTE   1,3,4,6,10
6411  025111     006     010
```

```
                                .EVEN

                                ENDTST
                                L10103:
                                        TRAP    C$ETST

                                        BADHEAD
                                        ;*************** TEST 34 ***************
                                        ;* THIS TEST IS DESIGNED TO MAKE SURE THAT A NODST INSTRUCTION
                                        ;* DOES NOT WRITE INTO PORT B OF THE MULTIPORT RAM.
                                        ;* TO DO THIS,WE'LL PUT A 125 INTO INDAT2,THEN WE'LL PUT A
                                        ;* 125 INTO BOTH SP1 AND BR. LAST WE'LL DO A NODST BR,SUBOC,SP1
                                        ;* IF THERE IS A WRITE INTO PORTB,INDAT2 WILL CONTAIN A 377.
                                        BADHEAD
                                        ;*************** TEST 34 ***************

                                BGNTST
                                T34::
                                        MYINT
025116 013701 002516                    MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
                                        ROMCLK
025122 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
025126 000525                           00525                   ;PUT A 125 INTO BRG.
                                        ROMCLK
025130 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
025134 062221                           062221                  ;NOW INTO OIDAT2
                                        ROMCLK
025136 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
025142 063221                           63221                   ;NOW INTO SP1
                                        ROMCLK
025144 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
025150 060361                           060361                  ;NOW THE 'NODST  BR,SUBOC,SP1"
                                                                ;THE NODST SHOULD NOT MODIFY INDAT2!

                                        ROMCLK
025152 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
025156 020420                           020420                  ;PUT CONTENT OF INDAT2 IN BRG.

                                        ROMCLK
025160 004537 003044                    JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
025164 061220                           061220                  ;PUT BRG INTO BSEL0

025166 111104                           MOVB    (R1),R4         ;SEE WHAT CAME BACK,
025170 012705 000125                    MOV     #125,R5         ;SHOULD BE 125  IF 377 CAME BACK,
                                                                ;YOU CAN BET THAT THE 'NODST" WROTE
                                                                ;INTO THE MULTIPORT RAM!  WATCH SIGNAL
                                                                ; 'D1 WRITE OUT L"

025174 020405                           CMP     R4,R5           ;NOW LOOK.
025176 001406                           BEQ     10$

025200                                  ERROR   7
025204 104455                           TRAP    C$ERDF
025206 000007                           .WORD   7
025210 005055                           .WORD   EM0
```

```
025212  007026                          .WORD   ERR7
025214                          10$:
025214                                  ENDTST
025214                          L10104:
025214  104401                          TRAP    C$ETST
025216                                  BADHEAD
                                        ;*************** TEST 35 ***************
                                        ;*
                                        ;* EXTENDED CRAM TEST FOR M8206. IN THIS TEST WE WILL LOAD DATA
                                        ;* THROUGHOUT THE CRAM (TEST DATA IS JUST 4K OF DIAG. CODE) AND
                                        ;* THEN READ IT BACK AND VERIFY THAT IT IS CORRECT.
025216                                  BADHEAD
                                        ;*************** TEST 35 ***************
025216                                  BGNTST
025216                          T35::
025216                                  SKIP06  10$                     ;DO TEST ONLY IF IT IS A M8206
                                        ;GOTO 10$ IF M8206
025226                                  EXIT    TST                     ;OTHERWIZE,SKIP TEST.
025226  104432                          TRAP    C$EXIT
025230  000132                          .WORD   L10105-.
025232                          10$:    MYINT
025232  013701  002516                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
025236  012702  012146                  MOV     #ROMMAP,R2              ;GET ADDR. OF LIST.
025242  012711  002000                  MOV     #2000,(R1)              ;SET TO WRITE DATA.
025246  005003                          CLR     R3                      ;CRAM ADDR ZERO.
025250  010361  000004          15$:    MOV     R3,4(R1)                ;SET ADDR.
025254  012261  000006                  MOV     (R2)+,6(R1)             ;WRITE DATA.
025260  020337  002436                  CMP     R3,MEMSZ                ;DONE WHOLE CRAM?
025264  001402                          BEQ     20$                     ;YES,EXIT THIS LOOP.
025266  005203                          INC     R3                      ;NO,UPDAT ADDR.
025270  000767                          BR      15$
025272  005003                  20$:    CLR     R3                      ;NOW WE WILL READ BACK,STARTING AT
                                                                        ;CRAM ADDR. ZERO.
025274  012705  012146                  MOV     #ROMMAP,R5              ;GET ADDR. LIST OF DATA
025300  010361  000004          30$:    MOV     R3,4(R1)                ;SET ADDR.
025304  011502                          MOV     (R5),R2                 ;PUT EXPECTED INTO R2
025306  016104  000006                  MOV     6(R1),R4                ;READ ACCUAL
025312  020204                          CMP     R2,R4                   ;EQUAL?
025314  001411                          BEQ     40$                     ;YES,CONTINUE.
025316  010300                          MOV     R3,R0
025320                                  ERROR   1                       ;ERROR CRAM DATA TEST,DATA
025324  104455                          TRAP    C$ERDF
025326  000001                          .WORD   1
025330  005055                          .WORD   EM0
025332  006032                          .WORD   ERR1
```

```
6524                                                    ;READ NOT DATA THAT WAS WRITTEN.
6525
6526  025334                            ESCAPE TST
6527  025334  104410                    TRAP    C$ESCAPE
6528  025336  000024                    .WORD   L10105-.
6529  025340  020337  002436    40$:    CMP     R3,MEMSZ        ;ALL DONE?
6530  025344  001002                    BNE     50$
6531
6532  025346                            EXIT    TST
6533  025346  104432                    TRAP    C$EXIT
6534  025350  000012                    .WORD   L10105-.
6535
6536  025352  005203            50$:    INC     R3              ;UPDATE ADDR.
6537  025354  062705  000002            ADD     #2,R5
6538  025360  000747                    BR      30$
6539
6540  025362                    ENDTST
6541  025362                    L10105:
6542  025362  104401                    TRAP    C$ETST
6543
6544
6545  025364                            BADHEAD
6546                                    ;*************** TEST 36 ***************
6547                                    ;*
6548                                    ;* THIS TEST LOADS MICRO-CODE INTO A M8206 MCPU THEN EXECTUES IT.
6549                                    ;* THE MICRO CODE IS DESIGNED TO WRITE ALL ONES INTO THE SEL REGS.
6550                                    ;* THIS TEST IS ONLY PERFORMED ON AN M8206.
6551  025364                            BADHEAD
6552                                    ;*************** TEST 36 ***************
6553
6554  025364                            BGNTST
6555  025364                    T36::
6556
6557  025364                            SKIP06  1$              ;ONLY DO THIS TEST IF M8206
6558                                    ;GOTO 1$ IF M8206
6559  025374                            EXIT    TST
6560  025374  104432                    TRAP    C$EXIT
6561  025376  000442                    .WORD   L10106-.
6562
6563  025400                    1$:     MYINT
6564  025400  013701  002516            MOV     KMCSR,R1        ;RECORD DEVICE ADDR.
6565
6566  025404  004537  026006            JSR     R5,LOADER       ;LOAD THE MICRO CODE
6567
6568  025410  000777                    777                     ;MOVE #377,BRG
6569  025412  061220                    061220                  ;MOVE BRG,BSEL0
6570  025414  061222                    061222                  ;MOVE BRG,BSEL2
6571  025416  061223                    061223                  ;MOVE   BRG,BSEL3
6572  025420  061224                    061224                  ;MOVE   BRG,BSEL4
6573  025422  061225                    061225                  ;MOVE   BRG,BSEL5
6574  025424  061226                    061226                  ;MOVE BRG,BSEL6
6575  025426  061227                    061227                  ;MOVE   BRG,BSEL7
6576  025430  123000                    123000                  ;MOVE BSEL0,SP0
6577  025432  101410                    101410                  ;BRANCH BACK ONE UNTIL <>377
6578
6579  025434  000400                    400                     ;MOVE   #0,BRG
```

```
6580  025436  061220                        61220                   ;MOVE BRG,BSEL0
6581  025440  061222                        61222                   ;MOVE BRG,BSEL2
6582  025442  061223                        61223                   ;MOVE BRG,BSEL3
6583  025444  061224                        61224                   ;MOVE BRG,BSEL4
6584  025446  061225                        61225                   ;MOVE BRG,BSEL5
6585  025450  061226                        61226                   ;MOVE BRG,,BSEL6
6586  025452  061227                        61227                   ;MOVE BRG,BSEL7
6587  025454  123000                        123000                  ;MOVE BSEL0,SP0
6588  025456  104022                        104022                  ;BRANCH BACK ONE LOCATION.
6589  025460  177777                        177777
6590
6591  025462  012711  040000                MOV     #040000,(R1)    ;INITIALIZE MCPU
6592  025466  012711  100000                MOV     #100000,(R1)    ;START CPU.
6593
6594  025472  012700  000062                MOV     #50.,R0         ;THE CYCLE TIME ON THE M8206 IS
6595                                                                 ;200NS.  WE ARE ASKING THE MCPU TO
6596                                                                 ;DO 8 INSTRUCTIONS.  WE'LL DELAY
6597                                                                 ;100 PDP11 INSTRUCTIONS
6598                                                                 ;THIS REALLY SHOULD BE PLENTY OF TIME.
6599
6600  025476  005300                20$:     DEC     R0
6601  025500  001376                        BNE     20$
6602
6603  025502  005005                        CLR     R5              ;JUST FOR TYPEOUT.
6604  025504  012705  000377                MOV     #377,R5         ;EXPECT 377
6605  025510  111104                        MOVB    (R1),R4         ;READ MCPU
6606  025512  120405                        CMPB    R4,R5           ;SEE IF OK.
6607  025514  001410                        BEQ     30$
6608
6609  025516                                ERROR   29              ;ERROR! MCPU WAS TO WRITE ALL
6610  025522  104455                        TRAP    C$ERDF
6611  025524  000035                        .WORD   29
6612  025526  005055                        .WORD   EM0
6613  025530  010350                        .WORD   ERR29
6614                                                                 ;ONES INTO BSEL0,BUT INSTEAD FAILED.
6615  025532                                ESCAPE  TST
6616  025532  104410                        TRAP    C$ESCAPE
6617  025534  000304                        .WORD   L10106-.
6618
6619  025536  012705  177777        30$:    MOV     #177777,R5      ;EXPECT ALL ONES
6620  025542  016104  000002                MOV     2(R1),R4        ;RECIEVED
6621  025546  020405                        CMP     R4,R5           ;RECIEVE OK?
6622  025550  001410                        BEQ     40$
6623
6624  025552                                ERROR   29              ;ERROR! MCPU WAS TO WRITE ALL ONES
6625  025556  104455                        TRAP    C$ERDF
6626  025560  000035                        .WORD   29
6627  025562  005055                        .WORD   EM0
6628  025564  010350                        .WORD   ERR29
6629                                                                 ;INTO BSEL 2&3
6630
6631  025566                                ESCAPE  TST
6632  025566  104410                        TRAP    C$ESCAPE
6633  025570  000250                        .WORD   L10106-.
6634
6635  025572  016104  000004        40$:    MOV     4(R1),R4        ;READ BSEL 4&5
```

```
6636  025576  020405                CMP     R4,R5           ;READ OK?
6637  025600  001410                BEQ     50$
6638
6639  025602                        ERROR   29              ;ERROR! FAILED TO WRITE BSEL $85
6640  025606  104455                TRAP    C$ERDF
6641  025610  000035                .WORD   29
6642  025612  005055                .WORD   EM0
6643  025614  010350                .WORD   ERR29
6644                                                        ; TO ALL ONES.
6645  025616                        ESCAPE  TST
6646  025616  104410                TRAP    C$ESCAPE
6647  025620  000220                .WORD   L10106-.
6648
6649  025622  016104  000006  50$:  MOV     6(R1),R4        ;READ BSEL 6&7
6650  025626  020405                CMP     R4,R5           ;READ OK?
6651  025630  001410                BEQ     60$
6652
6653  025632                        ERROR   29              ;ERROR! FAILED TO WRITE BSEL 6&7
6654  025636  104455                TRAP    C$ERDF
6655  025640  000035                .WORD   29
6656  025642  005055                .WORD   EM0
6657  025644  010350                .WORD   ERR29
6658                                                        ; TO ALL ONES.
6659  025646                        ESCAPE  TST
6660  025646  104410                TRAP    C$ESCAPE
6661  025650  000170                .WORD   L10106-.
6662  025652  105011  60$:  CLRB    (R1)            ;SIGNAL MCPU TO WRITE ALL ZEROS.
6663  025654  005005                CLR     R5              ;EXPECT TO READ ALL ZEROS.
6664
6665  025656  005004                CLR     R4
6666  025660  111104                MOVB    (R1),R4         ;READ BSEL0
6667  025662  001410                BEQ     70$             ;EXPECT ZERO.
6668
6669  025664                        ERROR   29              ;MCPU FAILED TO CLEAR BSEL0
6670  025670  104455                TRAP    C$ERDF
6671  025672  000035                .WORD   29
6672  025674  005055                .WORD   EM0
6673  025676  010350                .WORD   ERR29
6674
6675  025700                        ESCAPE  TST
6676  025700  104410                TRAP    C$ESCAPE
6677  025702  000136                .WORD   L10106-.
6678  025704  016104  000002  70$:  MOV     2(R1),R4        ;READ BSEL 2&3
6679  025710  001410                BEQ     80$             ;IF ZERO,OK
6680
6681  025712                        ERROR   29              ;MCPU FAILED TO CLEAR BSEL 2&3
6682  025716  104455                TRAP    C$ERDF
6683  025720  000035                .WORD   29
6684  025722  005055                .WORD   EM0
6685  025724  010350                .WORD   ERR29
6686  025726                        ESCAPE  TST
6687  025726  104410                TRAP    C$ESCAPE
6688  025730  000110                .WORD   L10106-.
6689  025732                80$:
6690  025732  016104  000004        MOV     4(R1),R4        ;READ BSEL 4&5
6691  025736  001410                BEQ     90$
```

```
                        ERROR   29                      ;MCPU FAILED TO CLEAR BSEL 485
                        TRAP    C$ERDF
                        .WORD   29
                        .WORD   EM0
                        .WORD   ERR29
                        ESCAPE  TST
                        TRAP    C$ESCAPE
                        .WORD   L10106-.
                90$:
                        MOV     6(R1),R4                ;READ BSEL 687
                        BEQ     95$

                        ERROR   29                      ;MCPU FAILED TO CLEAR BSEL 687
                        TRAP    C$ERDF
                        .WORD   29
                        .WORD   EM0
                        .WORD   ERR29

                95$:    EXIT    TST
                        TRAP    C$EXIT
                        .WORD   L10106-.


                        ;
                        ;LOADER   SUBROUTINE USED BY THIS TEST TO LOAD MICRO CODE INTO A M8206
                        ;
026006  012711  002000  LOADER: MOV     #2000,(R1)

026012  005000                  CLR     R0

026014  010061  000004  10$:    MOV     R0,4(R1)        ;SET ADDR.
026020  005200                  INC     R0
026022  011561  000006          MOV     (R5),6(R1)      ;WRIITE MICRO CODE.
026026  022527  177777          CMP     (R5)+,#177777   ;SEE IF TERM.
026032  001370                  BNE     10$
026034  005011                  CLR     (R1)
026036  000205                  RTS     R5

026040                  L10106: ENDTST
026040  104401                  TRAP    C$ETST

026042                          BADHEAD
                                ;*************** TEST 37 ***************
                                ;*
                                ;*NEGATIVE ADDRESS TEST.
                                ;*      IN THIS TEST, WE'LL MAKE SURE THAT THE M8207
                                ;*      DOES NOT RESPOND TO AN ADDRESS THAT ISN'T ASSIGNED
                                ;*      TO IT
                                ;*
026042                          BADHEAD
                                ;*************** TEST 37 ***************

026042                          BGNTST
```

```
6748  026042                                T37::
6749  026042                                      MYINT
6750  026042  013701  002516                      MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
6751
6752  026046  012711  000641                      MOV     #641,(R1)       ;PUT A DEFINITE PATTERN IN MCPU.
6753  026052  012737  026130  000004              MOV     #20$,@#4        ;SET UP FOR TRAPS FROM NON-EX.
6754  026060  005037  000006                      CLR     @#6
6755  026064  012702  160000                      MOV     #160000,R2      ;GET STARTING ADDRESS.
6756
6757  026070  022712  000641          10$:        CMP     #641,(R2)       ;SEE IF CONTENTS OF THE ADDRESS
6758                                                                        ;POINTED TO BY R2 EQUALS THE CONTENTS
6759                                                                        ;OF THE MCPU CSR
6760  026074  001420                              BEQ     40$
6761
6762  026076  062702  000002          15$:        ADD     #2,R2           ;UPDATE ADDRESS.
6763  026102  020227  177700                      CMP     R2,#177700      ;DONE?     ;BO
6764  026106  001370                              BNE     10$             ;NO-LOOP
6765
6766  026110  013737  002464  000004  17$:        MOV     SAVE4,@#4       ;RESTORE TRAP CATCHER
6767  026116  013737  002466  000006              MOV     SAVE6,@#6       ;FROM VALUES SAVED BY INIT SECTION
6768  026124                                      EXIT    TST             ;EXIT, ALL DONE
6769  026124  104432                              TRAP    C$EXIT
6770  026126  000052                              .WORD   L10107-.
6771
6772  026130  062706  000004          20$:        ADD     #4,SP           ;SAVE FROM TRAP
6773  026134  000760                              BR      15$             ;LOOP
6774
6775  026136                            40$:      ;*OH NO, WE MAY HAVE A DUAL ADDRESS PROBLEM!
6776
6777  026136  012711  000174                      MOV     #174,(R1)       ;WRITE NEW PATTERN IN MCPU CSR
6778  026142  022712  000174                      CMP     #174,(R2)       ;DID NEW PATTERN SHOW UP IN ADDR?
6779  026146  001403                              BEQ     60$
6780
6781  026150  012711  000641          50$:        MOV     #641,(R1)       ;PUT OLD PATTERN BACK IN MCPU CSR.
6782  026154  000750                              BR      15$             ;LOOP
6783
6784  026156  020102                    60$:      CMP     R1,R2           ;IS THIS THE MCPU ADDRESS?
6785  026160  001773                              BEQ     50$             ;YES-NO ERROR
6786
6787  026162                                      ERROR   40              ;DUAL ADDRESS ERROR
6788  026166  104455                              TRAP    C$ERDF
6789  026170  000050                              .WORD   40
6790  026172  005055                              .WORD   EM0
6791  026174  010636                              .WORD   ERR40
6792  026176  000744                              BR      17$
6793
6794  026200                                      ENDTST
6795  026200                            L10107:
6796  026200  104401                              TRAP    C$ETST
6797
6798  026202                                      BADHEAD
6799                                              ;*************** TEST 38 ****************
6800                                              ;*
6801                                              ;*BYTE ADDRESSING TEST
6802                                              ;*      HERE, WE'RE GOING TO MAKE SURE THAT WE CAN
6803                                              ;*      WRITE INTO ONLY A HIGH OR LOW BYTE OF THE MCPU.
```

```
                                        ;*
                                        BADHEAD
                                        ;*************** TEST 38 ****************

                                        BGNTST
                                T38::
                                        MYINT
026202  013701  002516                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
026206  005061  000002                  CLR     2(R1)                   ;CLEAR CSR
026212  112761  177777  000002          MOVB    #-1,2(R1)               ;WRITE ALL ONES INTO LOW BYTE
                                                                        ;OF CSR
026220  032761  177400  000002          BIT     #177400,2(R1)           ;SEE IF HIGH BYTE GOT WRITTEN
026226  001410                          BEQ     10$

026230                                  ERROR   41                      ;HIGH BYTE GOT WRITTEN INTO ON A LOW BYTE
026234  104455                          TRAP    C$ERDF
026236  000051                          .WORD   41
026240  005055                          .WORD   EMO
026242  010702                          .WORD   ERR41
026244                                  ESCAPE  TST                     ;OPERATION
026244  104410                          TRAP    C$ESCAPE
026246  000040                          .WORD   L10110-.

026250  005061  000002          10$:    CLR     2(R1)
026254  112761  177777  000003          MOVB    #-1,3(R1)               ;WRITE INTO HIGH BYTE
026262  032761  000377  000002          BIT     #377,2(R1)              ;SEE IF LOW BYTE GOT WRITTEN
026270  001406                          BEQ     20$

026272                                  ERROR   42                      ;LOW BYTE GOT WRITTEN INTO ON A
026276  104455                          TRAP    C$ERDF
026300  000052                          .WORD   42
026302  005055                          .WORD   EMO
026304  010744                          .WORD   ERR42
                                                                        ;HIGH BYTE OPERATION.

026306                          20$:
026306                                  ENDTST
026306                          L10110:
026306  104401                          TRAP    C$ETST

026310                                  BADHEAD
                                        ;*************** TEST 39 ****************
                                        ;*
                                        ;*IN THIS TEST WE'RE GOING TO MAKE SURE THAT THE PC
                                        ;*REG COUNTS UP PROPERLY. THE PC REG SHOULD INCREMENT
                                        ;*ONCE AFTER EACH INSTRUCTION.
                                        ;*
026310                                  BADHEAD
                                        ;*************** TEST 39 ****************

026310                                  BGNTST
026310                          T39::
026310                                  SKIP07  10$                     ;ONLY DO IF M8207
                                        ;GOTO 10$ IF M8207
026320                                  EXIT TST
026320  104432                          TRAP    C$EXIT
```

```
6860  026322  000122                    .WORD   L10111-.
6861
6862  026324                    10$:    MYINT
6863  026324  013701  002516            MOV     KMCSR,R1            ;RECORD DEVICE ADDR.
6864  026330                            MSTCLR
6865  026334                            ROMCLK
6866  026334  004537  003044            JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
6867  026340  000400                    400
6868  026342                            ROMCLK
6869  026342  004537  003044            JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
6870  026346  061233                    61233
6871  026350                            SROMCLK
6872  026350  004537  003100            JSR     R5,..SROMCLK
6873  026354  100000                    100000
6874  026356  012705  000001            MOV     #1,R5               ;START AT ZERO
6875
6876  026362                    20$:    ROMCLK                      ;READ PC HIGH REG.
6877  026362  004537  003044            JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
6878  026366  121265                    121265
6879
6880  026370                            ROMCLK                      ;READ PC LOW REG.
6881  026370  004537  003044            JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
6882  026374  121244                    121244
6883  026376  016104  000004            MOV     4(R1),R4            ;GET WHOLE PICTURE
6884  026402  042704  170000            BIC     #170000,R4
6885  026406  020405                    CMP     R4,R5               ;INCREMENT OK?
6886  026410  001410                    BEQ     30$
6887
6888  026412                            ERROR   47                  ;PC FAILED TO INCREMENT PROPERLY
6889  026416  104455                    TRAP    C$ERDF
6890  026420  000057                    .WORD   47
6891  026422  005055                    .WORD   EM0
6892  026424  011264                    .WORD   ERR47
6893                                                                ;SHOULD INCREMENT BY ONE
6894                                                                ;FOR EACH INSTRUCTION.
6895  026426                            ESCAPE  TST
6896  026426  104410                    TRAP    C$ESCAPE
6897  026430  000014                    .WORD   L10111-.
6898
6899  026432  062705  000002    30$:    ADD     #2,R5               ;UPDATE EXPECTED ADDRESS BY 2.
6900  026436  020527  000777            CMP     R5,#777
6901  026442  001347                    BNE     20$
6902
6903  026444                            ENDTST
6904  026444                    L10111:
6905  026444  104401                    TRAP    C$ETST
6906
6907  026446                            BADHEAD
6908                                    ;*************** TEST 40 ***************
6909                                    ;*
6910                                    ;*IN THIS TEST WE'LL MAKE SURE THAT 'BRANCH FIELD H'' DOESN'T
6911                                    ;*GET SUCH HIGH.
6912                                    ;*FIRST WE'LL CLEAR THE PC HIGH REG. THEN WE'LL DO A BRANCH INSTR
6913                                    ;*WITH BAB BITS 11&12 SET. IF PCR BITS 8&9 SET THEN WE'LL KNOW
6914                                    ;*WE WERE SUCCESSFUL IF PCR BITS 8&9 FAIL TO SET, WE'LL KNOW
6915                                    ;*THAT THE MUX SELECTED THE WRONG INPUT TO BE CLOCKED INTO THE PCR.
```

```
6916
6917  026446                        ;*
6918                                BADHEAD
6919                                ;*************** TEST 40 ***************
6920  026446                        BGNTST
6921  026446                  T40::
6922  026446                        SKIP07  10$             ;ONLY DO IF M8207
6923                                ;GOTO 10$ IF M8207
6924  026456                        EXIT TST
6925  026456  104432                TRAP    C$EXIT
6926  026460  000062                .WORD   L10112-.
6927
6928  026462                  10$:  MYINT                   ;INITIALIZE PARAMETERS
6929  026462  013701  002516        MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
6930  026466                        MSTCLR                  ;CLEAR DEVICE.
6931
6932  026472                        ROMCLK                  ;DO A "BRANCH ALWAYS" WITH
6933  026472  004537  003044        JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6934  026476  114400                114400                  ;BAB BITS 11&12 SET THIS SHOULD CLOCK
6935                                                        ;THESE BITS INTO BITS 8&9 OF THE PCR.
6936  026500                        ROMCLK                  ;NOW READ THE PCR HIGH
6937  026500  004537  003044        JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
6938  026504  121265                121265                  ;AND PUT INTO PORT5.
6939                                                        ;REG. BR NO CLK OF BAB BITS
6940  026506  116105  000005        MOVB    5(R1),R5        ;READ THE PCR.
6941  026512  112704  000003        MOVB    #3,R4           ;EXPECT BITS 8,9 TO BE SET.
6942  026516  042705  177774        BIC     #177774,R5      ;;STRIP ANY JUNK                VRG0182
6943  026522  020405                CMP     R4,R5           ;OK?
6944  026524  001406                BEQ     20$
6945
6946  026526                        ERROR   15              ;"BRANCH FIELD H" STUCK HIGH OR
6947  026532  104455                TRAP    C$ERDF
6948  026534  000017                .WORD   15
6949  026536  005055                .WORD   EM0
6950  026540  010006                .WORD   ERR15
6951                                                        ;OTHER PROBLEM IN THIS AREA.
6952
6953  026542                  20$:
6954  026542                        ENDTST
6955  026542                  L10112:
6956  026542  104401                TRAP    C$ETST
6957
6958  026544                        BADHEAD
6959                                ;*************** TEST 41 ***************
6960                                ;*
6961                                ;*IN THIS TEST WE'RE GOING TO MAKE SURE THAT ONLY SP0
6962                                ;*IS SELECTED FOR SOURCE WHEN THE DESTINATION
6963                                ;*IS THE OUTBUS
6964                                ;*FIRST WE'LL WRITE EACH SP ADDRS INTO ITSELF THEN WE'LL
6965                                ;*MOV SP TO OBUS4. THAT SHOULD SELECT
6966                                ;*SP ADDRESS 0. IF ANY OTHER DATA SHOWS UP, WE'LL
6967                                ;*BLAME IT ON THE SELECTION OF A DIFFERENT SCRATCH PAD.
6968  026544                        BADHEAD
6969                                ;*************** TEST 41 ***************
6970
6971  026544                        BGNTST
```

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE TESTS

 6972  026544                         T41::
 6973  026544                               MYINT
 6974  026544  013701  002516               MOV     KMCSR,R1            ;RECORD DEVICE ADDR.
 6975  026550  005005                       CLR     R5                  ;START WITH ADDR-ZERO
 6976
 6977  026552  042737  000017  026574 10$:  BIC     #17,20$             ;STRIP SP ADDR FIELD FROM INSTR
 6978  026560  010561  000004               MOV     R5,4(R1)            ;PUT SP ADDR INTO PORT4.
 6979  026564  050537  026574               BIS     R5,20$              ;ADD SP ADDR TO INSTR.
 6980  026570                               ROMCLK
 6981  026570  004537  003044               JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
 6982  026574  123100                 20$:  123100                      ;WRITE TO SP
 6983  026576  005205                       INC     R5                  ;UPDATE ADDRESS
 6984  026600  120527  000020               CMPB    R5,#20              ;IF NOT THROUGH, REPEAT.
 6985  026604  001362                       BNE     10$
 6986
 6987  026606                               ROMCLK                      ;NOW MOV SP0 TO OBUS* PORT4
 6988  026606  004537  003044               JSR     R5,..ROMCLK         ;CLOCK INSTRUCTION
 6989  026612  061204                       061204                      ;
 6990  026614  116104  000004               MOVB    4(R1),R4            ;READ PORT4 IT S/B ZERO
 6991  026620  001410                       BEQ     30$
 6992  026622  012705  000000               MOV     #0,R5
 6993  026626                               ERROR   43                  ;SP0 NOT SELECTED FOR SOURCE-SEE
 6994  026632  104455                       TRAP    C$ERDF
 6995  026634  000053                       .WORD   43
 6996  026636  005055                       .WORD   EM0
 6997  026640  011006                       .WORD   ERR43
 6998                                                                   ;DISCUSSION IN HEADER.
 6999
 7000  026642                         30$:  ENDTST
 7001  026642                         L10113:
 7002  026642  104401                       TRAP    C$ETST
 7003
 7004  026644                               BADHEAD
 7005                                       ;*************** TEST 42 ***************
 7006                                       ;*
 7007                                       ;*IN THIS TEST WE ARE GOING TO MAKE SURE THAT THE
 7008                                       ;*SIGNAL 'MOV INST H' (AND ITS ASSOC. TRIBS) DOESN'T GET
 7009                                       ;*STUCK HIGH. IN ORDER TO DO THIS WE'LL CLEAR THE PC HIGH REG
 7010                                       ;*PUT KNOWN DATA IN THE BREG AND SP1 THEN WE'LL A BRANCH
 7011                                       ;*WITH CROM BITS 0-3 SET AS WELL AS CROM BIT 9 WITH CROM BITS 8 AND 11 CLEAR.
 7012                                       ;*IF 'MOV INST H' GETS STUCK HIGH, THE PC REG HIGH WILL GET LOADED
 7013                                       ;*WITH THE CONTENTS OF THE ALU
 7014  026644                               BADHEAD
 7015                                       ;*************** TEST 42 ***************
 7016
 7017  026644                               BGNTST
 7018  026644                         T42::
 7019  026644                               SKIP07  10$                 ;ONLY DO IF M8207
 7020                                       ;GOTO 10$ IF M8207
 7021  026654                               EXIT TST                    ;ELSE EXIT
 7022  026654  104432                       TRAP    C$EXIT
 7023  026656  000110                       .WORD   L10114-.
 7024
 7025  026660                         10$:  MYINT                       ;DO INITIAL TEST SET-UP.
 7026  026660  013701  002516               MOV     KMCSR,R1                    ;RECORD DEVICE ADDR.
 7027  026664                               MSTCLR                      ;DO A MASTER CLEAR.
```

```
7028  026670  005737  002470          TST     RUNB
7029  026674  001034                  BNE     20$
7030
7031                                  ;TO RUN THIS SECTION OF CODE YOU MUST TURN SW7 OF SWITCH PACK #E28
7032                                  ;OFF SO THAT M8207 NOT SELFSTARTING.
7033
7034  026676  012761  000002  000004  MOV     #2,4(R1)        ;PUT A 2 INTO SP1
7035  026704                          ROMCLK                  ;PORT4 TO SCRATCH PAD 1
7036  026704  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7037  026710  123101                  123101
7038  026712  012761  000004  000004  MOV     #4,4(R1)
7039  026720                          ROMCLK
7040  026720  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7041  026724  123100                  123100
7042  026726                          ROMCLK                  ;NOW DO A BRANCH ON C-BIT SET
7043  026726  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7044  026732  141201                  141201                  ;BASED ON SP CONTENTS
7045                                                          ;OK-WHAT WE ARE REALLY
7046                                                          ;INTERESTED IN IS SEEING IF THE
7047                                                          ;PC HIGH REG GETS LOADED WITH
7048                                                          ;THE CONTENTS OF THE ALU (2)
7049                                                          ;IF THIS OCCURS, WE CAN PROBABLY
7050                                                          ;SAY THAT 'MOV INSTR" REMAINED
7051                                                          ;HIGH.
7052  026734                          ROMCLK                  ;READ PC HIGH, PUT INTO PORT5
7053  026734  004537  003044          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7054  026740  121265                  121265
7055  026742  116104  000005          MOVB    5(R1),R4        ;READ PC REG HIGH FROM PORT
7056  026746  001407                  BEQ     20$             ;SHOULD BE CLEAR
7057  026750  005005                  CLR     R5
7058
7059  026752                          ERROR   15              ;ERROR-PC REG HIGH S/B CLEAR-SEE HEADER
7060  026756  104455                  TRAP    C$ERDF
7061  026760  000017                  .WORD   15
7062  026762  005055                  .WORD   EM0
7063  026764  010006                  .WORD   ERR15
7064                                                          ;DISCUSSION.
7065
7066  026766                  20$:
7067  026766                          ENDTST
7068  026766                  L10114:
7069  026766  104401                  TRAP    C$ETST
7070
7071  026770                          BADHEAD
7072                                  ;*************** TEST 43 ***************
7073                                  ;*TEST THAT MASTER CLEAR, CLEARS BITS IN THE NPR CONTROL REGISTER AND
7074                                  ;*MICROPROCESSOR MISCELLANEOUS REGISTER-FIRST WE'LL SET THE
7075                                  ;*PRIORITY UP SO THAT WHEN WE SET THE BUS REQUEST BIT THAT IT WON'T BUG US
7076                                  ;*THEN WE'LL SET ALL THE BITS IN BOTH REGS EXCEPT THE
7077                                  ;*NPR REQUEST. WE'LL LOOK TO SEE THAT ALL GOT SET, NEXT
7078                                  ;*WE'LL DO A MASTER CLEAR AND BE SURE THAT THEY ALL
7079                                  ;*CLEAR.
7080  026770                          BADHEAD
7081                                  ;*************** TEST 43 ***************
7082
7083  026770                          BGNTST
```

```
7084  026770
7085  026770
7086  026770  013701  002516          T43::   MYINT
7087  026774                                  MOV     KMCSR,R1                ;RECORD DEVICE ADDR.
7088  027000                                  MSTCLR
7089  027000  012700  000340                  SETPRI  #PRI07          ;DON'T ALLOW INTERRUPTS.
7090  027004  104441                          MOV     #PRI07,R0
7091  027006  012761  177777  000004          TRAP    C$SPRI
7092  027014  042761  000002  000004          MOV     #-1,4(R1)       ;DATA TO BE SET
7093  027022                                  BIC     #2,4(R1)        ;DON'T SET AC LOW!
7094  027022  004537  003044                  ROMCLK
7095  027026  121111                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7096  027030  042761  000400  000004          121111                  ;PUT INTO MISC REG.
7097  027036                                  BIC     #400,4(R1)      ;DON'T SET NPR BIT
7098  027036  004537  003044                  ROMCLK
7099  027042  121130                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7100  027044                                  121130                  ;PUT INTO NPR REG
7101  027044  004537  003044                  ROMCLK
7102  027050  121225                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7103                                          121225                  ;MOV MISC REG (11) TO PORT5
7104  027052
7105  027052  004537  003044                  ROMCLK
7106  027056  121204                          JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7107  027060  012737  146636  002452          121204                  ;MOVE NPR REG (10) TO PORT4
7108  027066  016104  000004                  MOV     #146636,$GDDAT  ;EXPECT ALL TO SET
7109  027072  042704  030140                  MOV     4(R1),R4        ;READ WHAT HAPPEN
7110  027076  023704  002452                  BIC     #030140,R4      ;MASK UNUSED BITS
7111  027102  001410                          CMP     $GDDAT,R4       ;DID ALL BITS GET SET?
7112  027104                                  BEQ     10$             ;YES CONTINUE.
7113  027104  104433                          BRESET
7114  027106                                  TRAP    C$RESET
7115  027112  104455                          ERROR   46              ;SO SORT OF PROBLEM SETTING BITS
7116  027114  000056                          TRAP    C$ERDF
7117  027116  005055                          .WORD   46
7118  027120  011214                          .WORD   EM0
7119                                          .WORD   ERR46
7120  027122                                                          ;IN THE NPR AND/OR MISC REG.
7121  027122  104406                  CKLOOP
7122                                  TRAP    C$CLP1
7123  027124  152761  000100  000001  10$:     BISB    #100,1(R1)      ;SET MASTER CLEAR
7124  027132  142761  000300  000001          BICB    #300,1(R1)      ;CLEAR MASTER CLEAR
7125
7126  027140                                  ROMCLK
7127  027140  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7128  027144  121225                          121225                  ;MOV MISC REG (11) TO PORT5
7129
7130  027146                                  ROMCLK
7131  027146  004537  003044                  JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7132  027152  121204                          121204                  ;MOV NPR REG (10) TO PORT4
7133  027154  016104  000004                  MOV     4(R1),R4        ;READ RESULTS
7134  027160  005037  002452                  CLR     $GDDAT          ;EXPECT ZERO
7135  027164  042704  010140                  BIC     #010140,R4      ;MASK UNUSED BITS
7136  027170  001407                          BEQ     20$             ;IF ALL ZERO, EVERYTHING COOL.
7137
7138  027172                                  ERROR   46              ;MASTER CLEAR FAILED TO CLEAR
7139  027176  104455                          TRAP    C$ERDF
```

```
7140  027200  000056                    .WORD   46
7141  027202  005055                    .WORD   EM0
7142  027204  011214                    .WORD   ERR46
7143                                                ;SOME BITS IN THE NPR AND/OR MISC REGS.
7144  027206                            CKLOOP
7145  027206  104406                    TRAP    C$CLP1
7146
7147  027210                      20$:
7148  027210  012761  000014  000004    MOV     #14,4(R1)       ;NOW WE ARE GOING TO TRY TO
7149  027216                            ROMCLK                  ;SET THE EXT BITS (16&17) IN THE NPR REG.
7150  027216  004537  003044            JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7151  027222  121110                    121110                  ;IF MASTER CLEAR FAILED TO CLEAR ITSELF
7152  027224                            ROMCLK                  ;THEN WE WILL BE UNABLE TO SET
7153  027224  004537  003044            JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7154  027230  121205                    121205                  ;THESE BITS
7155  027232  116104  000005            MOVB    5(R1),R4        ;READ REG
7156  027236  042704  000140            BIC     #140,R4         ;MASK UNUSED BITS
7157  027242  012737  000014  002452    MOV     #14,$GDDAT      ;STORE GOOD
7158  027250  023704  002452            CMP     $GDDAT,R4       ;DID BITS SET?
7159  027254  001407                    BEQ     30$             ;YES-CONTINUE
7160
7161  027256                            ERROR   46              ;MASTER CLEAR FAILED TO CLEAR
7162  027262  104455                    TRAP    C$ERDF
7163  027264  000056                    .WORD   46
7164  027266  005055                    .WORD   EM0
7165  027270  011214                    .WORD   ERR46
7166                                                ;ITSELF, THUS PROHIBITING US FROM
7167                                                ;FURTHER SETTING BITS IN THE NPR REG.
7168  027272                            CKLOOP
7169  027272  104406                    TRAP    C$CLP1
7170
7171  027274                      30$:  BRESET                  ;NOW WE'LL SEE IF A BUS RESET CLEARS
7172  027274  104433                    TRAP    C$RESET
7173                                                ;THESE BITS.
7174  027276  005737  002470            TST     RUNB            ;CAN'T DO THIS
7175  027302  001016                    BNE     40$             ;TEST IF RUN SW SET.
7176  027304                            ROMCLK
7177  027304  004537  003044            JSR     R5,..ROMCLK     ;CLOCK INSTRUCTION
7178  027310  121204                    121204                  ;READ MISC REG
7179  027312  116104  000004            MOVB    4(R1),R4
7180  027316  001410                    BEQ     40$             ;IF ZERO-END TST
7181
7182  027320  005037  002452            CLR     $GDDAT          ;S/B ZERO
7183
7184  027324                            ERROR   46              ;BUS RESET FAILED TO CLEAR NPR REG
7185  027330  104455                    TRAP    C$ERDF
7186  027332  000056                    .WORD   46
7187  027334  005055                    .WORD   EM0
7188  027336  011214                    .WORD   ERR46
7189                                                ;MASTER CLEAR WAS ABLE TO LOOK TO THE
7190                                                ;CIRCUITRY THAT CONVERTS BUS INIT
7191                                                ;TO "CLEAR"
7192
7193  027340                      40$:
7194  027340                            ENDTST
7195  027340                      L10115:
```

CZDMQD.P11 12-JAN-82 09:50 HARDWARE TESTS

7196 027340 104401 TRAP C$ETST

```
CZDMQD.P11    12-JAN-82 09:50          HARDWARE PARAMETER CODING SECTION

                                       .SBTTL HARDWARE PARAMETER CODING SECTION


                                       ;////////////////////////////////////////////////////////////////////////
                                       ;/ THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                       ;/ THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                       ;/ MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                       ;/ INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                       ;/ MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                       ;/ WITH THE OPERATOR.
                                       ;////////////////////////////////////////////////////////////////////////

 027342                                        BGNHRD
 027342  000016                                .WORD L10116-L$HARD/2
 027344                                L$HARD::
 027344                                        GPRMD   WPM,0,0,7,0,7,YES
 027344  000032                                .WORD   T$CODE
 027346  027400                                .WORD   WPM
 027350  000007                                .WORD   7
 027352  000000                                .WORD   T$LOLIM
 027354  000007                                .WORD   T$HILIM
 027356                                        GPRMA   ADDRES,2,0,160000,177776,YES
 027356  001031                                .WORD   T$CODE
 027360  027452                                .WORD   ADDRES
 027362  160000                                .WORD   T$LOLIM
 027364  177776                                .WORD   T$HILIM
                                       ;       GPRMA   VECTOR,4,0,0,674,YES
                                       ;       GPRMD   PRIRTY,6,0,7000,4,7,YES
                                       ;       GPRMD   LNUNIT,10,0,3,0,3,YES
                                       ;       GPRMD   SWPAC1,12,0,377,0,377,YES
                                       ;       GPRMD   SWPAC2,14,0,377,0,377,YES
                                       ;       GPRMD   LOOPBK,16,0,40000,0,1,YES
                                               GPRMD   ISRUN,24,0,7,0,1,YES
 027366                                        .WORD   T$CODE
 027366  012032                                .WORD   ISRUN
 027370  030050                                .WORD   7
 027372  000007                                .WORD   T$LOLIM
 027374  000000                                .WORD   T$HILIM
 027376  000001                                ENDHRD
 027400                                        .EVEN
 027400                                L10116:

 027400  044127  041511  020110        WPM:    .ASCIZ  'WHICH MICRO-CPU? (0=M8200,4=M8204,7=M8207''
 027406  044515  051103  026517
 027414  050103  037525  024040
 027422  036460  034115  030062
 027430  026060  036464  034115
 027436  030062  026064  036467
 027444  034115  030062  000067
 027452  044515  051103  026517  ADDRES: .ASCIZ  /MICRO-CPU  CSR ADDRESS : /
 027460  050103  020125  041440
 027466  051123  040440  042104
 027474  042522  051523  035040
 027502  000040
```

```
7253  027504  044515  051103  026517  VECTOR: .ASCIZ  /MICRO-CPU VECTOR ADDRESS : /
7254  027512  050103  020125  042526
7255  027520  052103  051117  040440
7256  027526  042104  042522  051523
7257  027534  035040  000040
7258  027540  044515  051103  026517  PRIRTY: .ASCIZ  /MICRO-CPU PRIORITY LEVEL : /
7259  027546  050103  020125  051120
7260  027554  047511  044522  054524
7261  027562  046040  053105  046105
7262  027570  035040  000040
7263  027574  044127  041511  020110  LNUNIT: .ASCIZ  /WHICH LINE UNIT (0-3)? 0=NONE,1=M8201,2=M8202,3=M8203 : /
7264  027602  044514  042516  052440
7265  027610  044516  020124  030050
7266  027616  031455  037451  030040
7267  027624  047075  047117  026105
7268  027632  036461  034115  030062
7269  027640  026061  036462  034115
7270  027646  030062  026062  036463
7271  027654  034115  030062  020063
7272  027662  020072     000
7273  027665     123  044527  041524  SWPAC1: .ASCIZ  /SWITCH PACK #1 (DDCMP LINE #) : /
7274  027672  020110  040520  045503
7275  027700  021440  020061  042050
7276  027706  041504  050115  046040
7277  027714  047111  020105  024443
7278  027722  035040  000040
7279  027726  053523  052111  044103  SWPAC2: .ASCIZ  /SWITCH PACK #2 (BM873 BOOT ADR) : /
7280  027734  050040  041501  020113
7281  027742  031043  024040  046502
7282  027750  033470  020063  047502
7283  027756  052117  040440  051104
7284  027764  020051  020072     000
7285  027771     127  046111  020114  LOOPBK: .ASCIZ  /WILL TEST CONNECTOR(S) BE USED ? 0=NO,1=YES : /
7286  027776  042524  052123  041440
7287  030004  047117  042516  052103
7288  030012  051117  051450  020051
7289  030020  042502  052440  042523
7290  030026  020104  020077  036460
7291  030034  047516  030454  054475
7292  030042  051505  035040  000040
7293  030050  044515  051103  026517  ISRUN:  .ASCIZ  'MICRO-PROCESSOR RUN SWITCH  TYPE 0 IF OFF, 1 IF ON :"
7294  030056  051120  041517  051505
7295  030064  047523  020122  052522
7296  030072  020116  053523  052111
7297  030100  044103  020040  054524
7298  030106  042520  030040  044440
7299  030114  020106  043117  026106
7300  030122  030440  044440  020106
7301  030130  047117  035040     000
7302
7303        030136              .EVEN
7304
7305
7306
7307
7308
```

```
                        .SBTTL  SOFTWARE PARAMETER CODING SECTION

                        ;////////////////////////////////////////////////////////////////
                        ;// THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                        ;// THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                        ;// MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                        ;// INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                        ;// MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                        ;// WITH THE OPERATOR.
                        ;////////////////////////////////////////////////////////////////

030136                          BGNSFT
030136  000000                  .WORD L10117-L$SOFT/2
030140                  L$SOFT::

030140                          ENDSFT
                                .EVEN
030140                  L10117:

                        .EVEN

030140                                  LASTAD
                                        .EVEN
030140  000000                          .WORD   0
030142  000000                          .WORD   0
030144                          L$LAST::

000001                  .END
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
ADDRES  027452          7222    7248#
ADR   = 000020 G        1545#
ADR1    012600          3426#   3455
ADR2    012604          3427#   3452
ADR4    012444          3376#   3403
ADR5    012450          3377#   3400
ASSEMB= 000010          1234
BIT0  = 000001 G        1518#   1865    1866    1879    1880    6025    6027    6028
BIT00 = 000001 G        1507#   1518
BIT01 = 000002 G        1506#   1517
BIT02 = 000004 G        1505#   1516
BIT03 = 000010 G        1504#   1515
BIT04 = 000020 G        1503#   1514
BIT05 = 000040 G        1502#   1513
BIT06 = 000100 G        1501#   1512
BIT07 = 000200 G        1500#   1511
BIT08 = 000400 G        1499#   1510
BIT09 = 001000 G        1498#   1509
BIT1  = 000002 G        1517#   1863    1865    1866    1873    1880    5498    6023    6025    6027
BIT10 = 002000 G        1497#   1997    2018    2026    3380    3431    3483    3512    6156
BIT11 = 004000 G        1496#   6106    6139    6144
BIT12 = 010000 G        1495#   3115
BIT13 = 020000 G        1494#   2000    2021    2029    3126    3383    3434    3486
BIT14 = 040000 G        1493#   6172
BIT15 = 100000 G        1492#   3130
BIT2  = 000004 G        1516#   1866    1880
BIT3  = 000010 G        1515#
BIT4  = 000020 G        1514#
BIT5  = 000040 G        1513#   3911
BIT6  = 000100 G        1512#   1850    1851    3934
BIT7  = 000200 G        1511#   1851
BIT8  = 000400 G        1510#   1973    3909    6106
BIT9  = 001000 G        1509#   6104    6106    6156
BOE   = 000400 G        1549#
CLKX    002360          1601#
CLRALL  003166          1886#   3997    4021    4045    4798    4824    4848    4900    4926    4950    5001    5027    5051
                        5102    5128    5152    5204    5230    5254    5305    5331    5355    5879    5916    5953    5989
CRAMA   003602          2027    2036#
CZDMQ   002000 G        1240#
C$AU  = 000052          1234#   3242
C$AUTO= 000061          1234#   3188
C$BRK = 000022          1234#   3590    3652    3743    3836
C$BSEG= 000004          1234#   3379    3429    3481    3506    3994    4020    4044    4096    4119    4142    4193    4218
                        4243    4295    4319    4343    4396    4420    4444    4497    4521    4545    4597    4621    4645
                        4697    4721    4745    4796    4821    4847    4898    4923    4949    5000    5024    5050    5101
                        5125    5151    5203    5227    5253    5304    5328    5354
C$BSUB= 000002          1234#
C$CEFG= 000045          1234#
C$CLCK= 000062          1234#
C$CLEA= 000012          1234#   3204
C$CLOS= 000035          1234#
C$CLP1= 000006          1234#   7121    7145    7169
C$CVEC= 000036          1234#
C$DCLN= 000044          1234#
C$DODU= 000051          1234#   3182
C$DRPT= 000024          1234#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
C$DU   = 000053  1234#  3224
C$EDIT= 000003  1234#  1311
C$ERDF= 000055  1234#  3281  3329  3349  3388  3439  3495  3518  3578  3640  3702  3735  3800
                3825  3872  3917  3940  3959  4009  4033  4057  4108  4131  4154  4206  4231
                4256  4308  4332  4356  4409  4433  4457  4510  4534  4558  4610  4634  4658
                4710  4734  4758  4810  4836  4860  4912  4938  4962  5013  5039  5063  5114
                5140  5164  5216  5242  5266  5317  5343  5367  5440  5508  5626  5709  5814
                5889  5908  5925  5963  5981  5998  6035  6052  6068  6111  6133  6146  6163
                6176  6314  6345  6367  6393  6465  6520  6610  6625  6640  6654  6670  6682
                6694  6706  6788  6819  6833  6889  6947  6994  7060  7115  7139  7162  7185
C$ERHR= 000056  1234#
C$ERRO= 000060  1234#
C$ERSF= 000054  1234#
C$ERSO= 000057  1234#
C$ESCA= 000010  1234#  3272  3289  3335  3393  3444  3500  3523  3583  3645  3707  3740  3833
                3922  4014  4038  4062  4113  4136  4159  4212  4237  4261  4313  4337  4361
                4414  4438  4462  4515  4539  4563  4615  4639  4663  4715  4739  4763  4815
                4841  4865  4917  4943  4967  5018  5044  5068  5119  5145  5169  5221  5247
                5271  5322  5348  5372  5449  5516  5661  5849  5895  5914  5969  5987  6319
                6372  6398  6527  6616  6632  6646  6660  6676  6687  6699  6824  6896
C$ESEG= 000005  1234#  3397  3448  3504  3527  4018  4042  4066  4117  4140  4163  4216  4241
                4265  4317  4341  4365  4418  4442  4466  4519  4543  4567  4619  4643  4667
                4719  4743  4767  4819  4845  4869  4921  4947  4971  5022  5048  5072  5123
                5149  5173  5225  5251  5275  5326  5352  5376
C$ESUB= 000003  1234#
C$ETST= 000001  1234#  3294  3357  3407  3459  3535  3597  3659  3750  3879  3966  4069  4166
                4268  4368  4469  4570  4670  4770  4872  4974  5075  5176  5278  5379  5457
                5526  5717  5859  5932  6005  6078  6188  6417  6473  6542  6734  6796  6842
                6905  6956  7002  7069  7196
C$EXIT= 000032  1234#  3370  3420  3473  3764  3894  3986  4088  4185  4287  4387  4488  4589
                4689  4789  4891  4993  5094  5196  5297  5396  5476  5569  5757  5874  5948
                6154  6404  6489  6533  6560  6712  6769  6859  6925  7022
C$GETB= 000026  1234#
C$GETW= 000027  1234#
C$GMAN= 000043  1234#
C$GPHR= 000042  1234#  3077
C$GPLO= 000030  1234#
C$GPRI= 000040  1234#
C$INIT= 000011  1234#  3162
C$INLP= 000020  1234#
C$MANI= 000050  1234#
C$MEM  = 000031  1234#
C$MSG  = 000023  1234#  2374  2403  2432  2460  2488  2517  2546  2574  2602  2630  2658  2686
                2715  2743  2766  2794  2822  2834  2853  2870  2887  2907  2926  2951  2970
                2990
C$OPEN= 000034  1234#
C$PNTB= 000014  1234#  2352  2358  2366  2371  2381  2387  2395  2400  2410  2416  2424  2429
                2439  2445  2452  2457  2467  2473  2480  2485  2495  2501  2509  2514  2524
                2530  2538  2543  2553  2559  2566  2571  2581  2587  2594  2599  2609  2615
                2622  2627  2637  2643  2650  2655  2665  2671  2678  2683  2693  2699  2707
                2712  2722  2728  2735  2740  2752  2758  2763  2773  2779  2786  2791  2801
                2807  2814  2819  2830  2849  2866  2883  2903  2922  2941  2947  2966  2986
C$PNTF= 000017  1234#  2843  2860  2877  2897  2916  2935  2960  2980
C$PNTS= 000016  1234#
C$PNTX= 000015  1234#
C$QIO  = 000377  1234#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
C$RDBU= 000007          1234#
C$REFG= 000047          1234#  3042  3048  3054  3060
C$RESE= 000033          1234#  3200  3221  6094  7113  7172
C$REVI= 000003          1234#  1310
C$RFLA= 000021          1234#
C$RPT = 000025          1234#  3011
C$SEFG= 000046          1234#
C$SPRI= 000041          1234#  7090
C$SVEC= 000037          1234#
C$TPRI= 000013          1234#
DFPTBL  002262 G        1431#
DH0     005723          2271#  2754
DH1     005724          2274#  2354  2383  2412  2497  2526
DH2     005755          2279#  2441  2469  2555  2583  2611  2639  2667  2724  2775  2803
DH3     005775          2282#  2695
DIAGMC= 000000          1234
EF.CON= 000036 G        1525#  3053
EF.NEW= 000035 G        1526#  3047
EF.PWR= 000034 G        1527#
EF.RES= 000037 G        1524#  3059
EF.STA= 000040 G        1523#  3041
EMB1    005410          2230#  6070
EMB50   005545          2250#  6037
EMB51   005577          2255#  6054
EM0     005055          2188#  3283  3331  3351  3390  3441  3497  3520  3580  3642  3704  3737  3802
                        3827   3874  3919  3942  3961  4011  4035  4059  4110  4133  4156  4208  4233
                        4258   4310  4334  4358  4411  4435  4459  4512  4536  4560  4612  4636  4660
                        4712   4736  4760  4812  4838  4862  4914  4940  4964  5015  5041  5065  5116
                        5142   5166  5218  5244  5268  5319  5345  5369  5442  5510  5628  5711  5816
                        5891   5910  5927  5965  5983  6000  6113  6135  6148  6165  6178  6316  6347
                        6369   6395  6467  6522  6612  6627  6642  6656  6672  6684  6696  6708  6790
                        6821   6835  6891  6949  6996  7062  7117  7141  7164  7187
EM1     005056          2189#  2348  2406
EM10    005260          2214#  2577  2633
EM11    005312          2219#  2661
EM12    005333          2222#  2689
EM16    005353          2225#  2718
EM17    005434          2235#  2748
EM2     005076          2192#  2377
EM27    005463          2239#
EM29    005512          2244#  2769
EM3     005131          2197#  2435
EM35    005434          2234#  2797
EM4     005144          2199#  2463
EM5     005171          2203#  2491
EM6     005216          2207#  2520
EM7     005233          2210#  2549  2605
ENDBUG  003044          1858#
ENDIT   012040          3056   3148  3152  3159#
ERRFLG  002350          1597#
ERR1    006032 G        2347#  3391  3442  3498  6523
ERR10   007154 G        2548#
ERR11   007276 G        2576#  3803  3828
ERR12   007420 G        2604#  3332  3352
ERR13   007542 G        2632#  5443
ERR14   007664 G        2660#  5511
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
ERR15   010006 G    2688#  5629  5817  6950  7063
ERR16   010134 G    2717#  5892  5911  5928  5966  5984  6001
ERR17   010256 G    2747#  6114  6136  6179
ERR2    00616C G    2376#  3521
ERR29   010350 G    2768#  6317  6348  6613  6628  6643  6657  6673  6685  6697  6709
ERR3    006306 G    2405#
ERR35   010472 G    2796#  6149  6166
ERR36   010614 G    2825#  3284
ERR4    006434 G    2434#
ERR40   010636 G    2837#  6791
ERR41   010702 G    2855#  6822
ERR42   010744 G    2872#  6836
ERR43   011006 G    2890#  6997
ERR44   011054 G    2909#  3875
ERR45   011124 G    2928#  5712
ERR46   011214 G    2953#  7118  7142  7165  7188
ERR47   011264 G    2973#  6892
ERR5    006556 G    2462#  4012  4036  4060  4111  4134  4157  4209  4234  4259  4311  4335  4359
                    4412   4436  4460  4513  4537  4561  4613  4637  4661  4713  4737  4761  4813
                    4839   4863  4915  4941  4965  5016  5042  5066  5117  5143  5167  5219  5245
                    5269   5320  5346  5370
ERR6    006700 G    2490#  3581  3643  3705  3738  6396
ERR7    007026 G    2519#  3920  3943  3962  6370  6468
EVL   = 000004 G    1543#
ESEND = 002100      1234#
ESLOAD= 000035      1234#  1334
FADR    002412      1614#  2703  5580* 5663* 5768* 5851*
FLAG    002406      1612#  2503  3551* 3555  3556  3591* 3592  3612* 3617  3618  3653* 3654  3675*
                    3676   3679  3680  3709* 3710  3713* 3714  3717  3744* 3745  5573* 5585  5664*
                    5761*  5773  5852*
FM1     005046      2183#  2349  2355  2378  2384  2407  2413  2436  2442  2464  2470  2492  2498
                    2521   2527  2550  2556  2578  2584  2606  2612  2634  2640  2662  2668  2690
                    2696   2719  2725  2749  2755  2770  2776  2798  2804
FTIME   002462      1634#  3031  3035*
FSAU  = 000015      1234#  3239  3241
FSAUTO= 000020      1234#  3166  3187
FSBGN = 000040      1234#  1240  1354  2347  2376  2405  2434  2462  2490  2519  2548  2576  2604
                    2632   2660  2688  2717  2747  2768  2796  2825  2837  2855  2872  2890  2909
                    2928   2953  2973  3001  3007  3025  3166  3198  3218  3239  3264  3272  3289
                    3293   3307  3335  3356  3367  3370  3379  3393  3406  3417  3420  3429  3444
                    3458   3470  3473  3481  3500  3506  3523  3534  3546  3583  3596  3607  3645
                    3658   3670  3707  3740  3749  3761  3764  3833  3878  3891  3894  3922  3965
                    3982   3986  3994  4014  4020  4038  4044  4062  4068  4085  4088  4096  4113
                    4119   4136  4142  4159  4165  4182  4185  4193  4212  4218  4237  4243  4261
                    4267   4284  4287  4295  4313  4319  4337  4343  4361  4367  4384  4387  4396
                    4414   4420  4438  4444  4462  4468  4485  4488  4497  4515  4521  4539  4545
                    4563   4569  4586  4589  4597  4615  4621  4639  4645  4663  4669  4686  4689
                    4697   4715  4721  4739  4745  4763  4769  4786  4789  4796  4815  4821  4841
                    4847   4865  4871  4888  4891  4898  4917  4923  4943  4949  4967  4973  4990
                    4993   5000  5018  5024  5044  5050  5068  5074  5091  5094  5101  5119  5125
                    5145   5151  5169  5175  5193  5196  5203  5221  5227  5247  5253  5271  5277
                    5294   5297  5304  5322  5328  5348  5354  5372  5378  5393  5396  5449  5456
                    5473   5476  5516  5525  5564  5569  5661  5716  5752  5757  5849  5858  5871
                    5874   5895  5914  5931  5945  5948  5969  5987  6004  6016  6077  6092  6154
                    6187   6202  6319  6372  6398  6404  6416  6430  6472  6485  6489  6527  6533
                    6541   6555  6560  6616  6632  6646  6660  6676  6687  6699  6712  6733  6748
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                    6769  6795  6809  6824  6841  6855  6859  6896  6904  6921  6925  6955  6972
                    7001  7018  7022  7068  7084  7195  7211  7323
FSCLEA= 000007      1234#  3198  3203
FSDU  = 000016      1234#  3218  3223
FSEND = 000041      1234#  1240  2375  2404  2433  2461  2489  2518  2547  2575  2603  2631  2659
                    2687  2716  2744  2767  2795  2823  2835  2854  2871  2888  2908  2927  2952
                    2971  2991  3005  3012  3163  3189  3205  3225  3243  3264  3272  3289  3293
                    3295  3307  3335  3356  3358  3367  3370  3393  3398  3406  3408  3417  3420
                    3444  3449  3458  3460  3470  3473  3500  3505  3523  3528  3534  3536  3546
                    3583  3596  3598  3607  3645  3658  3660  3670  3707  3740  3749  3751  3761
                    3764  3833  3878  3880  3891  3894  3922  3965  3967  3982  3986  4014  4019
                    4038  4043  4062  4067  4068  4070  4085  4088  4113  4118  4136  4141  4159
                    4164  4165  4167  4182  4185  4212  4217  4237  4242  4261  4266  4267  4269
                    4284  4287  4313  4318  4337  4342  4361  4366  4367  4369  4384  4387  4414
                    4419  4438  4443  4462  4467  4468  4470  4485  4488  4515  4520  4539  4544
                    4563  4568  4569  4571  4586  4589  4615  4620  4639  4644  4663  4668  4669
                    4671  4686  4689  4715  4720  4739  4744  4763  4768  4769  4771  4786  4789
                    4815  4820  4841  4846  4865  4870  4871  4873  4888  4891  4917  4922  4943
                    4948  4967  4972  4973  4975  4990  4993  5018  5023  5044  5049  5068  5073
                    5074  5076  5091  5094  5119  5124  5145  5150  5169  5174  5175  5177  5193
                    5196  5221  5226  5247  5252  5271  5276  5277  5279  5294  5297  5322  5327
                    5348  5353  5372  5377  5378  5380  5393  5396  5449  5456  5458  5473  5476
                    5516  5525  5527  5564  5569  5661  5716  5718  5752  5757  5849  5858  5860
                    5871  5874  5895  5914  5931  5933  5945  5948  5969  5987  6004  6006  6016
                    6077  6079  6092  6154  6187  6189  6202  6319  6372  6398  6404  6416  6418
                    6430  6472  6474  6485  6489  6527  6533  6541  6543  6555  6560  6616  6632
                    6646  6660  6676  6687  6699  6712  6733  6735  6748  6769  6795  6797  6809
                    6824  6841  6843  6855  6859  6896  6904  6906  6921  6925  6955  6957  6972
                    7001  7003  7018  7022  7068  7070  7084  7195  7197  7240  7330
FSHARD= 000004      1234#  7211  7238
FSHW  = 000013      1234#  1429  1447
FSINIT= 000006      1234#  3025  3161
FSJMP = 000050      1234#  3005  3370  3420  3473  3764  3894  3986  4088  4185  4287  4387  4488
                    4589  4689  4789  4891  4993  5094  5196  5297  5396  5476  5569  5757  5874
                    5948  6154  6404  6489  6533  6560  6712  6769  6859  6925  7022
FSMOD = 000000      1234#  1240
FSMSG = 000011      1234#  2347  2373  2376  2402  2405  2431  2434  2459  2462  2487  2490  2516
                    2519  2545  2548  2573  2576  2601  2604  2629  2632  2657  2660  2685  2688
                    2714  2717  2742  2747  2765  2768  2793  2796  2821  2825  2833  2837  2852
                    2855  2869  2872  2886  2890  2906  2909  2925  2928  2950  2953  2969  2973
                    2989
FSPROT= 000021      1234#  1354  1359
FSPWR = 000017      1234#
FSRPT = 000012      1234#  3001  3010
FSSEG = 000003      1234#  3379  3396  3429  3447  3481  3503  3506  3526  3994  4017  4020  4041
                    4044  4065  4096  4116  4119  4139  4142  4162  4193  4215  4218  4240  4243
                    4264  4295  4316  4319  4340  4343  4364  4396  4417  4420  4441  4444  4465
                    4497  4518  4521  4542  4545  4566  4597  4618  4621  4642  4645  4666  4697
                    4718  4721  4742  4745  4766  4796  4818  4821  4844  4847  4868  4898  4920
                    4923  4946  4949  4970  5000  5021  5024  5047  5050  5071  5101  5122  5125
                    5148  5151  5172  5203  5224  5227  5250  5253  5274  5304  5325  5328  5351
                    5354  5375
FSSOFT= 000005      1234#  7323  7328
FSSRV = 000010      1234#
FSSUB = 000002      1234#
FSSW  = 000014      1234#  1456  1462
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
F$TEST= 000001          1234#  3265  3293  3308  3356  3368  3406  3418  3458  3471  3534  3547  3596
                        3608   3658  3671  3749  3762  3878  3892  3965  3983  4068  4086  4165  4183
                        4267   4285  4367  4385  4468  4486  4569  4587  4669  4687  4769  4787  4871
                        4889   4973  4991  5074  5092  5175  5194  5277  5295  5378  5394  5456  5474
                        5525   5565  5716  5753  5858  5872  5931  5946  6004  6017  6077  6093  6187
                        6203   6416  6431  6472  6486  6541  6556  6733  6749  6795  6810  6841  6856
                        6904   6922  6955  6973  7001  7019  7068  7085  7195
GETPRM  011462          3062   3071#  3080
G$CNTO= 000200          1234#
G$DELM= 000372          1234#
G$DISP= 000003          1234#
G$EXCP= 000400          1234#
G$HILI= 000002          1234#
G$LOLI= 000001          1234#
G$NO  = 000000          1234#
G$OFFS= 000400          1234#  7215  7221  7232
G$OFSI= 000376          1234#  7215  7221  7232
G$PRMA= 000001          1234#  7221
G$PRMD= 000002          1234#  7215  7232
G$PRML= 000000          1234#
G$RADA= 000140          1234#
G$RADB= 000000          1234#
G$RADD= 000040          1234#
G$RADL= 000120          1234#
G$RADO= 000020          1234#  7215  7221  7232
G$XFER= 000004          1234#
G$YES = 000010          1234#  7215  7221  7232
HELP  = 000000          1225#  1264  1353  1414  1723  3003  3008
HOE   = 100000 G        1556#
IBE   = 010000 G        1553#
IDU   = 000040 G        1546#
IER   = 020000 G        1554#
INIFLG  002474          1643#
INSTU   003616          2028   2039#
ISR   = 000100 G        1547#
ISRUN   030050          7233   7293#
IXE   = 004000 G        1552#
I$AU  = 000041          1234#  3239#  3243#
I$AUTO= 000041          1234#  3166#  3189#
I$CLN = 000041          1234#  3198#  3205#
I$DU  = 000041          1234#  3218#  3225#
I$HRD = 000041          7211#  7240#
I$INIT= 000041          1234#  3025#  3163#
I$MOD = 000040          1234#  1240#
I$MSG = 000041          1234#  2347#  2375#  2376#  2404#  2405#  2433#  2434#  2461#  2462#  2489#  2490#  2518#
                        2519#  2547#  2548#  2575#  2576#  2603#  2604#  2631#  2632#  2659#  2660#  2687#  2688#
                        2716#  2717#  2744#  2747#  2767#  2768#  2795#  2796#  2823#  2825#  2835#  2837#  2854#
                        2855#  2871#  2872#  2888#  2890#  2908#  2909#  2927#  2928#  2952#  2953#  2971#  2973#
                        2991#
I$PROT= 000040          1234#  1354#
I$PTAB= 000041          1234#
I$PWR = 000041          1234#
I$RPT = 000041          1234#  3001#  3012#
I$SEG = 000041          1234#  3264   3307  3367  3379#  3393  3398#  3417  3429#  3444  3449#  3470  3481#
                        3500   3505#  3506#  3523  3528#  3546  3607  3670  3761  3891  3982  3994#  4014
                        4019#  4020#  4038  4043#  4044#  4062  4067#  4085  4096#  4113  4118#  4119#  4136
```

```
                    4141#  4142#  4159   4164#  4182   4193#  4212   4217#  4218#  4237   4242#  4243#  4261
                    4266#  4284   4295#  4313   4318#  4319#  4337   4342#  4343#  4361   4366#  4384   4396#
                    4414   4419#  4420#  4438   4443#  4444#  4462   4467#  4485   4497#  4515   4520#  4521#
                    4539   4544#  4545#  4563   4568#  4586   4597#  4615   4620#  4621#  4639   4644#  4645#
                    4663   4668#  4686   4697#  4715   4720#  4721#  4739   4744#  4745#  4763   4768#  4786
                    4796#  4815   4820#  4821#  4841   4846#  4847#  4865   4870#  4888   4898#  4917   4922#
                    4923#  4943   4948#  4949#  4967   4972#  4990   5000#  5018   5023#  5024#  5044   5049#
                    5050#  5068   5073#  5091   5101#  5119   5124#  5125#  5145   5150#  5151#  5169   5174#
                    5193   5203#  5221   5226#  5227#  5247   5252#  5253#  5271   5276#  5294   5304#  5322
                    5327#  5328#  5348   5353#  5354#  5372   5377#  5393   5473   5564   5752   5871   5945
                    6016   6092   6202   6430   6485   6555   6748   6809   6855   6921   6972   7018   7084
I$SETU= 000041       1234#
I$SFT = 000041       7323#  7330#
I$SRV = 000041       1234#
I$SUB = 000041       1234#  3264   3307   3367   3417   3470   3546   3607   3670   3761   3891   3982   4085
                     4182   4284   4384   4485   4586   4686   4786   4888   4990   5091   5193   5294   5393
                     5473   5564   5752   5871   5945   6016   6092   6202   6430   6485   6555   6748   6809
                     6855   6921   6972   7018   7084
I$TST = 000041       1234#  3264#  3272   3289   3293#  3295#  3307#  3335   3356#  3358#  3367#  3370   3406#
                     3408#  3417#  3420   3458#  3460#  3470#  3473   3534#  3536#  3546#  3583   3596#  3598#
                     3607#  3645   3658#  3660#  3670#  3707   3740   3749#  3751#  3761#  3764   3833   3878#
                     3880#  3891#  3894   3922   3965#  3967#  3982#  3986   4068#  4070#  4085#  4088   4165#
                     4167#  4182#  4185   4267#  4269#  4284#  4287   4367#  4369#  4384#  4387   4468#  4470#
                     4485#  4488   4569#  4571#  4586#  4589   4669#  4671#  4686#  4689   4769#  4771#  4786#
                     4789   4871#  4873#  4888#  4891   4973#  4975#  4990#  4993   5074#  5076#  5091#  5094
                     5175#  5177#  5193#  5196   5277#  5279#  5294#  5297   5378#  5380#  5393#  5396   5449
                     5456#  5458#  5473#  5476   5516   5525#  5527#  5564#  5569   5661   5716#  5718#  5752#
                     5757   5849   5858#  5860#  5871#  5874   5895   5914   5931#  5933#  5945#  5948   5969
                     5987   6004#  6006#  6016#  6077#  6079#  6092#  6154   6187#  6189#  6202#  6319   6372
                     6398   6404   6416#  6418#  6430#  6472#  6474#  6485#  6489   6527   6533   6541#  6543#
                     6555#  6560   6616   6632   6646   6660   6676   6687   6699   6712   6733#  6735#  6748#
                     6769   6795#  6797#  6809#  6824   6841#  6843#  6855#  6859   6896   6904#  6906#  6921#
                     6925   6955#  6957#  6972#  7001#  7003#  7018#  7022   7068#  7070#  7084#  7195#  7197#
JSJMP = 000167        1234#  3005
KMACTV 002376         1608#
KMCSR  002516         1679#  3083*  3168   3265   3310   3373   3423   3476   3548   3609   3672   3767   3897
                      3990   4091   4188   4290   4390   4491   4592   4692   4792   4894   4996   5097   5199
                      5300   5399   5479   5566   5754   5878   5952   6018   6096   6128   6204   6432   6493
                      6564   6750   6811   6863   6929   6974   7026   7086
KMCSRH 002520         1680#  1850*  1851*  1863*  1865*  1866*  1873*  1877*  1879*  1880*  1883*  3085*  3086*
KMCTL  002522         1681#  3088*  3089*
KMNUM  002400         1609#
KMPO4  002524         1682#  3091*  3092*
KMPO6  002526         1683#  1864*  1874*  3094*  3095*
KMRLVL 002510         1676#  3099*  3100*
KMRVEC 002506         1675#  3097*
KMTLVL 002514         1678#  3105*  3106*
KMTVEC 002512         1677#  3102*  3103*
LNUNIT 027574         7263#
LOADER 026006         6566   6720#
LOCK   002340         1589#  3712*
LOE    = 040000 G     1555#
LOGDEV 002342         1594#  3066*  3072*  3073   3076   3181
LOKFLG 002476         1645#
LOOPBK 027771         7285#
LOT    = 000010 G     1544#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
LSACP   002110 G          1341#
LSAPT   002036 G          1299#
LSAU    012144 G          1326    3239#
LSAUT   002070 G          1325#
LSAUTO  012042 G          1342    3166#
LSCCP   002106 G          1339#
LSCLEA  012134 G          1340    3198#
LSCO    002032 G          1295#
LSDEPO  002011 G          1277#
LSDESC  002312 G          1332    1578#
LSDESP  002076 G          1331#
LSDEVP  002060 G          1317#
LSDISP  002132 G          1302    1369#
LSDLY   002116 G          1347#
LSDTP   002040 G          1301#
LSDTYP  002034 G          1297#
LSDU    012140 G          1328    3218#
LSDUT   002072 G          1327#
LSDVTY  002730 G          1318    1711#
LSEF    002052 G          1312#
LSENVI  002044 G          1305#
LSETP   002102 G          1335#
LSEXP1  002046 G          1307#
LSEXP4  002064 G          1321#
LSEXP5  002066 G          1323#
LSHARD  027344 G          1284    7211    7212#
LSHIME  002120 G          1349#
LSHPCP  002016 G          1283#
LSHPTP  002022 G          1287#
LSHW    002262 G          1288    1429    1430#
LSICP   002104 G          1337#
LSINIT  011340 G          1338    3025#
LSLADP  002026 G          1291#
LSLAST  030144 G          1292    7342#
LSLOAD  002100 G          1333#
LSLUN   002074 G          1329#
LSMREV  002050 G          1309#
LSNAME  002000 G          1266#
LSPRIO  002042 G          1303#
LSPROT  002122 G          1344    1354#
LSPRT   002112 G          1343#
LSREPP  002062 G          1319#
LSREV   002010 G          1275#
LSRPT   011332 G          3001#
LSSOFT  030140 G          7323    7324#
LSSPC   002056 G          1315#
LSSPCP  002020 G          1285#
LSSPTP  002024 G          1289#
LSSTA   002030 G          1293#
LSSW    002312 G          1456    1457#
LSTEST  002114 G          1345#
LSTIML  002014 G          1281#
LSUNIT  002012 G          1279#   3073
L10001  002310            1429    1447#
L10002  002312            1456    1462#
L10003  006156            2373#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

L10004	006304	2402#		
L10005	006432	2431#		
L10006	006554	2459#		
L10007	006676	2487#		
L10010	007024	2516#		
L10011	007152	2545#		
L10012	007274	2573#		
L10013	007416	2601#		
L10014	007540	2629#		
L10015	007662	2657#		
L10016	010004	2685#		
L10017	010132	2714#		
L10020	010254	2742#		
L10021	010346	2765#		
L10022	010470	2793#		
L10023	010612	2821#		
L10024	010634	2833#		
L10025	010700	2852#		
L10026	010742	2869#		
L10027	011004	2886#		
L10030	011052	2906#		
L10031	011122	2925#		
L10032	011212	2950#		
L10033	011262	2969#		
L10034	011330	2989#		
L10035	011336	3006	3010#	
L10036	012040	3161#		
L10037	012132	3187#		
L10040	012136	3203#		
L10041	012142	3223#		
L10042	012144	3241#		
L10043	012254	3273	3290	3293#
L10044	012420	3336	3356#	
L10045	012550	3371	3406#	
L10046	012710	3421	3458#	
L10047	013152	3474	3534#	
L10050	013354	3584	3596#	
L10051	013566	3646	3658#	
L10052	014110	3708	3741	3749#
L10053	014466	3765	3834	3878#
L10054	014734	3895	3923	3965#
L10055	015200	3987	4068#	
L10056	015430	4089	4165#	
L10057	015674	4186	4267#	
L10060	016140	4288	4367#	
L10061	016404	4388	4468#	
L10062	016650	4489	4569#	
L10063	017114	4590	4669#	
L10064	017360	4690	4769#	
L10065	017640	4790	4871#	
L10066	020120	4892	4973#	
L10067	020374	4994	5074#	
L10070	020650	5095	5175#	
L10071	021126	5197	5277#	
L10072	021402	5298	5378#	
L10073	021574	5397	5450	5456#

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
L10074  021742          5477    5517    5525#
L10075  022320          5570    5662    5716#
L10076  022556          5758    5850    5858#
L10077  022772          5875    5896    5915    5931#
L10100  023206          5949    5970    5988    6004#
L10101  023450          6077#
L10102  024042          6155    6187#
L10103  025114          6320    6373    6399    6405    6416#
L10104  025214          6472#
L10105  025362          6490    6528    6534    6541#
L10106  026040          6561    6617    6633    6647    6661    6677    6688    6700    6713    6733#
L10107  026200          6770    6795#
L10110  026306          6825    6841#
L10111  026444          6860    6897    6904#
L10112  026542          6926    6955#
L10113  026642          7001#
L10114  026766          7023    7068#
L10115  027340          7195#
L10116  027400          7211    7239#
L10117  030140          7323    7329#
MASKX   002362          1602#
MEMLIM  002374          1607#
MEMSET  003474          2007#   3995    4094    4191    4293    4393    4494    4594    4695    4794    4896    4998    5099
                        5201    5302
MEMSZ   002436          1624#   2002    2023    3145*   3153*   3402    3454    3508    3530    3592    3654    3710    3745
                        3764    3771    3818    5396    5476    5569    5757    5874    5948    6503    6529
MRO     002434          1623#   2054    3375*   3381    3401*   3402    3425*   3432    3453*   3454    3479*   3482    3484
                        3485    3492    3507*   3508    3510*   3511    3513    3515    3529*   3530    3552*   3563    3573
                        3587*   3613*   3614*   3625    3635    3647*   3649*   6310*
NEWST   011454          3044    3050    3064#   3074
NEXT    002336          1588#
ONE     002372          1606#
O$APTS= 000000          1234#   1293
O$AU  = 000001          1234#   1263#   1325
O$BGNR= 000000          1234#   1319
O$BGNS= 000000          1234#   1285
O$DU  = 000001          1234#   1263#   1327
O$ERRT= 000000          1234#   1335
O$GNSW= 000000          1234#   1289
O$POIN= 000001          1234#   1263#   1351
O$SETU= 000000          1234#   1279    7340
PATCH   002774          1854#
PNT   = 001000 G        1550#
PRI   = 002000 G        1551#
PRIRTY  027540          7258#
PRI00 = 000000 G        1539#
PRI01 = 000040 G        1538#
PRI02 = 000100 G        1537#
PRI03 = 000140 G        1536#
PRI04 = 000200 G        1535#
PRI05 = 000240 G        1534#
PRI06 = 000300 G        1533#
PRI07 = 000340 G        1532#   7089
PSTACK  002344          1595#   3030*   6138
QV.FLG  002477          1646#
RAMDAT  003330          1961#   4004    4028    4052    4103    4126    4149    4201    4226    4251    4303    4327    4351
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                    4404    4428    4452    4505    4529    4553    4605    4629    4653    4705    4729    4753    4805
                    4831    4855    4907    4933    4957    5008    5034    5058    5109    5135    5159    5211    5237
                    5261    5312    5338    5362
REGADR  002530      1687#
RETADR  002352      1598#
ROMMAP  012146      1996    3254#   6495    6509
RUN     002410      1613#
RUNB    002470      1637#   3157*   3846    5674    6141    7028    7174
RUNINH  002472      1638#   3108*   3157    3848    5676
SAVACT  002402      1610#
SAVE4   002464      1635#   3033*   3036    3184    3286    6766
SAVE6   002466      1636#   3034*   3037    3185    3287    6767
SAVNUM  002404      1611#
SAVPC   002366      1604#
SAVSP   002364      1603#
SETBR0  003220      1902#   4397    4421    4445
SETBR1  003230      1909#   4498    4522    4546
SETBR4  003240      1917#   4598    4622    4646
SETBR7  003250      1925#   4698    4722    4746
SETC    003260      1933#   4194    4219    4244    4797    5971
SETZ    003312      1950#   4296    4320    4344    4899    5897
SFPTBL  002312 G    1458#
SSTACK  002730      1691#   3028
STAT    002356      1600#
STAT1   002500      1668#   3109*   3111*   3115*   3126*   3130*   3133*
STAT2   002502      1669#   3136*   3138*
STAT3   002504      1670#
STM     005627      2260#   2368    2397    2426    2454    2482    2511    2540    2568    2596    2624    2652    2680
                    2709    2737    2760    2788    2816    2827    2846    2863    2880    2900    2919    2944    2963
                    2983
STRTSW  002354      1599#
SUBRPC  002346      1596#
SVCGBL= 000000      1234#   1240    1247#   1266    1267    1275    1276    1277    1278    1279    1280    1281    1282
                    1283    1284    1285    1286    1287    1288    1289    1290    1291    1292    1293    1294    1295
                    1296    1297    1298    1299    1300    1301    1302    1303    1304    1305    1306    1307    1308
                    1309    1310    1312    1313    1315    1316    1317    1318    1319    1320    1321    1322    1323
                    1324    1325    1326    1327    1328    1329    1330    1331    1332    1333    1334    1335    1336
                    1337    1338    1339    1340    1341    1342    1343    1344    1345    1346    1347    1348    1349
                    1350    1354    1355    1369    1370    1430    1431    1432    1457    1458    1459    1578    1579
                    1711    1712    2347    2348    2376    2377    2405    2406    2434    2435    2462    2463    2490
                    2491    2519    2520    2548    2549    2576    2577    2604    2605    2632    2633    2660    2661
                    2688    2689    2717    2718    2747    2748    2768    2769    2796    2797    2825    2826    2837
                    2838    2855    2856    2872    2873    2890    2891    2909    2910    2928    2929    2953    2954
                    2973    2974    3001    3002    3025    3026    3166    3167    3198    3199    3218    3219    3239
                    3240    7212    7213    7324    7325    7342#   7343
SVCINS= 000000      1234#   1244#   1267    1268    1269    1270    1271    1272    1273    1274    1275    1276    1277
                    1278    1279    1280    1281    1282    1283    1284    1285    1286    1287    1288    1289    1290
                    1291    1292    1293    1294    1295    1296    1297    1298    1299    1300    1301    1302    1303
                    1304    1305    1306    1307    1308    1309    1310    1311    1312    1313    1314    1315    1316
                    1317    1318    1319    1320    1321    1322    1323    1324    1325    1326    1327    1328    1329
                    1330    1331    1332    1333    1334    1335    1336    1337    1338    1339    1340    1341    1342
                    1343    1344    1345    1346    1347    1348    1349    1350    1351    1368    1369    1370    1371
                    1372    1373    1374    1375    1376    1377    1378    1379    1380    1381    1382    1383    1384
                    1385    1386    1387    1388    1389    1390    1391    1392    1393    1394    1395    1396    1397
                    1398    1399    1400    1401    1402    1403    1404    1405    1406    1407    1408    1409    1410
                    1411    1412    1413    1429    1430    1456    1457    1579    1583    1584    1712    1716    1717
```

2348 2349 2350 2351 2352 2353 2354 2355 2356 2357 2358 2359 2360
2361 2362 2363 2364 2365 2366 2367 2368 2369 2370 2371 2372 2373
2374 2375 2377 2378 2379 2380 2381 2382 2383 2384 2385 2386 2387
2388 2389 2390 2391 2392 2393 2394 2395 2396 2397 2398 2399 2400
2401 2402 2403 2404 2406 2407 2408 2409 2410 2411 2412 2413 2414
2415 2416 2417 2418 2419 2420 2421 2422 2423 2424 2425 2426 2427
2428 2429 2430 2431 2432 2433 2435 2436 2437 2438 2439 2440 2441
2442 2443 2444 2445 2446 2447 2448 2449 2450 2451 2452 2453 2454
2455 2456 2457 2458 2459 2460 2461 2463 2464 2465 2466 2467 2468
2469 2470 2471 2472 2473 2474 2475 2476 2477 2478 2479 2480 2481
2482 2483 2484 2485 2486 2487 2488 2489 2491 2492 2493 2494 2495
2496 2497 2498 2499 2500 2501 2502 2503 2504 2505 2506 2507 2508
2509 2510 2511 2512 2513 2514 2515 2516 2517 2518 2520 2521 2522
2523 2524 2525 2526 2527 2528 2529 2530 2531 2532 2533 2534 2535
2536 2537 2538 2539 2540 2541 2542 2543 2544 2545 2546 2547 2549
2550 2551 2552 2553 2554 2555 2556 2557 2558 2559 2560 2561 2562
2563 2564 2565 2566 2567 2568 2569 2570 2571 2572 2573 2574 2575
2577 2578 2579 2580 2581 2582 2583 2584 2585 2586 2587 2589 2589
2590 2591 2592 2593 2594 2595 2596 2597 2598 2599 2600 2601 2602
2603 2605 2606 2607 2608 2609 2610 2611 2612 2613 2614 2615 2616
2617 2618 2619 2620 2621 2622 2623 2624 2625 2626 2627 2628 2629
2630 2631 2633 2634 2635 2636 2637 2638 2639 2640 2641 2642 2643
2644 2645 2646 2647 2648 2649 2650 2651 2652 2653 2654 2655 2656
2657 2658 2659 2661 2662 2663 2664 2665 2666 2667 2668 2669 2670
2671 2672 2673 2674 2675 2676 2677 2678 2679 2680 2681 2682 2683
2684 2685 2686 2687 2689 2690 2691 2692 2693 2694 2695 2696 2697
2698 2699 2700 2701 2702 2703 2704 2705 2706 2707 2708 2709 2710
2711 2712 2713 2714 2715 2716 2718 2719 2720 2721 2722 2723 2724
2725 2726 2727 2728 2729 2730 2731 2732 2733 2734 2735 2736 2737
2738 2739 2740 2741 2742 2743 2744 2748 2749 2750 2751 2752 2753
2754 2755 2756 2757 2758 2759 2760 2761 2762 2763 2764 2765 2766
2767 2769 2770 2771 2772 2773 2774 2775 2776 2777 2778 2779 2780
2781 2782 2783 2784 2785 2786 2787 2788 2789 2790 2791 2792 2793
2794 2795 2797 2798 2799 2800 2801 2802 2803 2804 2805 2806 2807
2808 2809 2810 2811 2812 2813 2814 2815 2816 2817 2818 2819 2820
2821 2822 2823 2827 2828 2829 2830 2831 2832 2834 2835 2839 2840
2841 2842 2843 2844 2845 2846 2847 2848 2849 2850 2851 2853 2854
2857 2858 2859 2860 2861 2862 2863 2864 2865 2866 2867 2868 2870
2871 2874 2875 2876 2877 2878 2879 2880 2881 2882 2883 2884 2885
2887 2888 2892 2893 2894 2895 2896 2897 2898 2899 2900 2901 2902
2903 2904 2905 2907 2908 2911 2912 2913 2914 2915 2916 2917 2918
2919 2920 2921 2922 2923 2924 2926 2927 2930 2931 2932 2933 2934
2935 2936 2937 2938 2939 2940 2941 2942 2943 2944 2945 2946 2947
2948 2949 2951 2952 2955 2956 2957 2958 2959 2960 2961 2962 2963
2964 2965 2966 2967 2968 2970 2971 2975 2976 2977 2978 2979 2980
2981 2982 2983 2984 2985 2986 2987 2988 2990 2991 3005 3006 3007
3011 3012 3041 3042 3043 3044 3045 3047 3048 3049 3050 3051 3053
3054 3055 3056 3057 3059 3060 3061 3062 3063 3076 3077 3078 3079
3080 3081 3162 3163 3181 3182 3183 3188 3189 3200 3201 3204 3205
3221 3222 3224 3225 3242 3243 3272 3273 3274 3281 3282 3283 3284
3285 3289 3290 3291 3294 3295 3329 3330 3331 3332 3333 3335 3336
3337 3349 3350 3351 3352 3353 3357 3358 3370 3371 3372 3379 3380
3388 3389 3390 3391 3392 3393 3394 3395 3397 3398 3407 3408 3420
3421 3422 3429 3430 3439 3440 3441 3442 3443 3444 3445 3446 3448
3449 3459 3460 3473 3474 3475 3481 3482 3495 3496 3497 3498 3499
3500 3501 3502 3504 3505 3506 3507 3518 3519 3520 3521 3522 3523

3524	3525	3527	3528	3535	3536	3578	3579	3580	3581	3582	3583	3584
3585	3590	3591	3597	3598	3640	3641	3642	3643	3644	3645	3646	3647
3652	3653	3659	3660	3702	3703	3704	3705	3706	3707	3708	3709	3735
3736	3737	3738	3739	3740	3741	3742	3743	3744	3750	3751	3764	3765
3766	3800	3801	3802	3803	3804	3825	3826	3827	3828	3829	3833	3834
3835	3836	3837	3872	3873	3874	3875	3876	3879	3880	3894	3895	3896
3917	3918	3919	3920	3921	3922	3923	3924	3940	3941	3942	3943	3944
3959	3960	3961	3962	3963	3966	3967	3986	3987	3988	3994	3995	4009
4010	4011	4012	4013	4014	4015	4016	4018	4019	4020	4021	4033	4034
4035	4036	4037	4038	4039	4040	4042	4043	4044	4045	4057	4058	4059
4060	4061	4062	4063	4064	4066	4067	4069	4070	4088	4089	4090	4096
4097	4108	4109	4110	4111	4112	4113	4114	4115	4117	4118	4119	4120
4131	4132	4133	4134	4135	4136	4137	4138	4140	4141	4142	4143	4154
4155	4156	4157	4158	4159	4160	4161	4163	4164	4166	4167	4185	4186
4187	4193	4194	4206	4207	4208	4209	4210	4212	4213	4214	4216	4217
4218	4219	4231	4232	4233	4234	4235	4237	4238	4239	4241	4242	4243
4244	4256	4257	4258	4259	4260	4261	4262	4263	4265	4266	4268	4269
4287	4288	4289	4295	4296	4308	4309	4310	4311	4312	4313	4314	4315
4317	4318	4319	4320	4332	4333	4334	4335	4336	4337	4338	4339	4341
4342	4343	4344	4356	4357	4358	4359	4360	4361	4362	4363	4365	4366
4368	4369	4387	4388	4389	4396	4397	4409	4410	4411	4412	4413	4414
4415	4416	4418	4419	4420	4421	4433	4434	4435	4436	4437	4438	4439
4440	4442	4443	4444	4445	4457	4458	4459	4460	4461	4462	4463	4464
4466	4467	4469	4470	4488	4489	4490	4497	4498	4510	4511	4512	4513
4514	4515	4516	4517	4519	4520	4521	4522	4534	4535	4536	4537	4538
4539	4540	4541	4543	4544	4545	4546	4558	4559	4560	4561	4562	4563
4564	4565	4567	4568	4570	4571	4589	4590	4591	4597	4598	4610	4611
4612	4613	4614	4615	4616	4617	4619	4620	4621	4622	4634	4635	4636
4637	4638	4639	4640	4641	4643	4644	4645	4646	4658	4659	4660	4661
4662	4663	4664	4665	4667	4668	4670	4671	4689	4690	4691	4697	4698
4710	4711	4712	4713	4714	4715	4716	4717	4719	4720	4721	4722	4734
4735	4736	4737	4738	4739	4740	4741	4743	4744	4745	4746	4758	4759
4760	4761	4762	4763	4764	4765	4767	4768	4770	4771	4789	4790	4791
4796	4797	4810	4811	4812	4813	4814	4815	4816	4817	4819	4820	4821
4822	4836	4837	4838	4839	4840	4841	4842	4843	4845	4846	4847	4848
4860	4861	4862	4863	4864	4865	4866	4867	4869	4870	4872	4873	4891
4892	4893	4898	4899	4912	4913	4914	4915	4916	4917	4918	4919	4921
4922	4923	4924	4938	4939	4940	4941	4942	4943	4944	4945	4947	4948
4949	4950	4962	4963	4964	4965	4966	4967	4968	4969	4971	4972	4974
4975	4993	4994	4995	5000	5001	5013	5014	5015	5016	5017	5018	5019
5020	5022	5023	5024	5025	5039	5040	5041	5042	5043	5044	5045	5046
5048	5049	5050	5051	5063	5064	5065	5066	5067	5068	5069	5070	5072
5073	5075	5076	5094	5095	5096	5101	5102	5114	5115	5116	5117	5118
5119	5120	5121	5123	5124	5125	5126	5140	5141	5142	5143	5144	5145
5146	5147	5149	5150	5151	5152	5164	5165	5166	5167	5168	5169	5170
5171	5173	5174	5176	5177	5196	5197	5198	5203	5204	5216	5217	5218
5219	5220	5221	5222	5223	5225	5226	5227	5228	5242	5243	5244	5245
5246	5247	5248	5249	5251	5252	5253	5254	5266	5267	5268	5269	5270
5271	5272	5273	5275	5276	5278	5279	5297	5298	5299	5304	5305	5317
5318	5319	5320	5321	5322	5323	5324	5326	5327	5328	5329	5343	5344
5345	5346	5347	5348	5349	5350	5352	5353	5354	5355	5367	5368	5369
5370	5371	5372	5373	5374	5376	5377	5379	5380	5396	5397	5398	5440
5441	5442	5443	5444	5449	5450	5451	5457	5458	5476	5477	5478	5508
5509	5510	5511	5512	5516	5517	5518	5526	5527	5569	5570	5571	5626
5627	5628	5629	5630	5661	5662	5663	5709	5710	5711	5712	5713	5717
5718	5757	5758	5759	5814	5815	5816	5817	5818	5849	5850	5851	5859

```
                    5860  5874  5875  5876  5889  5890  5891  5892  5893  5895  5896  5897  5908
                    5909  5910  5911  5912  5914  5915  5916  5925  5926  5927  5928  5929  5932
                    5933  5948  5949  5950  5963  5964  5965  5966  5967  5969  5970  5971  5981
                    5982  5983  5984  5985  5987  5988  5989  5998  5999  6000  6001  6002  6005
                    6006  6035  6036  6037  6038  6039  6052  6053  6054  6055  6056  6068  6069
                    6070  6071  6072  6078  6079  6094  6095  6111  6112  6113  6114  6115  6133
                    6134  6135  6136  6137  6146  6147  6148  6149  6150  6154  6155  6156  6163
                    6164  6165  6166  6167  6176  6177  6178  6179  6180  6188  6189  6314  6315
                    6316  6317  6318  6319  6320  6321  6345  6346  6347  6348  6349  6367  6368
                    6369  6370  6371  6372  6373  6374  6393  6394  6395  6396  6397  6398  6399
                    6400  6404  6405  6406  6417  6418  6465  6466  6467  6468  6469  6473  6474
                    6489  6490  6491  6520  6521  6522  6523  6524  6527  6528  6529  6533  6534
                    6535  6542  6543  6560  6561  6562  6610  6611  6612  6613  6614  6616  6617
                    6618  6625  6626  6627  6628  6629  6632  6633  6634  6640  6641  6642  6643
                    6644  6646  6647  6648  6654  6655  6656  6657  6658  6660  6661  6662  6670
                    6671  6672  6673  6674  6676  6677  6678  6682  6683  6684  6685  6686  6687
                    6688  6689  6694  6695  6696  6697  6698  6699  6700  6701  6706  6707  6708
                    6709  6710  6712  6713  6714  6734  6735  6769  6770  6771  6788  6789  6790
                    6791  6792  6796  6797  6819  6820  6821  6822  6823  6824  6825  6826  6833
                    6834  6835  6836  6837  6842  6843  6859  6860  6861  6889  6890  6891  6892
                    6893  6896  6897  6898  6905  6906  6925  6926  6927  6947  6948  6949  6950
                    6951  6956  6957  6994  6995  6996  6997  6998  7002  7003  7022  7023  7024
                    7060  7061  7062  7063  7064  7069  7070  7089  7090  7091  7113  7114  7115
                    7116  7117  7118  7119  7121  7122  7139  7140  7141  7142  7143  7145  7146
                    7162  7163  7164  7165  7166  7169  7170  7172  7173  7185  7186  7187  7188
                    7189  7196  7197  7211  7212  7215  7216  7217  7218  7219  7220  7221  7222
                    7223  7224  7225  7232  7233  7234  7235  7236  7237  7238  7239  7323  7324
                    7328  7329  7339  7340  7341  7342
SVCSUB= 000000       1234#  1246#
SVCTAG= 000000       1234#  1248#  1447  1448  1462  1463  2373  2374  2402  2403  2431  2432  2459
                    2460  2487  2488  2516  2517  2545  2546  2573  2574  2601  2602  2629  2630
                    2657  2658  2685  2686  2714  2715  2742  2743  2765  2766  2793  2794  2821
                    2822  2833  2834  2852  2853  2869  2870  2886  2887  2906  2907  2925  2926
                    2950  2951  2969  2970  2989  2990  3010  3011  3161  3162  3187  3188  3203
                    3204  3223  3224  3241  3242  3293  3294  3356  3357  3396  3397  3406  3407
                    3447  3448  3458  3459  3503  3504  3526  3527  3534  3535  3596  3597  3658
                    3659  3749  3750  3878  3879  3965  3966  4017  4018  4041  4042  4065  4066
                    4068  4069  4116  4117  4139  4140  4162  4163  4165  4166  4215  4216  4240
                    4241  4264  4265  4267  4268  4316  4317  4340  4341  4364  4365  4367  4368
                    4417  4418  4441  4442  4465  4466  4468  4469  4518  4519  4542  4543  4566
                    4567  4569  4570  4618  4619  4642  4643  4666  4667  4669  4670  4718  4719
                    4742  4743  4766  4767  4769  4770  4818  4819  4844  4845  4868  4869  4871
                    4872  4920  4921  4946  4947  4970  4971  4973  4974  5021  5022  5047  5048
                    5071  5072  5074  5075  5122  5123  5148  5149  5172  5173  5175  5176  5224
                    5225  5250  5251  5274  5275  5277  5278  5325  5326  5351  5352  5375  5376
                    5378  5379  5456  5457  5525  5526  5716  5717  5858  5859  5931  5932  6004
                    6005  6077  6078  6187  6188  6416  6417  6472  6473  6541  6542  6733  6734
                    6795  6796  6841  6842  6904  6905  6955  6956  7001  7002  7068  7069  7195
                    7196  7239  7240  7329  7330
SVCTST= 000000       1234#  1245#  3264  3265  3307  3308  3367  3368  3417  3418  3470  3471  3546
                    3547  3607  3608  3670  3671  3761  3762  3891  3892  3982  3983  4085  4086
                    4182  4183  4284  4285  4384  4385  4485  4486  4586  4587  4686  4687  4786
                    4787  4888  4889  4990  4991  5091  5092  5193  5194  5294  5295  5393  5394
                    5473  5474  5564  5565  5752  5753  5871  5872  5945  5946  6016  6017  6092
                    6093  6202  6203  6430  6431  6485  6486  6555  6556  6748  6749  6809  6810
                    6855  6856  6921  6922  6972  6973  7018  7019  7084  7085
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
SV05    003632  2049#  3281   3329   3349   3388   3439   3495   3518   3578   3640   3702   3735   3800
                3825   3872   3917   3940   3959   4009   4033   4057   4108   4131   4154   4206   4231
                4256   4308   4332   4356   4409   4433   4457   4510   4534   4558   4610   4634   4658
                4710   4734   4758   4810   4836   4860   4912   4938   4962   5013   5039   5063   5114
                5140   5164   5216   5242   5266   5317   5343   5367   5440   5508   5626   5709   5814
                5889   5908   5925   5963   5981   5998   6111   6133   6146   6163   6176   6314   6345
                6367   6393   6465   6520   6610   6625   6640   6654   6670   6682   6694   6706   6788
                6819   6833   6889   6947   6994   7060   7115   7139   7162   7185
SWPAC1  027665  7273#
SWPAC2  027726  7279#
S$LSYM= 010000  1234#  1448#  1463#  2374#  2403#  2432#  2460#  2488#  2517#  2546#  2574#  2602#  2630#
                2658#  2686#  2715#  2743#  2766#  2794#  2822#  2834#  2853#  2870#  2887#  2907#  2926#
                2951#  2970#  2990#  3011#  3162#  3188#  3204#  3224#  3242#  3294#  3357#  3379#  3407#
                3429#  3459#  3481#  3506#  3535#  3597#  3659#  3750#  3879#  3966#  3994#  4020#  4044#
                4069#  4096#  4119#  4142#  4166#  4193#  4218#  4243#  4268#  4295#  4319#  4343#  4368#
                4396#  4420#  4444#  4469#  4497#  4521#  4545#  4570#  4597#  4621#  4645#  4670#  4697#
                4721#  4745#  4770#  4796#  4821#  4847#  4872#  4898#  4923#  4949#  4974#  5000#  5024#
                5050#  5075#  5101#  5125#  5151#  5176#  5203#  5227#  5253#  5278#  5304#  5328#  5354#
                5379#  5457#  5526#  5717#  5859#  5932#  6005#  6078#  6188#  6417#  6473#  6542#  6734#
                6796#  6842#  6905#  6956#  7002#  7069#  7196#  7240#  7330#
TEMP    002440  1625#  6020*  6045*  6064*  6099*  6107*
TFM1    003666  2069#  2363   2392   2421
TFM2    003712  2073#  2449   2477   2563   2647   2783   2811
TFM3    003730  2076#  2506   2535
TFM36   004017  2087#
TFM4    003755  2080#  2591   2619   2675   2732
TFM40   004263  2117#  2840
TFM41   004105  2097#  2857
TFM42   004174  2107#  2874
TFM43   004344  2126#  2894
TFM44   004427  2135#  2913
TFM45   004466  2141#  2932
TFM45A  004524  2146#  2938
TFM46   004647  2161#  2957
TFM47   004733  2170#  2977
TFM5    003773  2083#  2704
TMMC    005010  2178#  2912   2931
TYPE    002432  1622#  3146*  3151*  3156*  3370   3420   3473   3894   6292
T$ARGC= 000001  1267#  1268#  1269#  1270#  1271#  1272#  2348#  2353   2354#  2359   2360#  2367   2368#
                2372   2377#  2382   2383#  2388   2389#  2396   2397#  2401   2406#  2411   2412#  2417
                2418#  2425   2426#  2430   2435#  2440   2441#  2446   2447#  2453   2454#  2458   2463#
                2468   2469#  2474   2475#  2481   2482#  2486   2491#  2496   2497#  2502   2503#  2510
                2511#  2515   2520#  2525   2526#  2531   2532#  2539   2540#  2544   2549#  2554   2555#
                2560   2561#  2567   2568#  2572   2577#  2582   2583#  2588   2589#  2595   2596#  2600
                2605#  2610   2611#  2616   2617#  2623   2624#  2628   2633#  2638   2639#  2644   2645#
                2651   2652#  2656   2661#  2666   2667#  2672   2673#  2679   2680#  2684   2689#  2694
                2695#  2700   2701#  2708   2709#  2713   2718#  2723   2724#  2729   2730#  2736   2737#
                2741   2748#  2753   2754#  2759   2760#  2764   2769#  2774   2775#  2780   2781#  2787
                2788#  2792   2797#  2802   2803#  2808   2809#  2815   2816#  2820   2827#  2831   2839#
                2844   2846#  2850   2857#  2861   2863#  2867   2874#  2878   2880#  2884   2892#  2898
                2900#  2904   2911#  2917   2919#  2923   2930#  2936   2938#  2942   2944#  2948   2955#
                2961   2963#  2967   2975#  2981   2983#  2987
T$CODE= 012032  7215#  7221#  7232#
T$ERRN= 000056  1234#  3282#  3330#  3350#  3389#  3440#  3496#  3519#  3579#  3641#  3703#  3736#  3801#
                3826#  3873#  3918#  3941#  3960#  4010#  4034#  4058#  4109#  4132#  4155#  4207#  4232#
                4257#  4309#  4333#  4357#  4410#  4434#  4458#  4511#  4535#  4559#  4611#  4635#  4659#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                    4711#  4735#  4759#  4811#  4837#  4861#  4913#  4939#  4963#  5014#  5040#  5064#  5115#
                    5141#  5165#  5217#  5243#  5267#  5318#  5344#  5368#  5441#  5509#  5627#  5710#  5815#
                    5890#  5909#  5926#  5964#  5982#  5999#  6036#  6053#  6069#  6112#  6134#  6147#  6164#
                    6177#  6315#  6346#  6368#  6394#  6466#  6521#  6611#  6626#  6641#  6655#  6671#  6683#
                    6695#  6707#  6789#  6820#  6834#  6890#  6948#  6995#  7061#  7116#  7140#  7163#  7186#
TSEXCP= 000000       7215#  7220   7221#  7225   7232#  7237
TSFLAG= 000040       3005#  3007   3272#  3289#  3335#  3370#  3393#  3420#  3444#  3473#  3500#  3523#  3583#
                    3645#  3707#  3740#  3764#  3833#  3894#  3922#  3986#  4014#  4038#  4062#  4088#  4113#
                    4136#  4159#  4185#  4212#  4237#  4261#  4287#  4313#  4337#  4361#  4387#  4414#  4438#
                    4462#  4488#  4515#  4539#  4563#  4589#  4615#  4639#  4663#  4689#  4715#  4739#  4763#
                    4789#  4815#  4841#  4865#  4891#  4917#  4943#  4967#  4993#  5018#  5044#  5068#  5094#
                    5119#  5145#  5169#  5196#  5221#  5247#  5271#  5297#  5322#  5348#  5372#  5396#  5449#
                    5476#  5516#  5569#  5661#  5757#  5849#  5874#  5895#  5914#  5948#  5969#  5987#  6154#
                    6319#  6372#  6398#  6404#  6489#  6527#  6533#  6560#  6616#  6632#  6646#  6660#  6676#
                    6687#  6699#  6712#  6769#  6824#  6859#  6896#  6925#  7022#
TSGMAN= 000000       1234#
TSHILI= 000001       7215#  7219   7221#  7224   7232#  7236
TSLAST= 000001       1234#  7340#
TSLOLI= 000000       7215#  7218   7221#  7223   7232#  7235
TSLSYM= 010000       1234#  1448   1463   2374   2403   2432   2460   2488   2517   2546   2574   2602   2630
                    2658   2686   2715   2743   2766   2794   2822   2834   2853   2870   2887   2907   2926
                    2951   2970   2990   3011   3162   3188   3204   3224   3242   3294   3357   3407   3459
                    3535   3597   3659   3750   3879   3966   4069   4166   4268   4368   4469   4570   4670
                    4770   4872   4974   5075   5176   5278   5379   5457   5526   5717   5859   5932   6005
                    6078   6188   6417   6473   6542   6734   6796   6842   6905   6956   7002   7069   7196
                    7240   7330
TSLTNO= 000053       7343#
TSNEST= 000000       1234#  1240#  1354#  1359#  1429#  1447#  1456#  1462#  2347#  2373#  2376#  2402#  2405#
                    2431#  2434#  2459#  2462#  2487#  2490#  2516#  2519#  2545#  2548#  2573#  2576#  2601#
                    2604#  2629#  2632#  2657#  2660#  2685#  2688#  2714#  2717#  2742#  2747#  2765#  2768#
                    2793#  2796#  2821#  2825#  2833#  2837#  2852#  2855#  2869#  2872#  2886#  2890#  2906#
                    2909#  2925#  2928#  2950#  2953#  2969#  2973#  2989#  3001#  3010#  3025#  3161#  3166#
                    3187#  3198#  3203#  3218#  3223#  3239#  3241#  3265#  3293#  3308#  3356#  3368#  3379#
                    3396#  3406#  3418#  3429#  3447#  3458#  3471#  3481#  3503#  3506#  3526#  3534#  3547#
                    3596#  3608#  3658#  3671#  3749#  3762#  3878#  3892#  3965#  3983#  3994#  4017#  4020#
                    4041#  4044#  4065#  4068#  4086#  4096#  4116#  4119#  4139#  4142#  4162#  4165#  4183#
                    4193#  4215#  4218#  4240#  4243#  4264#  4267#  4285#  4295#  4316#  4319#  4340#  4343#
                    4364#  4367#  4385#  4396#  44174   4420#  4441#  4444#  4465#  4468#  4486#  4497#  4518#
                    4521#  4542#  4545#  4566#  4569#  4587#  4597#  4618#  4621#  4642#  4645#  4666#  4669#
                    4687#  4697#  4718#  4721#  4742#  4745#  4766#  4769#  4787#  4796#  4818#  4821#  4844#
                    4847#  4868#  4871#  4889#  4898#  4920#  4923#  4946#  4949#  4970#  4973#  4991#  5000#
                    5021#  5024#  5047#  5050#  5071#  5074#  5092#  5101#  5122#  5125#  5148#  5151#  5172#
                    5175#  5194#  5203#  5224#  5227#  5250#  5253#  5274#  5277#  5295#  5304#  5325#  5328#
                    5351#  5354#  5375#  5378#  5394#  5456#  5474#  5525#  5565#  5716#  5753#  5858#  5872#
                    5931#  5946#  6004#  6017#  6077#  6093#  6187#  6203#  6416#  6431#  6472#  6486#  6541#
                    6556#  6733#  6749#  6795#  6810#  6841#  6856#  6904#  6922#  6955#  6973#  7001#  7019#
                    7068#  7085#  7195#  7211#  7238#  7323#  7328#
T$NS0 = 000000       1240#
T$NS1 = 000005       1354#  1359   1429#  1447   1456#  1462   2347#  2373   2376#  2402   2405#  2431   2434#
                    2459   2462#  2487   2490#  2516   2519#  2545   2548#  2573   2576#  2601   2604#  2629
                    2632#  2657   2660#  2685   2688#  2714   2717#  2742   2747#  2765   2768#  2793   2796#
                    2821   2825#  2833   2837#  2852   2855#  2869   2872#  2886   2890#  2906   2909#  2925
                    2928#  2950   2953#  2969   2973#  2989   3001#  3010   3025#  3161   3166#  3187   3198#
                    3203   3218#  3223   3239#  3241   3265#  3293   3308#  3356   3368#  3406   3418#  3458
                    3471#  3534   3547#  3596   3608#  3658   3671#  3749   3762#  3878   3892#  3965   3983#
                    4068   4086#  4165   4183#  4267   4285#  4367   4385#  4468   4486#  4569   4587#  4669
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                    4687#  4769   4787#  4871   4889#  4973   4991#  5074   5092#  5175   5194#  5277   5295#
                    5378   5394#  5456   5474#  5525   5565#  5716   5753#  5858   5872#  5931   5946#  6004
                    6017#  6077   6093#  6187   6203#  6416   6431#  6472   6486#  6541   6556#  6733   6749#
                    6795   6810#  6841   6856#  6904   6922#  6955   6973#  7001   7019#  7068   7085#  7195
                    7211#  7238   7323#  7328
T$NS2 = 000003      3379#  3396   3429#  3447   3481#  3503   3506#  3526   3994#  4017   4020#  4041   4044#
                    4065   4096#  4116   4119#  4139   4142#  4162   4193#  4215   4218#  4240   4243#  4264
                    4295#  4316   4319#  4340   4343#  4364   4396#  4417   4420#  4441   4444#  4465   4497#
                    4518   4521#  4542   4545#  4566   4597#  4618   4621#  4642   4645#  4666   4697#  4718
                    4721#  4742   4745#  4766   4796#  4818   4821#  4844   4847#  4868   4898#  4920   4923#
                    4946   4949#  4970   5000#  5021   5024#  5047   5050#  5071   5101#  5122   5125#  5148
                    5151#  5172   5203#  5224   5227#  5250   5253#  5274   5304#  5325   5328#  5351   5354#
                    5375
T$PTNU= 000000      1234#
T$SAVL= 177777      1234#
T$SEGL= 177777      1234#  3379#  3394   3396#  3398   3429#  3445   3447#  3449   3481#  3501   3503#  3505
                    3506#  3524   3526#  3528   3994#  4015   4017#  4019   4020#  4039   4041#  4043   4044#
                    4063   4065#  4067   4096#  4114   4116#  4118   4119#  4137   4139#  4141   4142#  4160
                    4162#  4164   4193#  4213   4215#  4217   4218#  4238   4240#  4242   4243#  4262   4264#
                    4266   4295#  4314   4316#  4318   4319#  4338   4340#  4342   4343#  4362   4364#  4366
                    4396#  4415   4417#  4419   4420#  4439   4441#  4443   4444#  4463   4465#  4467   4497#
                    4516   4518#  4520   4521#  4540   4542#  4544   4545#  4564   4566#  4568   4597#  4616
                    4618#  4620   4621#  4640   4642#  4644   4645#  4664   4666#  4668   4697#  4716   4718#
                    4720   4721#  4740   4742#  4744   4745#  4764   4766#  4768   4796#  4816   4818#  4820
                    4821#  4842   4844#  4846   4847#  4866   4868#  4870   4898#  4918   4920#  4922   4923#
                    4944   4946#  4948   4949#  4968   4970#  4972   5000#  5019   5021#  5023   5024#  5045
                    5047#  5049   5050#  5069   5071#  5073   5101#  5120   5122#  5124   5125#  5146   5148#
                    5150   5151#  5170   5172#  5174   5203#  5222   5224#  5226   5227#  5248   5250#  5252
                    5253#  5272   5274#  5276   5304#  5323   5325#  5327   5328#  5349   5351#  5353   5354#
                    5373   5375#  5377
T$SEK0= 010002      3379#  3394   3396   3429#  3445   3447   3481#  3501   3503   3506#  3524   3526   3994#
                    4015   4017   4020#  4039   4041   4044#  4063   4065   4096#  4114   4116   4119#  4137
                    4139   4142#  4160   4162   4193#  4213   4215   4218#  4238   4240   4243#  4262   4264
                    4295#  4314   4316   4319#  4338   4340   4343#  4362   4364   4396#  4415   4417   4420#
                    4439   4441   4444#  4463   4465   4497#  4516   4518   4521#  4540   4542   4545#  4564
                    4566   4597#  4616   4618   4621#  4640   4642   4645#  4664   4666   4697#  4716   4718
                    4721#  4740   4742   4745#  4764   4766   4796#  4816   4818   4821#  4842   4844   4847#
                    4866   4868   4898#  4918   4920   4923#  4944   4946   4949#  4968   4970   5000#  5019
                    5021   5024#  5045   5047   5050#  5069   5071   5101#  5120   5122   5125#  5146   5148
                    5151#  5170   5172   5203#  5222   5224   5227#  5248   5250   5253#  5272   5274   5304#
                    5323   5325   5328#  5349   5351   5354#  5373   5375
T$SUBN= 000000      1234#  3264#  3307#  3367#  3417#  3470#  3546#  3607#  3670#  3761#  3891#  3982#  4085#
                    4182#  4284#  4384#  4485#  4586#  4686#  4786#  4888#  4990#  5091#  5193#  5294#  5393#
                    5473#  5564#  5752#  5871#  5945#  6016#  6092#  6202#  6430#  6485#  6555#  6748#  6809#
                    6855#  6921#  6972#  7018#  7084#
T$TAGL= 177777      1234#
T$TAGN= 010120      1234#  1354#  1429#  1456#  2347#  2376#  2405#  2434#  2462#  2490#  2519#  2548#  2576#
                    2604#  2632#  2660#  2688#  2717#  2747#  2768#  2796#  2825#  2837#  2855#  2872#  2890#
                    2909#  2928#  2953#  2973#  3001#  3025#  3166#  3198#  3218#  3239#  3265#  3308#  3368#
                    3418#  3471#  3547#  3608#  3671#  3762#  3892#  3983#  4086#  4183#  4285#  4385#  4486#
                    4587#  4687#  4787#  4889#  4991#  5092#  5194#  5295#  5394#  5474#  5565#  5753#  5872#
                    5946#  6017#  6093#  6203#  6431#  6486#  6556#  6749#  6810#  6856#  6922#  6973#  7019#
                    7085#  7211#  7323#
T$TEMP= 000005      1359#  1370#  1371#  1372#  1373#  1374#  1375#  1376#  1377#  1378#  1379#  1380#  1381#
                    1382#  1383#  1384#  1385#  1386#  1387#  1388#  1389#  1390#  1391#  1392#  1393#  1394#
                    1395#  1396#  1397#  1398#  1399#  1400#  1401#  1402#  1403#  1404#  1405#  1406#  1407#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                1408#  1409#  1410#  1411#  1412#  1413#  1447#  1462#  2373#  2402#  2431#  2459#  2487#
                2516#  2545#  2573#  2601#  2629#  2657#  2685#  2714#  2742#  2765#  2793#  2821#  2833#
                2852#  2869#  2886#  2906#  2925#  2950#  2969#  2989#  3005#  3006   3010#  3161#  3187#
                3203#  3223#  3241#  3272#  3273   3289#  3290   3293#  3335#  3336   3356#  3370#  3371
                3393#  3394#  3396#  3406#  3420#  3421   3444#  3445#  3447#  3458#  3473#  3474   3500#
                3501#  3503#  3523#  3524#  3526#  3534#  3583#  3584   3596#  3645#  3646   3658#  3707#
                3708   3740#  3741   3749#  3764#  3765   3833#  3834   3878#  3894#  3895   3922#  3923
                3965#  3986#  3987   4014#  4015#  4017#  4038#  4039#  4041#  4062#  4063#  4065#  4068#
                4088#  4089   4113#  4114#  4116#  4136#  4137#  4139#  4159#  4160#  4162#  4165#  4185#
                4186   4212#  4213#  4215#  4237#  4238#  4240#  4261#  4262#  4264#  4267#  4287#  4288
                4313#  4314#  4316#  4337#  4338#  4340#  4361#  4362#  4364#  4367#  4387#  4388   4414#
                4415#  4417#  4438#  4439#  4441#  4462#  4463#  4465#  4468#  4488#  4489   4515#  4516#
                4518#  4539#  4540#  4542#  4563#  4564#  4566#  4569#  4589#  4590   4615#  4616#  4618#
                4639#  4640#  4642#  4663#  4664#  4666#  4669#  4689#  4690   4715#  4716#  4718#  4739#
                4740#  4742#  4763#  4764#  4766#  4769#  4789#  4790   4815#  4816#  4818#  4841#  4842#
                4844#  4865#  4866#  4868#  4871#  4891#  4892   4917#  4918#  4920#  4943#  4944#  4946#
                4967#  4968#  4970#  4973#  4993#  4994   5018#  5019#  5021#  5044#  5045#  5047#  5068#
                5069#  5071#  5074#  5094#  5095   5119#  5120#  5122#  5145#  5146#  5148#  5169#  5170#
                5172#  5175#  5196#  5197   5221#  5222#  5224#  5247#  5248#  5250#  5271#  5272#  5274#
                5277#  5297#  5298   5322#  5323#  5325#  5348#  5349#  5351#  5372#  5373#  5375#  5378#
                5396#  5397   5449#  5450   5456#  5476#  5477   5516#  5517   5525#  5569#  5570   5661#
                5662   5716#  5757#  5758   5849#  5850   5858#  5874#  5875   5895#  5896   5914#  5915
                5931#  5948#  5949   5969#  5970   5987#  5988   6004#  6077#  6154#  6155   6187#  6319#
                6320   6372#  6373   6398#  6399   6404#  6405   6416#  6472#  6489#  6490   6527#  6528
                6533#  6534   6541#  6560#  6561   6616#  6617   6632#  6633   6646#  6647   6660#  6661
                6676#  6677   6687#  6688   6699#  6700   6712#  6713   6733#  6769#  6770   6795#  6824#
                6825   6841#  6859#  6860   6896#  6897   6904#  6925#  6926   6955#  7001#  7022#  7023
                7068#  7195#  7215#  7221#  7232#  7238#  7328#
T$TEST= 000053  1234#  3257   3261   3264#  3299   3304   3307#  3360   3364   3367#  3410   3414   3417#
                3462   3467   3470#  3539   3543   3546#  3600   3604   3607#  3662   3667   3670#  3753
                3758   3761#  3882   3888   3891#  3969   3979   3982#  4072   4082   4085#  4169   4179
                4182#  4271   4281   4284#  4371   4381   4384#  4472   4482   4485#  4573   4583   4586#
                4673   4683   4686#  4773   4783   4786#  4875   4885   4888#  4977   4987   4990#  5078
                5088   5091#  5179   5190   5193#  5281   5291   5294#  5382   5390   5393#  5461   5470
                5473#  5529   5561   5564#  5719   5749   5752#  5861   5868   5871#  5935   5942   5945#
                6007   6013   6016#  6081   6089   6092#  6191   6199   6202#  6420   6427   6430#  6476
                6482   6485#  6546   6552   6555#  6737   6745   6748#  6799   6806   6809#  6845   6852
                6855#  6908   6918   6921#  6959   6969   6972#  7005   7015   7018#  7072   7081   7084#
                7343
T$TSTM= 177777  1234#  2352   2358   2366   2371   2374   2381   2387   2395   2400   2403   2410   2416
                2424   2429   2432   2439   2445   2452   2457   2460   2467   2473   2480   2485   2488
                2495   2501   2509   2514   2517   2524   2530   2538   2543   2546   2553   2559   2566
                2571   2574   2581   2587   2594   2599   2602   2609   2615   2622   2627   2630   2637
                2643   2650   2655   2658   2665   2671   2678   2683   2686   2693   2699   2707   2712
                2715   2722   2728   2735   2740   2743   2752   2758   2763   2766   2773   2779   2786
                2791   2794   2801   2807   2814   2819   2822   2830   2834   2843   2849   2853   2860
                2866   2870   2877   2883   2887   2897   2903   2907   2916   2922   2926   2935   2941
                2947   2951   2960   2966   2970   2980   2986   2990   3011   3042   3048   3054   3060
                3077   3162   3182   3188   3200   3204   3221   3224   3242   3272   3281   3289   3294
                3329   3335   3349   3357   3370   3379   3388   3393   3397   3407   3420   3429   3439
                3444   3448   3459   3473   3481   3495   3500   3504   3506   3518   3523   3527   3535
                3578   3583   3590   3597   3640   3645   3652   3659   3702   3707   3735   3740   3743
                3750   3764   3800   3825   3833   3836   3872   3879   3894   3917   3922   3940   3959
                3966   3986   3994   4009   4014   4018   4020   4033   4038   4042   4044   4057   4062
                4066   4069   4088   4096   4108   4113   4117   4119   4131   4136   4140   4142   4154
                4159   4163   4166   4185   4193   4206   4212   4216   4218   4231   4237   4241   4243
```

```
                    4256    4261    4265    4268    4287    4295    4308    4313    4317    4319    4332    4337    4341
                    4343    4356    4361    4365    4368    4387    4396    4409    4414    4418    4420    4433    4438
                    4442    4444    4457    4462    4466    4469    4488    4497    4510    4515    4519    4521    4534
                    4539    4543    4545    4558    4563    4567    4570    4589    4597    4610    4615    4619    4621
                    4634    4639    4643    4645    4658    4663    4667    4670    4689    4697    4710    4715    4719
                    4721    4734    4739    4743    4745    4758    4763    4767    4770    4789    4796    4810    4815
                    4819    4821    4836    4841    4845    4847    4860    4865    4869    4872    4891    4898    4912
                    4917    4921    4923    4938    4943    4947    4949    4962    4967    4971    4974    4993    5000
                    5013    5018    5022    5024    5039    5044    5048    5050    5063    5068    5072    5075    5094
                    5101    5114    5119    5123    5125    5140    5145    5149    5151    5164    5169    5173    5176
                    5196    5203    5216    5221    5225    5227    5242    5247    5251    5253    5266    5271    5275
                    5278    5297    5304    5317    5322    5326    5328    5343    5348    5352    5354    5367    5372
                    5376    5379    5396    5440    5449    5457    5476    5508    5516    5526    5569    5626    5661
                    5709    5717    5757    5814    5849    5859    5874    5889    5895    5908    5914    5925    5932
                    5948    5963    5969    5981    5987    5998    6005    6035    6052    6068    6078    6094    6111
                    6133    6146    6154    6163    6176    6188    6314    6319    6345    6367    6372    6393    6398
                    6404    6417    6465    6473    6489    6520    6527    6533    6542    6560    6610    6616    6625
                    6632    6640    6646    6654    6660    6670    6676    6682    6687    6694    6699    6706    6712
                    6734    6769    6788    6796    6819    6824    6833    6842    6859    6889    6896    6905    6925
                    6947    6956    6994    7002    7022    7060    7069    7090    7113    7115    7121    7139    7145
                    7162    7169    7172    7185    7196
T$TSTS= 000001       1234#   3265#   3308#   3368#   3418#   3471#   3547#   3608#   3671#   3762#   3892#   3983#   4086#
                    4183#   4285#   4385#   4486#   4587#   4687#   4787#   4889#   4991#   5092#   5194#   5295#   5394#
                    5474#   5565#   5753#   5872#   5946#   6017#   6093#   6203#   6431#   6486#   6556#   6749#   6810#
                    6856#   6922#   6973#   7019#   7085#
T$$AU  = 010042      3239#   3241
T$$AUT= 010037       3166#   3187
T$$CLE= 010040       3198#   3203
T$$DU  = 010041      3218#   3223
T$$HAR= 010116       7211#   7239
T$$HW  = 010001      1429#   1447
T$$INI= 010036       3025#   3161
T$$MSG= 010034       2347#   2373    2376#   2402    2405#   2431    2434#   2459    2462#   2487    2490#   2516    2519#
                    2545    2548#   2573    2576#   2601    2604#   2629    2632#   2657    2660#   2685    2688#   2714
                    2717#   2742    2747#   2765    2768#   2793    2796#   2821    2825#   2833    2837#   2852    2855#
                    2869    2872#   2886    2890#   2906    2909#   2925    2928#   2950    2953#   2969    2973#   2989
T$$PRO= 010000       1354#
T$$RPT= 010035       3001#   3005    3010
T$$SEG= 010002       3379#   3393    3396#   3429#   3444    3447#   3481#   3500    3503#   3506#   3523    3526#   3994#
                    4014    4017#   4020#   4038    4041#   4044#   4062    4065#   4096#   4113    4116#   4119#   4136
                    4139#   4142#   4159    4162#   4193#   4212    4215#   4218#   4237    4240#   4243#   4261    4264#
                    4295#   4313    4316#   4319#   4337    4340#   4343#   4361    4364#   4396#   4414    4417#   4420#
                    4438    4441#   4444#   4462    4465#   4497#   4515    4518#   4521#   4539    4542#   4545#   4563
                    4566#   4597#   4615    4618#   4621#   4639    4642#   4645#   4663    4666#   4697#   4715    4718#
                    4721#   4739    4742#   4745#   4763    4766#   4796#   4815    4818#   4821#   4841    4844#   4847#
                    4865    4868#   4898#   4917    4920#   4923#   4943    4946#   4949#   4967    4970#   5000#   5018
                    5021#   5024#   5044    5047#   5050#   5068    5071#   5101#   5119    5122#   5125#   5145    5148#
                    5151#   5169    5172#   5203#   5221    5224#   5227#   5247    5250#   5253#   5271    5274#   5304#
                    5322    5325#   5328#   5348    5351#   5354#   5372    5375#
T$$SOF= 010117       7323#   7329
T$$SW  = 010002      1456#   1462
T$$TES= 010115       3265#   3272    3289    3293    3308#   3335    3356    3368#   3370    3406    3418#   3420    3458
                    3471#   3473    3534    3547#   3583    3596    3608#   3645    3658    3671#   3707    3740    3749
                    3762#   3764    3833    3878    3892#   3894    3922    3965    3983#   3986    4068    4086#   4088
                    4165    4183#   4185    4267    4285#   4287    4367    4385#   4387    4468    4486#   4488    4569
                    4587#   4589    4669    4687#   4689    4769    4787#   4789    4871    4889#   4891    4973    4991#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                    4993    5074    5092#   5094    5175    5194#   5196    5277    5295#   5297    5378    5394#   5396
                    5449    5456    5474#   5476    5516    5525    5565#   5569    5661    5716    5753#   5757    5849
                    5858    5872#   5874    5895    5914    5931    5946#   5948    5969    5987    6004    6017#   6077
                    6093#   6154    6187    6203#   6319    6372    6398    6404    6416    6431#   6472    6486#   6489
                    6527    6533    6541    6556#   6560    6616    6632    6646    6660    6676    6687    6699    6712
                    6733    6749#   6769    6795    6810#   6824    6841    6856#   6859    6896    6904    6922#   6925
                    6955    6973#   7001    7019#   7022    7068    7085#   7195
T1          012146 G    1370    3264#
T10         014470 G    1379    3891#
T11         014736 G    1380    3982#
T12         015202 G    1381    4085#
T13         015432 G    1382    4182#
T14         015676 G    1383    4284#
T15         016142 G    1384    4384#
T16         016406 G    1385    4485#
T17         016652 G    1386    4586#
T18         017116 G    1387    4686#
T19         017362 G    1388    4786#
T2          012256 G    1371    3307#
T20         017642 G    1389    4888#
T21         020122 G    1390    4990#
T22         020376 G    1391    5091#
T23         020654 G    1392    5193#
T24         021130 G    1393    5294#
T25         021404 G    1394    5393#
T26         021576 G    1395    5473#
T27         021744 G    1396    5564#
T28         022322 G    1397    5752#
T29         022560 G    1398    5871#
T3          012422 G    1372    3367#
T30         022774 G    1399    5945#
T31         023210 G    1400    6016#
T32         023452 G    1401    6092#
T33         024044 G    1402    6202#
T34         025116 G    1403    6430#
T35         025216 G    1404    6485#
T36         025364 G    1405    6555#
T37         026042 G    1406    6748#
T38         026202 G    1407    6809#
T39         026310 G    1408    6855#
T4          012552 G    1373    3417#
T40         026446 G    1409    6921#
T41         026544 G    1410    6972#
T42         026644 G    1411    7018#
T43         026770 G    1412    7084#
T5          012712 G    1374    3470#
T6          013154 G    1375    3546#
T7          013356 G    1376    3607#
T8          013570 G    1377    3670#
T9          014112 G    1378    3761#
UAM     =   000200 G    1548#
VECTOR      027504      7253#
WPM         027400      7216    7241#
WROM        003422      1988#
WTYPE       002414      1615#   1871    1970    1972    1995    2017    3081*   3147    3149    3154    3489    3846    3985
                        4088    4185    4287    4387    4488    4589    4689    4789    4824    4891    4926    4993    5027
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

```
                      5094   5128   5196   5230   5297   5331   5619   5674   5807   6058   6488   6559   6858
                      6924   7021
XSALWA= 000000         1234#
XSFALS= 000040         1234#
XSOFFS= 000400         1234#
XSTRUE= 000020         1234#
ZERO    002370         1605#
$BDADR  002450         1629#
$BDDAT  002454         1631#
$GDADR  002446         1628#
$GDDAT  002452         1630#  2956   6030*  6031   6033   6303*  6308   6311*  6312   6336*  6341   6343   6358*
                       6363   6365   6384*  6389   6391   7107*  7110   7134*  7157*  7158   7182*
$LSTIN= 000000         1242#
$LSTTA= 000000         1243#
$REG0   002430         1621#  2054*  2360   2389
$REG1   002426         1620#  2053*
$REG2   002424         1619#  2052*  2362   2391   2418   2532   2590   2674
$REG3   002422         1618#  2051*
$REG4   002420         1617#  2050*  2361   2390   2419   2447   2475   2504   2533   2561   2589   2617   2645
                       2673   2701   2730   2781   2809
$REG5   002416         1616#  2049*  2420   2448   2476   2505   2534   2562   2618   2646   2702   2731   2782
                       2810
$TEMP0  002442         1626#
$TMP0   002444         1627#  6021*  6046*  6047   6050   6061   6100*  6130   6137
.       = 030144       1225#  1608#  1609#  1610#  1611#  1644#  1690#  1716#  1854#  3006   3273   3290   3336
                       3371   3394   3421   3445   3474   3501   3524   3584   3646   3708   3741   3765   3834
                       3895   3910   3923   3987   4015   4039   4063   4089   4114   4137   4160   4186   4213
                       4238   4262   4288   4314   4338   4362   4388   4415   4439   4463   4489   4516   4540
                       4564   4590   4616   4640   4664   4690   4716   4740   4764   4790   4816   4842   4866
                       4892   4918   4944   4968   4994   5019   5045   5069   5095   5120   5146   5170   5186
                       5197   5222   5248   5272   5298   5323   5349   5373   5397   5450   5477   5517   5570
                       5662   5758   5850   5875   5896   5915   5949   5970   5988   6155   6320   6373   6399
                       6405   6409#  6413#  6490   6528   6534   6561   6617   6633   6647   6661   6677   6688
                       6700   6713   6770   6825   6860   6897   6926   7023   7303#
.MSTCL  002756         1849#  3310   3425   3479   3551   3612   3675   3770   3900   3993   4094   4191   4293
                       4393   4494   4594   4695   4794   4896   4998   5099   5201   5302   5401   5481   5572
                       5760   5877   5951   6020   6099   6110   6162   6175   6185   6206   6865   6931   7028
                       7088
.REGT   003050         1863#  1872
.ROMCL  003044         1860#  1889   1892   1895   1905   1913   1921   1929   1937   1940   1943   1954   1977
                       1983   3315   3318   3321   3339   3342   3558   3561   3565   3568   3571   3620   3623
                       3627   3630   3633   3682   3685   3689   3692   3695   3719   3722   3725   3728   3774
                       3777   3780   3787   3790   3809   3813   3853   3856   3859   3862   3902   3906   3926
                       3932   3945   3948   3951   5405   5408   5413   5429   5432   5488   5491   5500   5594
                       5597   5604   5607   5686   5689   5692   5696   5782   5785   5792   5795   5881   5899
                       5918   5955   5973   5991   6041   6211   6225   6229   6233   6247   6251   6255   6262
                       6265   6269   6273   6277   6286   6289   6305   6338   6356   6360   6379   6382   6386
                       6434   6437   6440   6443   6448   6452   6866   6869   6877   6881   6933   6937   6981
                       6988   7036   7040   7043   7053   7094   7098   7101   7105   7127   7131   7150   7153
                       7177
.SROMC  003100         1869#  1898   1946   1957   3793   3805   3999   4002   4023   4026   4047   4050   4098
                       4101   4121   4124   4144   4147   4196   4199   4221   4224   4246   4249   4298   4301
                       4322   4325   4346   4349   4399   4402   4423   4426   4447   4450   4500   4503   4524
                       4527   4548   4551   4600   4603   4624   4627   4648   4651   4700   4703   4724   4727
                       4748   4751   4800   4803   4826   4829   4850   4853   4902   4905   4928   4931   4952
                       4955   5003   5006   5029   5032   5053   5056   5104   5107   5130   5133   5154   5157
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- USER SYMBOLS

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- MACRO NAMES

```
BADHEA  1769#  3256  3260  3298  3303  3359  3363  3409  3413  3461  3466  3538  3542  3599  3603
        3661   3666  3752  3757  3881  3887  3968  3978  4071  4081  4168  4178  4270  4280  4370
        4380   4471  4481  4572  4582  4672  4682  4772  4782  4874  4884  4976  4986  5077  5087
        5178   5189  5280  5290  5381  5389  5460  5469  5528  5560  5718  5748  5860  5867  5934
        5941   6006  6012  6080  6088  6190  6198  6419  6426  6475  6481  6545  6551  6736  6744
        6798   6805  6844  6851  6907  6917  6958  6968  7004  7014  7071  7080
BCOMPL     1#  1234#  3043  3049  3055
BERROR     1#  1234#
BGNAU      1#  1234#  3238
BGNAUT     1#  1234#  3165
BGNCLN     1#  1234#  3197
BGNDU      1#  1234#  3217
BGNHRD     1#  1234#  7210
BGNHW      1#  1234#  1428
BGNINI     1#  1234#  3024
BGNMOD     1#  1234#  1239
BGNMSG     1#  1234#  2347  2376  2405  2434  2462  2490  2519  2548  2576  2604  2632  2660  2688
        2717   2747  2768  2796  2824  2836  2854  2871  2889  2908  2927  2952  2972
BGNPRO     1#  1234#  1353
BGNPTA     1#  1234#
BGNRPT     1#  1234#  3000
BGNSEG     1#  1234#  3378  3428  3480  3505  3993  4019  4043  4095  4118  4141  4192  4217  4242
        4294   4318  4342  4395  4419  4443  4496  4520  4544  4596  4620  4644  4696  4720  4744
        4795   4820  4846  4897  4922  4948  4999  5023  5049  5100  5124  5150  5202  5226  5252
        5303   5327  5353
BGNSET     1#  1234#
BGNSFT     1#  1234#  7322
BGNSRV     1#  1234#
BGNSUB     1#  1234#
BGNSW      1#  1234#  1455
BGNTST     1#  1234#  3263  3306  3366  3416  3469  3545  3606  3669  3760  3890  3981  4084  4181
        4283   4383  4484  4585  4685  4785  4887  4989  5090  5192  5293  5392  5472  5563  5751
        5870   5944  6015  6091  6201  6429  6484  6554  6747  6808  6854  6920  6971  7017  7083
BNCOMP     1#  1234#  3061  3079
BNERRO     1#  1234#
BREAK      1#  1234#  3589  3651  3742  3835
BRESET     1#  1234#  3199  3220  6093  7112  7171
CKLOOP     1#  1234#  7120  7144  7168
CLOCK      1#  1234#
CLOSE      1#  1234#
CLRMAR  1809#  3776  3789
CLRVEC     1#  1234#
COMMEN     1#  1234#
DELAY      1#  1234#
DESCRI     1#  1234#  1577
DEVTYP     1#  1234#  1710
DISPAT     1#  1234#  1367
DISPLA     1#  1234#
DOCLN      1#  1234#
DODU       1#  1234#  3180
DORPT      1#  1234#
EDSCAL  1764#  3257  3261  3299  3304  3360  3364  3410  3414  3462  3467  3539  3543  3600  3604
        3662   3667  3753  3758  3882  3888  3969  3979  4072  4082  4169  4179  4271  4281  4371
        4381   4472  4482  4573  4583  4673  4683  4773  4783  4875  4885  4977  4987  5078  5088
        5179   5190  5281  5291  5382  5390  5461  5470  5529  5561  5719  5749  5861  5868  5935
        5942   6007  6013  6081  6089  6191  6199  6420  6427  6476  6482  6546  6552  6737  6745
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- MACRO NAMES

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- MACRO NAMES

```
GETBYT      1#    1234#
GETPRI      1#    1234#
GETWOR      1#    1234#
GMANIA      1#    1234#
GMANID      1#    1234#
GMANIL      1#    1234#
GPHARD      1#    1234#  3075
GPRMA       1#    1234#  7220
GPRMD       1#    1234#  7214  7231
GPRML       1#    1234#
HEADER      1#    1234#  1265
INLOOP      1#    1234#
IOSETU      1#    1234#
IOSTAR      1#    1234#
KT11        1#    1234#
K40NLY   1759#    1798#  3762  5394  5474  5567  5755  5872  5946
LASTAD      1#    1234#  7338
MACEX    1780#    3368   3418  3471  3892
MACEX2   1789#    4086   4183  4285  4385  4486  4587  4687  4787  4889  4991  5092  5194  5295
MANUAL      1#    1234#
MDT0     2317#    2760
MDT1     2297#    2360   2389
MDT2     2301#    2418
MDT3     2305#    2447   2475  2561  2645  2781  2809
MDT4     2309#    2503
MDT5     2313#    2532
MDT6     2319#    2589   2673
MDT7     2323#    2617   2730
MDT8     2326#    2701
MEMORY      1#    1234#
MSTCLR   1845#    3309   3424  3478  3550  3611  3674  3769  3899  3992  4093  4190  4292  4392  4493
         4593     4694   4793  4895  4997  5098  5200  5301  5400  5480  5571  5759  5876  5950  6019
         6098     6109   6161  6174  6184  6205  6864  6930  7027  7087
MYINT    1774#    3372   3422  3475  3547  3608  3671  3766  3896  3989  4090  4187  4289  4389  4490
         4591     4691   4791  4893  4995  5096  5198  5299  5398  5478  5565  5753  5877  5951  6017
         6095     6203   6431  6492  6563  6749  6810  6862  6928  6973  7025  7085
MSBYTE      1#    1234#  1266#  1272  1273  1274
MSCHEC      1#    1234#  3005#  3370#  3420#  3473#  3764#  3894#  3986#  4088#  4185#  4287#  4387#  4488#  4589#
         4689#    4789#  4891#  4993#  5094#  5196#  5297#  5396#  5476#  5569#  5757#  5874#  5948#  6154#  6404#
         6489#    6533#  6560#  6712#  6769#  6859#  6925#  7022#
MSCNTO      1#    1234#  7215#  7221#  7232#
MSCOUN      1#    1234#  2348#  2354#  2360#  2368#  2377#  2383#  2389#  2397#  2406#  2412#  2418#  2426#  2435#
         2441#    2447#  2454#  2463#  2469#  2475#  2482#  2491#  2497#  2503#  2511#  2520#  2526#  2532#  2540#
         2549#    2555#  2561#  2568#  2577#  2583#  2589#  2596#  2605#  2611#  2617#  2624#  2633#  2639#  2645#
         2652#    2661#  2667#  2673#  2680#  2689#  2695#  2701#  2709#  2718#  2724#  2730#  2737#  2748#  2754#
         2760#    2769#  2775#  2781#  2788#  2797#  2803#  2809#  2816#  2827#  2839#  2846#  2857#  2863#  2874#
         2880#    2892#  2900#  2911#  2919#  2930#  2938#  2944#  2955#  2963#  2975#  2983#
MSDATA      1#    1234#  1266#  1275  1277  1279  1281  1283  1285  1287  1289  1291  1293  1295  1297
         1299     1301   1303   1305#  1307  1309  1312  1315  1317  1319  1321  1323  1325  1327  1329
         1331     1333   1335   1337  1339  1341  1343  1345  1347  1349  1578#  1711#
MSDECR      1#    1234#  1359#  1447#  1462#  2373#  2402#  2431#  2459#  2487#  2516#  2545#  2573#  2601#  2629#
         2657#    2685#  2714#  2742#  2765#  2793#  2821#  2833#  2852#  2869#  2886#  2906#  2925#  2950#  2969#
         2989#    3010#  3161#  3187#  3203#  3223#  3241#  3293#  3356#  3396#  3406#  3447#  3458#  3503#  3526#
         3534#    3596#  3658#  3749#  3878#  3965#  4017#  4041#  4065#  4068#  4116#  4139#  4162#  4165#  4215#
         4240#    4264#  4267#  4316#  4340#  4364#  4367#  4417#  4441#  4465#  4468#  4518#  4542#  4566#  4569#
         4618#    4642#  4666#  4669#  4718#  4742#  4766#  4769#  4818#  4844#  4868#  4871#  4920#  4946#  4970#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- MACRO NAMES

```
                4973#  5021#  5047#  5071#  5074#  5122#  5148#  5172#  5175#  5224#  5250#  5274#  5277#  5325#  5351#
                5375#  5378#  5456#  5525#  5716#  5858#  5931#  6004#  6077#  6187#  6416#  6472#  6541#  6733#  6795#
                6841#  6904#  6955#  7001#  7068#  7195#  7238#  7328#
MSDEFA             1#  1234#  7215#  7221#  7232#
MSENDE             1#  1234#  1447#  1462#  2373#  2402#  2431#  2459#  2487#  2516#  2545#  2573#  2601#  2629#  2657#
                2685#  2714#  2742#  2765#  2793#  2821#  2833#  2852#  2869#  2886#  2906#  2925#  2950#  2969#  2989#
                3010#  3161#  3187#  3203#  3223#  3241#  3293#  3356#  3396#  3406#  3447#  3458#  3503#  3526#  3534#
                3596#  3658#  3749#  3878#  3965#  4017#  4041#  4065#  4068#  4116#  4139#  4162#  4165#  4215#  4240#
                4264#  4267#  4316#  4340#  4364#  4367#  4417#  4441#  4465#  4468#  4518#  4542#  4566#  4569#  4618#
                4642#  4666#  4669#  4718#  4742#  4766#  4769#  4818#  4844#  4868#  4871#  4920#  4946#  4970#  4973#
                5021#  5047#  5071#  5074#  5122#  5148#  5172#  5175#  5224#  5250#  5274#  5277#  5325#  5351#  5375#
                5378#  5456#  5525#  5716#  5858#  5931#  6004#  6077#  6187#  6416#  6472#  6541#  6733#  6795#  6841#
                6904#  6955#  7001#  7068#  7195#  7238#  7328#
MSERRI             1#  1234#  3281#  3329#  3349#  3388#  3439#  3495#  3518#  3578#  3640#  3702#  3735#  3800#  3825#
                3872#  3917#  3940#  3959#  4009#  4033#  4057#  4108#  4131#  4154#  4206#  4231#  4256#  4308#  4332#
                4356#  4409#  4433#  4457#  4510#  4534#  4558#  4610#  4634#  4658#  4710#  4734#  4758#  4810#  4836#
                4860#  4912#  4938#  4962#  5013#  5039#  5063#  5114#  5140#  5164#  5216#  5242#  5266#  5317#  5343#
                5367#  5440#  5508#  5626#  5709#  5814#  5889#  5908#  5925#  5963#  5981#  5998#  6035#  6052#  6068#
                6111#  6133#  6146#  6163#  6176#  6314#  6345#  6367#  6393#  6465#  6520#  6610#  6625#  6640#  6654#
                6670#  6682#  6694#  6706#  6788#  6819#  6833#  6889#  6947#  6994#  7060#  7115#  7139#  7162#  7185#
MSESCA             1#  1234#  3272#  3273   3289#  3290   3335#  3336   3393#  3444#  3500#  3523#  3583#  3584   3645#
                3646   3707#  3708   3740#  3741   3833#  3834   3922#  3923   4014#  4038#  4062#  4113#  4136#  4159#
                4212#  4237#  4261#  4313#  4337#  4361#  4414#  4438#  4462#  4515#  4539#  4563#  4615#  4639#  4663#
                4715#  4739#  4763#  4815#  4841#  4865#  4917#  4943#  4967#  5018#  5044#  5068#  5119#  5145#  5169#
                5221#  5247#  5271#  5322#  5348#  5372#  5449#  5450   5516#  5517   5661#  5662   5849#  5850   5895#
                5896   5914#  5915   5969#  5970   5987#  5988   6319#  6320   6372#  6373   6398#  6399   6527#  6528
                6616#  6617   6632#  6633   6646#  6647   6660#  6661   6676#  6677   6687#  6688   6699#  6700   6824#
                6825   6896#  6897
MSESCS             1#  1234#  3272#  3289#  3335#  33  3#  3394   3444#  3445   3500#  3501   3523#  3524   3583#  3645#
                3707#  3740#  3833#  3922#  4014#  40  5   4038#  4039   4062#  4063   4113#  4114   4136#  4137   4159#
                4160   4212#  4213   4237#  4238   4261#  4262   4313#  4314   4337#  4338   4361#  4362   4414#  4415
                4438#  4439   4462#  4463   4515#  4516   4539#  4540   4563#  4564   4615#  4616   4639#  4640   4663#
                4664   4715#  4716   4739#  4740   4763#  4764   4815#  4816   4841#  4842   4865#  4866   4917#  4918
                4943#  4944   4967#  4968   5018#  5019   5044#  5045   5068#  5069   5119#  5120   5145#  5146   5169#
                5170   5221#  5222   5247#  5248   5271#  5272   5322#  5323   5348#  5349   5372#  5373   5449#  5516#
                5661#  5849#  5895#  5914#  5969#  5987#  6319#  6372#  6398#  6527#  6616#  6632#  6646#  6660#  6676#
                6687#  6699#  6824#  6896#
MSEXCP             1#  1234#  7215#  7221#  7232#
MSEXIT             1#  1234#  3005#  3370#  3371   3420#  3421   3473#  3474   3764#  3765   3894#  3895   3986#  3987
                4088#  4089   4185#  4186   4287#  4288   4387#  4388   4488#  4489   4589#  4590   4689#  4690   4789#
                4790   4891#  4892   4993#  4994   5094#  5095   5196#  5197   5297#  5298   5396#  5397   5476#  5477
                5569#  5570   5757#  5758   5874#  5875   5948#  5949   6154#  6155   6404#  6405   6489#  6490   6533#
                6534   6560#  6561   6712#  6713   6769#  0770   6859#  6860   6925#  6926   7022#  7023
MSEXSE             1#  1234#  3005#  3370#  3420#  3473#  3764#  3894#  3986#  4088#  4185#  4287#  4387#  4488#  4589#
                4689#  4789#  4891#  4993#  5094#  5196#  5297#  5396#  5476#  5569#  5757#  5874#  5948#  6154#  6404#
                6489#  6533#  6560#  6712#  6769#  6859#  6925#  7022#
MSEXTJ             1#  1234#  3005#  3006   3370#  3420#  3473#  3764#  3894#  3986#  4088#  4185#  4287#  4387#  4488#
                4589#  4689#  4789#  4891#  4993#  5094#  5196#  5297#  5396#  5476#  5569#  5757#  5874#  5948#  6154#
                6404#  6489#  6533#  6560#  6712#  6769#  6859#  6925#  7022#
MSGEN              1#  1234#  1240#  1266#  1275#  1277#  1279#  1281#  1283#  1285#  1287#  1289#  1291#  1293#  1295#
                1297#  1299#  1301#  1303#  1305#  1307#  1309#  1312#  1315#  1317#  1319#  1321#  1323#  1325#  1327#
                1329#  1331#  1333#  1335#  1337#  1339#  1341#  1343#  1345#  1347#  1349#  1354#  1369#  1430#  1431#
                1447#  1457#  1458#  1462#  1578#  1711#  2347#  2373#  2376#  2402#  2405#  2431#  2434#  2459#  2462#
                2487#  2490#  2516#  2519#  2545#  2548#  2573#  2576#  2601#  2604#  2629#  2632#  2657#  2660#  2685#
                2688#  2714#  2717#  2742#  2747#  2765#  2768#  2793#  2796#  2821#  2825#  2833#  2837#  2852#  2855#
                2869#  2872#  2886#  2890#  2906#  2909#  2925#  2928#  2950#  2953#  2969#  2973#  2989#  3001#  3010#
```

```
                3025#  3161#  3166#  3187#  3198#  3203#  3218#  3223#  3239#  3241#  3264#  3293#  3307#  3356#  3367#
                3396#  3406#  3417#  3447#  3458#  3470#  3503#  3526#  3534#  3546#  3596#  3607#  3658#  3670#  3749#
                3761#  3878#  3891#  3965#  3982#  4017#  4041#  4065#  4068#  4085#  4116#  4139#  4162#  4165#  4182#
                4215#  4240#  4264#  4267#  4284#  4316#  4340#  4364#  4367#  4384#  4417#  4441#  4465#  4468#  4485#
                4518#  4542#  4566#  4569#  4586#  4618#  4642#  4666#  4669#  4686#  4718#  4742#  4766#  4769#  4786#
                4818#  4844#  4868#  4871#  4888#  4920#  4946#  4970#  4973#  4990#  5021#  5047#  5071#  5074#  5091#
                5122#  5148#  5172#  5175#  5193#  5224#  5250#  5274#  5277#  5294#  5325#  5351#  5375#  5378#  5393#
                5456#  5473#  5525#  5564#  5716#  5752#  5858#  5871#  5931#  5945#  6004#  6016#  6077#  6092#  6187#
                6202#  6416#  6430#  6472#  6485#  6541#  6555#  6733#  6748#  6795#  6809#  6841#  6855#  6904#  6921#
                6955#  6972#  7001#  7018#  7068#  7084#  7195#  7212#  7239#  7324#  7329#  7342#
MSGENB             1#  1234#
MSGETS             1#  1234#  1359#  1447#  1462#  2373#  2402#  2431#  2459#  2487#  2516#  2545#  2573#  2601#  2629#
                2657#  2685#  2714#  2742#  2765#  2793#  2821#  2833#  2852#  2869#  2886#  2906#  2925#  2950#  2969#
                2989#  3010#  3161#  3187#  3203#  3223#  3241#  3293#  3356#  3394#  3396#  3406#  3445#  3447#  3458#
                3501#  3503#  3524#  3526#  3534#  3596#  3658#  3749#  3878#  3965#  4015#  4017#  4039#  4041#  4063#
                4065#  4068#  4114#  4116#  4137#  4139#  4160#  4162#  4165#  4213#  4215#  4238#  4240#  4262#  4264#
                4267#  4314#  4316#  4338#  4340#  4362#  4364#  4367#  4415#  4417#  4439#  4441#  4463#  4465#  4468#
                4516#  4518#  4540#  4542#  4564#  4566#  4569#  4616#  4618#  4640#  4642#  4664#  4666#  4669#  4716#
                4718#  4740#  4742#  4764#  4766#  4769#  4816#  4818#  4842#  4844#  4866#  4868#  4871#  4918#  4920#
                4944#  4946#  4968#  4970#  4973#  5019#  5021#  5045#  5047#  5069#  5071#  5074#  5120#  5122#  5146#
                5148#  5170#  5172#  5175#  5222#  5224#  5248#  5250#  5272#  5274#  5277#  5323#  5325#  5349#  5351#
                5373#  5375#  5378#  5456#  5525#  5716#  5858#  5931#  6004#  6077#  6187#  6416#  6472#  6541#  6733#
                6795#  6841#  6904#  6955#  7001#  7068#  7195#  7238#  7328#
MSGETT             1#  1234#  3005#  3272#  3289#  3335#  3370#  3393#  3394   3420#  3444#  3445   3473#  3500#  3501
                3523#  3524   3583#  3645#  3707#  3740#  3764#  3833#  3894#  3922#  3986#  4014#  4015   4038#  4039
                4062#  4063   4088#  4113#  4114   4136#  4137   4159#  4160   4185#  4212#  4213   4237#  4238   4261#
                4262   4287#  4313#  4314   4337#  4338   4361#  4362   4387#  4414#  4415   4438#  4439   4462#  4463
                4488#  4515#  4516   4539#  4540   4563#  4564   4589#  4615#  4616   4639#  4640   4663#  4664   4689#
                4715#  4716   4739#  4740   4763#  4764   4789#  4815#  4816   4841#  4842   4865#  4866   4891#  4917#
                4918   4943#  4944   4967#  4968   4993#  5018#  5019   5044#  5045   5068#  5069   5094#  5119#  5120
                5145#  5146   5169#  5170   5196#  5221#  5222   5247#  5248   5271#  5272   5297#  5322#  5323   5348#
                5349   5372#  5373   5396#  5449#  5476#  5516#  5569#  5661#  5757#  5849#  5874#  5895#  5914#  5948#
                5969#  5987#  6154#  6319#  6372#  6398#  6404#  6489#  6527#  6533#  6560#  6616#  6632#  6646#  6660#
                6676#  6687#  6699#  6712#  6769#  6824#  6859#  6896#  6925#  7022#
MSGNGB             1#  1234#  1240#  1266#  1275#  1277#  1279#  1281#  1283#  1285#  1287#  1289#  1291#  1293#  1295#
                1297#  1299#  1301#  1303#  1305#  1307#  1309#  1312#  1315#  1317#  1319#  1321#  1323#  1325#  1327#
                1329#  1331#  1333#  1335#  1337#  1339#  1341#  1343#  1345#  1347#  1349#  1354#  1368#  1369   1429#
                1430   1431   1456#  1457   1458   1578#  1711#  2347#  2376#  2405#  2434#  2462#  2490#  2519#  2548#
                2576#  2604#  2632#  2660#  2688#  2717#  2747#  2768#  2796#  2825#  2837#  2855#  2872#  2890#  2909#
                2928#  2953#  2973#  3001#  3025#  3166#  3198#  3218#  3239#  7211#  7212   7323#  7324   7339#  7342
MSGNIN             1#  1234#  1266#  1267   1268   1269   1270   1271   1272#  1273#  1274#  1275#  1276   1277#  1278
                1279#  1280   1281#  1282   1283#  1284   1285#  1286   1287#  1288   1289#  1290   1291#  1292   1293#
                1294   1295#  1296   1297#  1298   1299#  1300   1301#  1302   1303#  1304   1305#  1306   1307#  1308
                1309#  1310   1311   1312#  1313   1314#  1315#  1316   1317#  1318   1319#  1320   1321#  1322   1323#
                1324   1325#  1326   1327#  1328   1329#  1330   1331#  1332   1333#  1334   1335#  1336   1337#  1338
                1339#  1340   1341#  1342   1343#  1344   1345#  1346   1347#  1348   1349#  1350   1368#  1370#  1371#
                1372#  1373#  1374#  1375#  1376#  1377#  1378#  1379#  1380#  1381#  1382#  1383#  1384#  1385#  1386#
                1387#  1388#  1389#  1390#  1391#  1392#  1393#  1394#  1395#  1396#  1397#  1398#  1399#  1400#  1401#
                1402#  1403#  1404#  1405#  1406#  1407#  1408#  1409#  1410#  1411#  1412#  1429#  1456#  1578#  1579
                1583   1711#  1712   1716   2348#  2349#  2350#  2351   2352#  2353   2354#  2355#  2356#  2357   2358#
                2359   2360#  2361#  2362#  2363#  2364#  2365   2366#  2367   2368#  2369#  2370   2371#  2372   2374#
                2377#  2378#  2379#  2380   2381#  2382   2383#  2384#  2385#  2386   2387#  2388   2389#  2390#  2391#
                2392#  2393#  2394   2395#  2396   2397#  2398#  2399   2400#  2401   2403#  2406#  2407#  2408#  2409
                2410#  2411   2412#  2413#  2414#  2415   2416#  2417   2418#  2419#  2420#  2421#  2422#  2423   2424#
                2425   2426#  2427#  2428   2429#  2430   2432#  2435#  2436#  2437#  2438   2439#  2440   2441#  2442#
                2443#  2444   2445#  2446   2447#  2448#  2449#  2450#  2451   2452#  2453   2454#  2455#  2456   2457#
```

```
                4923#  4938#  4939#  4940#  4941#  4943#  4944#  4947#  4949#  4962#  4963#  4964#  4965#  4967#  4968#
                4971#  4974#  4993#  4994#  5000#  5013#  5014#  5015#  5016#  5018#  5019#  5022#  5024#  5039#  5040#
                5041#  5042#  5044#  5045#  5048#  5050#  5063#  5064#  5065#  5066#  5068#  5069#  5072#  5075#  5094#
                5095#  5101#  5114#  5115#  5116#  5117#  5119#  5120#  5123#  5125#  5140#  5141#  5142#  5143#  5145#
                5146#  5149#  5151#  5164#  5165#  5166#  5167#  5169#  5170#  5173#  5176#  5196#  5197#  5203#  5216#
                5217#  5218#  5219#  5221#  5222#  5225#  5227#  5242#  5243#  5244#  5245#  5247#  5248#  5251#  5253#
                5266#  5267#  5268#  5269#  5271#  5272#  5275#  5278#  5297#  5298#  5304#  5317#  5318#  5319#  5320#
                5322#  5323#  5326#  5328#  5343#  5344#  5345#  5346#  5348#  5349#  5352#  5354#  5367#  5368#  5369#
                5370#  5372#  5373#  5376#  5379#  5396#  5397#  5440#  5441#  5442#  5443#  5449#  5450#  5457#  5476#
                5477#  5508#  5509#  5510#  5511#  5516#  5517#  5526#  5569#  5570#  5626#  5627#  5628#  5629#  5661#
                5662#  5709#  5710#  5711#  5712#  5717#  5757#  5758#  5814#  5815#  5816#  5817#  5849#  5850#  5859#
                5874#  5875#  5889#  5890#  5891#  5892#  5895#  5896#  5908#  5909#  5910#  5911#  5914#  5915#  5925#
                5926#  5927#  5928#  5932#  5948#  5949#  5963#  5964#  5965#  5966#  5969#  5970#  5981#  5982#  5983#
                5984#  5987#  5988#  5998#  5999#  6000#  6001#  6005#  6035#  6036#  6037#  6038#  6052#  6053#  6054#
                6055#  6068#  6069#  6070#  6071#  6078#  6094#  6111#  6112#  6113#  6114#  6133#  6134#  6135#  6136#
                6146#  6147#  6148#  6149#  6154#  6155#  6163#  6164#  6165#  6166#  6176#  6177#  6178#  6179#  6188#
                6314#  6315#  6316#  6317#  6319#  6320#  6345#  6346#  6347#  6348#  6367#  6368#  6369#  6370#  6372#
                6373#  6393#  6394#  6395#  6396#  6398#  6399#  6404#  6405#  6417#  6465#  6466#  6467#  6468#  6473#
                6489#  6490#  6520#  6521#  6522#  6523#  6527#  6528#  6533#  6534#  6542#  6560#  6561#  6610#  6611#
                6612#  6613#  6616#  6617#  6625#  6626#  6627#  6628#  6632#  6633#  6640#  6641#  6642#  6643#  6646#
                6647#  6654#  6655#  6656#  6657#  6660#  6661#  6670#  6671#  6672#  6673#  6676#  6677#  6682#  6683#
                6684#  6685#  6687#  6688#  6694#  6695#  6696#  6697#  6699#  6700#  6706#  6707#  6708#  6709#  6712#
                6713#  6734#  6769#  6770#  6788#  6789#  6790#  6791#  6796#  6819#  6820#  6821#  6822#  6824#  6825#
                6833#  6834#  6835#  6836#  6842#  6859#  6860#  6889#  6890#  6891#  6892#  6896#  6897#  6905#  6925#
                6926#  6947#  6948#  6949#  6950#  6956#  6994#  6995#  6996#  6997#  7002#  7022#  7023#  7060#  7061#
                7062#  7063#  7069#  7089#  7090#  7113#  7115#  7116#  7117#  7118#  7121#  7139#  7140#  7141#  7142#
                7145#  7162#  7163#  7164#  7165#  7169#  7172#  7185#  7186#  7187#  7188#  7196#  7211#  7215#  7216
                7217   7218   7219   7221#  7222   7223   7224   7232#  7233   7234   7235   7236   7238#  7323#  7328#
                7339#  7340#  7341#
MSGNLS             1#  1234#  3396#  3447#  3503#  3526#  4017#  4041#  4065#  4116#  4139#  4162#  4215#  4240#  4264#
                4316#  4340#  4364#  4417#  4441#  4465#  4518#  4542#  4566#  4618#  4642#  4666#  4718#  4742#  4766#
                4818#  4844#  4868#  4920#  4946#  4970#  5021#  5047#  5071#  5122#  5148#  5172#  5224#  5250#  5274#
                5325#  5351#  5375#
MSGNSU             1#  1234#
MSGNTA             1#  1234#  1447#  1462#  2373#  2402#  2431#  2459#  2487#  2516#  2545#  2573#  2601#  2629#  2657#
                2685#  2714#  2742#  2765#  2793#  2821#  2833#  2852#  2869#  2886#  2906#  2925#  2950#  2969#  2989#
                3010#  3161#  3187#  3203#  3223#  3241#  3293#  3356#  3406#  3458#  3534#  3596#  3658#  3749#  3878#
                3965#  4068#  4165#  4267#  4367#  4468#  4569#  4669#  4769#  4871#  4973#  5074#  5175#  5277#  5378#
                5456#  5525#  5716#  5858#  5931#  6004#  6077#  6187#  6416#  6472#  6541#  6733#  6795#  6841#  6904#
                6955#  7001#  7068#  7195#  7238#  7239   7328#  7329
MSGNTE             1#  1234#  3264#  3307#  3367#  3417#  3470#  3546#  3607#  3670#  3761#  3891#  3982#  4085#  4182#
                4284#  4384#  4485#  4586#  4686#  4786#  4888#  4990#  5091#  5193#  5294#  5393#  5473#  5564#  5752#
                5871#  5945#  6016#  6092#  6202#  6430#  6485#  6555#  6748#  6809#  6855#  6921#  6972#  7018#  7084#
MSHAPT             1#  1234#  1266#
MSHNAP             1#  1234#  1266#  1305
MSINCR             1#  1234#  1240#  1354#  1429#  1456#  2347#  2352#  2358#  2366#  2371#  2374#  2376#  2381#  2387#
                2395#  2400#  2403#  2405#  2410#  2416#  2424#  2429#  2432#  2434#  2439#  2445#  2452#  2457#  2460#
                2462#  2467#  2473#  2480#  2485#  2488#  2490#  2495#  2501#  2509#  2514#  2517#  2519#  2524#  2530#
                2538#  2543#  2546#  2548#  2553#  2559#  2566#  2571#  2574#  2576#  2581#  2587#  2594#  2599#  2602#
                2604#  2609#  2615#  2622#  2627#  2630#  2632#  2637#  2643#  2650#  2655#  2658#  2660#  2665#  2671#
                2678#  2683#  2686#  2688#  2693#  2699#  2707#  2712#  2715#  2717#  2722#  2728#  2735#  2740#  2743#
                2747#  2752#  2758#  2763#  2766#  2768#  2773#  2779#  2786#  2791#  2794#  2796#  2801#  2807#  2814#
                2819#  2822#  2825#  2830#  2834#  2837#  2843#  2849#  2853#  2855#  2860#  2866#  2870#  2872#  2877#
                2883#  2887#  2890#  2897#  2903#  2907#  2909#  2916#  2922#  2926#  2928#  2935#  2941#  2947#  2951#
                2953#  2960#  2966#  2970#  2973#  2980#  2986#  2990#  3001#  3011#  3025#  3042#  3048#  3054#  3060#
                3077#  3162#  3166#  3182#  3188#  3198#  3200#  3204#  3218#  3221#  3224#  3239#  3242#  3264#  3265#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- MACRO NAMES

```
                3272#  3281#  3289#  3294#  3307#  3308#  3329#  3335#  3349#  3357#  3367#  3368#  3370#  3379#  3388#
                3393#  3397#  3407#  3417#  3418#  3420#  3429#  3439#  3444#  3448#  3459#  3470#  3471#  3473#  3481#
                3495#  3500#  3504#  3506#  3518#  3523#  3527#  3535#  3546#  3547#  3578#  3583#  3590#  3597#  3607#
                3608#  3640#  3645#  3652#  3659#  3670#  3671#  3702#  3707#  3735#  3740#  3743#  3750#  3761#  3762#
                3764#  3800#  3825#  3833#  3836#  3872#  3879#  3891#  3892#  3894#  3917#  3922#  3940#  3959#  3966#
                3982#  3983#  3986#  3994#  4009#  4014#  4018#  4020#  4033#  4038#  4042#  4044#  4057#  4062#  4066#
                4069#  4085#  4086#  4088#  4096#  4108#  4113#  4117#  4119#  4131#  4136#  4140#  4142#  4154#  4159#
                4163#  4166#  4182#  4183#  4185#  4193#  4206#  4212#  4216#  4218#  4231#  4237#  4241#  4243#  4256#
                4261#  4265#  4268#  4284#  4285#  4287#  4295#  4308#  4313#  4317#  4319#  4332#  4337#  4341#  4343#
                4356#  4361#  4365#  4368#  4384#  4385#  4387#  4396#  4409#  4414#  4418#  4420#  4433#  4438#  4442#
                4444#  4457#  4462#  4466#  4469#  4485#  4486#  4488#  4497#  4510#  4515#  4519#  4521#  4534#  4539#
                4543#  4545#  4558#  4563#  4567#  4570#  4586#  4587#  4589#  4597#  4610#  4615#  4619#  4621#  4634#
                4639#  4643#  4645#  4658#  4663#  4667#  4670#  4686#  4687#  4689#  4697#  4710#  4715#  4719#  4721#
                4734#  4739#  4743#  4745#  4758#  4763#  4767#  4770#  4786#  4787#  4789#  4796#  4810#  4815#  4819#
                4821#  4836#  4841#  4845#  4847#  4860#  4865#  4869#  4872#  4888#  4889#  4891#  4898#  4912#  4917#
                4921#  4923#  4938#  4943#  4947#  4949#  4962#  4967#  4971#  4974#  4990#  4991#  4993#  5000#  5013#
                5018#  5022#  5024#  5039#  5044#  5048#  5050#  5063#  5068#  5072#  5075#  5091#  5092#  5094#  5101#
                5114#  5119#  5123#  5125#  5140#  5145#  5149#  5151#  5164#  5169#  5173#  5176#  5193#  5194#  5196#
                5203#  5216#  5221#  5225#  5227#  5242#  5247#  5251#  5253#  5266#  5271#  5275#  5278#  5294#  5295#
                5297#  5304#  5317#  5322#  5326#  5328#  5343#  5348#  5352#  5354#  5367#  5372#  5376#  5379#  5393#
                5394#  5396#  5440#  5449#  5457#  5473#  5474#  5476#  5508#  5516#  5526#  5564#  5565#  5569#  5626#
                5661#  5709#  5717#  5752#  5753#  5757#  5814#  5849#  5859#  5871#  5872#  5874#  5889#  5895#  5908#
                5914#  5925#  5932#  5945#  5946#  5948#  5963#  5969#  5981#  5987#  5998#  6005#  6016#  6017#  6035#
                6052#  6068#  6078#  6092#  6093#  6094#  6111#  6133#  6146#  6154#  6163#  6176#  6188#  6202#  6203#
                6314#  6319#  6345#  6367#  6372#  6393#  6398#  6404#  6417#  6430#  6431#  6465#  6473#  6485#  6486#
                6489#  6520#  6527#  6533#  6542#  6555#  6556#  6560#  6610#  6616#  6625#  6632#  6640#  6646#  6654#
                6660#  6670#  6676#  6682#  6687#  6694#  6699#  6706#  6712#  6734#  6748#  6749#  6769#  6788#  6796#
                6809#  6810#  6819#  6824#  6833#  6842#  6855#  6856#  6859#  6889#  6896#  6905#  6921#  6922#  6925#
                6947#  6956#  6972#  6973#  6994#  7002#  7018#  7019#  7022#  7060#  7069#  7084#  7 )85#  7090#  7113#
                7115#  7121#  7139#  7145#  7162#  7169#  7172#  7185#  7196#  7211#  7323#

MSIOSE             1#  1234#
MSLDRO             1#  1234#  3041#  3047#  3053#  3059#  3076#  3181#  7089#
MSMASK             1#  1234#
MSMCHI             1#  1234#
MSMCLO             1#  1234#
MSMSK1             1#  1234#
MSPOP              1#  1234#  1359#  1447#  1462#  2373#  2402#  2431#  2459#  2487#  2516#  2545#  2573#  2601#  2629#
                2657#  2685#  2714#  2742#  2765#  2793#  2821#  2833#  2852#  2869#  2886#  2906#  2925#  2950#  2969#
                2989#  3010#  3161#  3187#  3203#  3223#  3241#  3293#  3356#  3396#  3406#  3447#  3458#  3503#  3526#
                3534#  3596#  3658#  3749#  3878#  3965#  4017#  4041#  4065#  4068#  4116#  4139#  4162#  4165#  4215#
                4240#  4264#  4267#  4316#  4340#  4364#  4367#  4417#  4441#  4465#  4468#  4518#  4542#  4566#  4569#
                4618#  4642#  4666#  4669#  4718#  4742#  4766#  4769#  4818#  4844#  4868#  4871#  4920#  4946#  4970#
                4973#  5021#  5047#  5071#  5074#  5122#  5148#  5172#  5175#  5224#  5250#  5274#  5277#  5325#  5351#
                5375#  5378#  5456#  5525#  5716#  5858#  5931#  6004#  6077#  6187#  6416#  6472#  6541#  6733#  6795#
                6841#  6904#  6955#  7001#  7068#  7195#  7238#  7328#

MSPRIN             1#  1234#  2348#  2354#  2360#  2368#  2377#  2383#  2389#  2397#  2406#  2412#  2418#  2426#  2435#
                2441#  2447#  2454#  2463#  2469#  2475#  2482#  2491#  2497#  2503#  2511#  2520#  2526#  2532#  2540#
                2549#  2555#  2561#  2568#  2577#  2583#  2589#  2596#  2605#  2611#  2617#  2624#  2633#  2639#  2645#
                2652#  2661#  2667#  2673#  2680#  2689#  2695#  2701#  2709#  2718#  2724#  2730#  2737#  2748#  2754#
                2760#  2769#  2775#  2781#  2788#  2797#  2803#  2809#  2816#  2827#  2839#  2846#  2857#  2863#  2874#
                2880#  2892#  2900#  2911#  2919#  2930#  2938#  2944#  2955#  2963#  2975#  2983#

MSPUSH             1#  1234#  1240#  1354#  1429#  1456#  2347#  2376#  2405#  2434#  2462#  2490#  2519#  2548#  2576#
                2604#  2632#  2660#  2688#  2717#  2747#  2768#  2796#  2825#  2837#  2855#  2872#  2890#  2909#  2928#
                2953#  2973#  3001#  3025#  3166#  3198#  3218#  3239#  3264#  3265   3307#  3308   3367#  3368   3379#
                3417#  3418   3429#  3470#  3471   3481#  3506#  3546#  3547   3607#  3608   3670#  3671   3761#  3762
                3891#  3892   3982#  3983   3994#  4020#  4044#  4085#  4086   4096#  4119#  4142#  4182#  4183   4193#
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- MACRO NAMES

```
                4218#  4243#  4284#  4285   4295#  4319#  4343#  4384#  4385   4396#  4420#  4444#  4485#  4486   4497#
                4521#  4545#  4586#  4587   4597#  4621#  4645#  4686#  4687   4697#  4721#  4745#  4786#  4787   4796#
                4821#  4847#  4888#  4889   4898#  4923#  4949#  4990#  4991   5000#  5024#  5050#  5091#  5092   5101#
                5125#  5151#  5193#  5194   5203#  5227#  5253#  5294#  5295   5304#  5328#  5354#  5393#  5394   5473#
                5474   5564#  5565   5752#  5753   5871#  5872   5945#  5946   6016#  6017   6092#  6093   6202#  6203
                6430#  6431   6485#  6486   6555#  6556   6748#  6749   6809#  6810   6855#  6856   6921#  6922   6972#
                6973   7018#  7019   7084#  7085   7211#  7323#
MSPUT             1#   1234#  2348#  2354#  2360#  2368#  2377#  2383#  2389#  2397#  2406#  2412#  2418#  2426#  2435#
                2441#  2447#  2454#  2463#  2469#  2475#  2482#  2491#  2497#  2503#  2511#  2520#  2526#  2532#  2540#
                2549#  2555#  2561#  2568#  2577#  2583#  2589#  2596#  2605#  2611#  2617#  2624#  2633#  2639#  2645#
                2652#  2661#  2667#  2673#  2680#  2689#  2695#  2701#  2709#  2718#  2724#  2730#  2737#  2748#  2754#
                2760#  2769#  2775#  2781#  2788#  2797#  2803#  2809#  2816#  2827#  2839#  2846#  2857#  2863#  2874#
                2880#  2892#  2900#  2911#  2919#  2930#  2938#  2944#  2955#  2963#  2975#  2983#
MSPUT1            1#   1234#  2348#  2349   2350   2354#  2355   2356   2360#  2361   2362   2363   2364   2368#  2369
                2377#  2378   2379   2383#  2384   2385   2389#  2390   2391   2392   2393   2397#  2398   2406#  2407
                2408   2412#  2413   2414   2418#  2419   2420   2421   2422   2426#  2427   2435#  2436   2437   2441#
                2442   2443   2447#  2448   2449   2450   2454#  2455   2463#  2464   2465   2469#  2470   2471   2475#
                2476   2477   2478   2482#  2483   2491#  2492   2493   2497#  2498   2499   2503#  2504   2505   2506
                2507   2511#  2512   2520#  2521   2522   2526#  2527   2528   2532#  2533   2534   2535   2536   2540#
                2541   2549#  2550   2551   2555#  2556   2557   2561#  2562   2563   2564   2568#  2569   2577#  2578
                2579   2583#  2584   2585   2589#  2590   2591   2592   2596#  2597   2605#  2606   2607   2611#  2612
                2613   2617#  2618   2619   2620   2624#  2625   2633#  2634   2635   2639#  2640   2641   2645#  2646
                2647   2648   2652#  2653   2661#  2662   2663   2667#  2668   2669   2673#  2674   2675   2676   2680#
                2681   2689#  2690   2691   2695#  2696   2697   2701#  2702   2703   2704   2705   2709#  2710   2718#
                2719   2720   2724#  2725   2726   2730#  2731   2732   2733   2737#  2738   2748#  2749   2750   2754#
                2755   2756   2760#  2761   2769#  2770   2771   2775#  2776   2777   2781#  2782   2783   2784   2788#
                2789   2797#  2798   2799   2803#  2804   2805   2809#  2810   2811   2812   2816#  2817   2827#  2828
                2839#  2840   2841   2846#  2847   2857#  2858   2863#  2864   2874#  2875   2880#  2881   2892#  2893
                2894   2895   2900#  2901   2911#  2912   2913   2914   2919#  2920   2930#  2931   2932   2933   2938#
                2939   2944#  2945   2955#  2956   2957   2958   2963#  2964   2975#  2976   2977   2978   2983#  2984
MSRADI            1#   1234#  7215#  7221#  7232#
MSRBR0            1#   1234#
MSRNR0            1#   1234#  3076#  3078
MSSETS            1#   1234#  1240#  1354#  1429#  1456#  2347#  2376#  2405#  2434#  2462#  2490#  2519#  2548#  2576#
                2604#  2632#  2660#  2688#  2717#  2747#  2768#  2796#  2825#  2837#  2855#  2872#  2890#  2909#  2928#
                2953#  2973#  3001#  3025#  3166#  3198#  3218#  3239#  3265#  3308#  3368#  3379#  3418#  3429#  3471#
                3481#  3506#  3547#  3608#  3671#  3762#  3892#  3983#  3994#  4020#  4044#  4086#  4096#  4119#  4142#
                4183#  4193#  4218#  4243#  4285#  4295#  4319#  4343#  4385#  4396#  4420#  4444#  4486#  4497#  4521#
                4545#  4587#  4597#  4621#  4645#  4687#  4697#  4721#  4745#  4787#  4796#  4821#  4847#  4889#  4898#
                4923#  4949#  4991#  5000#  5024#  5050#  5092#  5101#  5125#  5151#  5194#  5203#  5227#  5253#  5295#
                5304#  5328#  5354#  5394#  5474#  5565#  5753#  5872#  5946#  6017#  6093#  6203#  6431#  6486#  6556#
                6749#  6810#  6856#  6922#  6973#  7019#  7085#  7211#  7323#
MSSTAR            1#   1234#
MSSVC             1#   1234#  2348#  2352   2354#  2358   2360#  2366   2368#  2371   2373#  2374   2377#  2381   2383#
                2387   2389#  2395   2397#  2400   2402#  2403   2406#  2410   2412#  2416   2418#  2424   2426#  2429
                2431#  2432   2435#  2439   2441#  2445   2447#  2452   2454#  2457   2459#  2460   2463#  2467   2469#
                2473   2475#  2480   2482#  2485   2487#  2488   2491#  2495   2497#  2501   2503#  2509   2511#  2514
                2516#  2517   2520#  2524   2526#  2530   2532#  2538   2540#  2543   2545#  2546   2549#  2553   2555#
                2559   2561#  2566   2568#  2571   2573#  2574   2577#  2581   2583#  2587   2589#  2594   2596#  2599
                2601#  2602   2605#  2609   2611#  2615   2617#  2622   2624#  2627   2629#  2630   2633#  2637   2639#
                2643   2645#  2650   2652#  2655   2657#  2658   2661#  2665   2667#  2671   2673#  2678   2680#  2683
                2685#  2686   2689#  2693   2695#  2699   2701#  2707   2709#  2712   2714#  2715   2718#  2722   2724#
                2728   2730#  2735   2737#  2740   2742#  2743   2748#  2752   2754#  2758   2760#  2763   2765#  2766
                2769#  2773   2775#  2779   2781#  2786   2788#  2791   2793#  2794   2797#  2801   2803#  2807   2809#
                2814   2816#  2819   2821#  2822   2827#  2830   2833#  2834   2839#  2843   2846#  2849   2852#  2853
                2857#  2860   2863#  2866   2869#  2870   2874#  2877   2880#  2883   2886#  2887   2892#  2897   2900#
```

```
                2903    2906#   2907    2911#   2916    2919#   2922    2925#   2926    2930#   2935    2938#   2941    2944#   2947
                2950#   2951    2955#   2960    2963#   2966    2969#   2970    2975#   2980    2983#   2986    2989#   2990    3005#
                3010#   3011    3041#   3042    3047#   3048    3053#   3054    3059#   3060    3076#   3077    3161#   3162    3181#
                3182    3187#   3188    3200#   3203#   3204    3221#   3223#   3224    3241#   3242    3272#   3281    3289#   3293#
                3294    3329    3335#   3349    3356#   3357    3370#   3379#   3388    3393#   3396#   3397    3406#   3407    3420#
                3429#   3439    3444#   3447#   3448    3458#   3459    3473#   3481#   3495    3500#   3503#   3504    3506#   3518
                3523#   3526#   3527    3534#   3535    3578    3583#   3590#   3596#   3597    3640    3645#   3652#   3658#   3659
                3702    3707#   3735    3740#   3743#   3749#   3750    3764#   3800    3825    3833#   3836#   3872    3878#   3879
                3894#   3917    3922#   3940    3959    3965#   3966    3986#   3994#   4009    4014#   4017#   4018    4020#   4033
                4038#   4041#   4042    4044#   4057    4062#   4065#   4066    4068#   4069    4088#   4096#   4108    4113#   4116#
                4117    4119#   4131    4136#   4139#   4140    4142#   4154    4159#   4162#   4163    4165#   4166    4185#   4193#
                4206    4212#   4215#   4216    4218#   4231    4237#   4240#   4241    4243#   4256    4261#   4264#   4265    4267#
                4268    4287#   4295#   4308    4313#   4316#   4317    4319#   4332    4337#   4340#   4341    4343#   4356    4361#
                4364#   4365    4367#   4368    4387#   4396#   4409    4414#   4417#   4418    4420#   4433    4438#   4441#   4442
                4444#   4457    4462#   4465#   4466    4468#   4469    4488#   4497#   4510    4515#   4518#   4519    4521#   4534
                4539#   4542#   4543    4545#   4558    4563#   4566#   4567    4569#   4570    4589#   4597#   4610    4615#   4618#
                4619    4621#   4634    4639#   4642#   4643    4645#   4658    4663#   4666#   4667    4669#   4670    4689#   4697#
                4710    4715#   4718#   4719    4721#   4734    4739#   4742#   4743    4745#   4758    4763#   4766#   4767    4769#
                4770    4789#   4796#   4810    4815#   4818#   4819    4821#   4836    4841#   4844#   4845    4847#   4860    4865#
                4868#   4869    4871#   4872    4891#   4898#   4912    4917#   4920#   4921    4923#   4938    4943#   4946#   4947
                4949#   4962    4967#   4970#   4971    4973#   4974    4993#   5000#   5013    5018#   5021#   5022    5024#   5039
                5044#   5047#   5048    5050#   5063    5068#   5071#   5072    5074#   5075    5094#   5101#   5114    5119#   5122#
                5123    5125#   5140    5145#   5148#   5149    5151#   5164    5169#   5172#   5173    5175#   5176    5196#   5203#
                5216    5221#   5224#   5225    5227#   5242    5247#   5250#   5251    5253#   5266    5271#   5274#   5275    5277#
                5278    5297#   5304#   5317    5322#   5325#   5326    5328#   5343    5348#   5351#   5352    5354#   5367    5372#
                5375#   5376    5378#   5379    5396#   5440    5449#   5456#   5457    5476#   5508    5516#   5525#   5526    5569#
                5626    5661#   5709    5716#   5717    5757#   5814    5849#   5858#   5859    5874#   5889    5895#   5908    5914#
                5925    5931#   5932    5948#   5963    5969#   5981    5987#   5998    6004#   6005    6035    6052    6068    6077#
                6078    6094#   6111    6133    6146    6154#   6163    6176    6187#   6188    6314    6319#   6345    6367    6372#
                6393    6398#   6404#   6416#   6417    6465    6472#   6473    6489#   6520    6527#   6533#   6541#   6542    6560#
                6610    6616#   6625    6632#   6640    6646#   6654    6660#   6670    6676#   6682    6687#   6694    6699#   6706
                6712#   6733#   6734    6769#   6788    6795#   6796    6819    6824#   6833    6841#   6842    6859#   6889    6896#
                6904#   6905    6925#   6947    6955#   6956    6994    7001#   7002    7022#   7060    7068#   7069    7089#   7090
                7113#   7115    7121#   7139    7145#   7162    7169#   7172#   7185    7195#   7196
MSTLAB             1#   1234#   2352#   2358#   2366#   2371#   2374#   2381#   2387#   2395#   2400#   2403#   2410#   2416#   2424#
                2429#   2432#   2439#   2445#   2452#   2457#   2460#   2467#   2473#   2480#   2485#   2488#   2495#   2501#   2509#
                2514#   2517#   2524#   2530#   2538#   2543#   2546#   2553#   2559#   2566#   2571#   2574#   2581#   2587#   2594#
                2599#   2602#   2609#   2615#   2622#   2627#   2630#   2637#   2643#   2650#   2655#   2658#   2665#   2671#   2678#
                2683#   2686#   2693#   2699#   2707#   2712#   2715#   2722#   2728#   2735#   2740#   2743#   2752#   2758#   2763#
                2766#   2773#   2779#   2786#   2791#   2794#   2801#   2807#   2814#   2819#   2822#   2830#   2834#   2843#   2849#
                2853#   2860#   2866#   2870#   2877#   2883#   2887#   2897#   2903#   2907#   2916#   2922#   2926#   2935#   2941#
                2947#   2951#   2960#   2966#   2970#   2980#   2986#   2990#   3011#   3042#   3048#   3054#   3060#   3077#   3162#
                3182#   3188#   3200#   3204#   3221#   3224#   3242#   3272#   3281#   3289#   3294#   3329#   3335#   3349#   3357#
                3370#   3379#   3388#   3393#   3397#   3407#   3420#   3429#   3439#   3444#   3448#   3459#   3473#   3481#   3495#
                3500#   3504#   3506#   3518#   3523#   3527#   3535#   3578#   3583#   3590#   3597#   3640#   3645#   3652#   3659#
                3702#   3707#   3735#   3740#   3743#   3750#   3764#   3800#   3825#   3833#   3836#   3872#   3879#   3894#   3917#
                3922#   3940#   3959#   3966#   3986#   3994#   4009#   4014#   4018#   4020#   4033#   4038#   4042#   4044#   4057#
                4062#   4066#   4069#   4088#   4096#   4108#   4113#   4117#   4119#   4131#   4136#   4140#   4142#   4154#   4159#
                4163#   4166#   4185#   4193#   4206#   4212#   4216#   4218#   4231#   4237#   4241#   4243#   4256#   4261#   4265#
                4268#   4287#   4295#   4308#   4313#   4317#   4319#   4332#   4337#   4341#   4343#   4356#   4361#   4365#   4368#
                4387#   4396#   4409#   4414#   4418#   4420#   4433#   4438#   4442#   4444#   4457#   4462#   4466#   4469#   4488#
                4497#   4510#   4515#   4519#   4521#   4534#   4539#   4543#   4545#   4558#   4563#   4567#   4570#   4589#   4597#
                4610#   4615#   4619#   4621#   4634#   4639#   4643#   4645#   4658#   4663#   4667#   4670#   4689#   4697#   4710#
                4715#   4719#   4721#   4734#   4739#   4743#   4745#   4758#   4763#   4767#   4770#   4789#   4796#   4810#   4815#
                4819#   4821#   4836#   4841#   4845#   4847#   4860#   4865#   4869#   4872#   4891#   4898#   4912#   4917#   4921#
                4923#   4938#   4943#   4947#   4949#   4962#   4967#   4971#   4974#   4993#   5000#   5013#   5018#   5022#   5024#
```

```
                5039#  5044#  5048#  5050#  5063#  5068#  5072#  5075#  5094#  5101#  5114#  5119#  5123#  5125#  5140#
                5145#  5149#  5151#  5164#  5169#  5173#  5176#  5196#  5203#  5216#  5221#  5225#  5227#  5242#  5247#
                5251#  5253#  5266#  5271#  5275#  5278#  5297#  5304#  5317#  5322#  5326#  5328#  5343#  5348#  5352#
                5354#  5367#  5372#  5376#  5379#  5396#  5440#  5449#  5457#  5476#  5508#  5516#  5526#  5569#  5626#
                5661#  5709#  5717#  5757#  5814#  5849#  5859#  5874#  5889#  5895#  5908#  5914#  5925#  5932#  5948#
                5963#  5969#  5981#  5987#  5998#  6005#  6035#  6052#  6068#  6078#  6094#  6111#  6133#  6146#  6154#
                6163#  6176#  6188#  6314#  6319#  6345#  6367#  6372#  6393#  6398#  6404#  6417#  6465#  6473#  6489#
                6520#  6527#  6533#  6542#  6560#  6610#  5616#  6625#  6632#  6640#  6646#  6654#  6660#  6670#  6676#
                6682#  6687#  6694#  6699#  6706#  6712#  6734#  6769#  6788#  6796#  6819#  6824#  6833#  6842#  6859#
                6889#  6896#  6905#  6925#  6947#  6956#  6994#  7002#  7022#  7060#  7069#  7090#  7113#  7115#  7121#
                7139#  7145#  7162#  7169#  7172#  7185#  7196#
MSTSTL            1#  1234#  2352#  2358#  2366#  2371#  2374#  2381#  2387#  2395#  2400#  2403#  2410#  2416#  2424#
                2429#  2432#  2439#  2445#  2452#  2457#  2460#  2467#  2473#  2480#  2485#  2488#  2495#  2501#  2509#
                2514#  2517#  2524#  2530#  2538#  2543#  2546#  2553#  2559#  2566#  2571#  2574#  2581#  2587#  2594#
                2599#  2602#  2609#  2615#  2622#  2627#  2630#  2637#  2643#  2650#  2655#  2658#  2665#  2671#  2678#
                2683#  2686#  2693#  2699#  2707#  2712#  2715#  2722#  2728#  2735#  2740#  2743#  2752#  2758#  2763#
                2766#  2773#  2779#  2786#  2791#  2794#  2801#  2807#  2814#  2819#  2822#  2830#  2834#  2843#  2849#
                2853#  2860#  2866#  2870#  2877#  2883#  2887#  2897#  2903#  2907#  2916#  2922#  2926#  2935#  2941#
                2947#  2951#  2960#  2966#  2970#  2980#  2986#  2990#  3011#  3042#  3048#  3054#  3060#  3077#  3162#
                3182#  3188#  3200#  3204#  3221#  3224#  3242#  3272#  3281#  3289#  3294#  3329#  3335#  3349#  3357#
                3370#  3379#  3388#  3393#  3397#  3407#  3420#  3429#  3439#  3444#  3448#  3459#  3473#  3481#  3495#
                3500#  3504#  3506#  3518#  3523#  3527#  3535#  3578#  3583#  3590#  3597#  3640#  3645#  3652#  3659#
                3702#  3707#  3735#  3740#  3743#  3750#  3764#  3800#  3825#  3833#  3836#  3872#  3879#  3894#  3917#
                3922#  3940#  3959#  3966#  3986#  3994#  4009#  4014#  4018#  4020#  4033#  4038#  4042#  4044#  4057#
                4062#  4066#  4069#  4088#  4096#  4108#  4113#  4117#  4119#  4131#  4136#  4140#  4142#  4154#  4159#
                4163#  4166#  4185#  4193#  4206#  4212#  4216#  4218#  4231#  4237#  4241#  4243#  4256#  4261#  4265#
                4268#  4287#  4295#  4308#  4313#  4317#  4319#  4332#  4337#  4341#  4343#  4356#  4361#  4365#  4368#
                4387#  4396#  4409#  4414#  4418#  4420#  4433#  4438#  4442#  4444#  4457#  4462#  4466#  4469#  4488#
                4497#  4510#  4515#  4519#  4521#  4534#  4539#  4543#  4545#  4558#  4563#  4567#  4570#  4589#  4597#
                4610#  4615#  4619#  4621#  4634#  4639#  4643#  4645#  4658#  4663#  4667#  4670#  4689#  4697#  4710#
                4715#  4719#  4721#  4734#  4739#  4743#  4745#  4758#  4763#  4767#  4770#  4789#  4796#  4810#  4815#
                4819#  4821#  4836#  4841#  4845#  4847#  4860#  4865#  4869#  4872#  4891#  4898#  4912#  4917#  4921#
                4923#  4938#  4943#  4947#  4949#  4962#  4967#  4971#  4974#  4993#  5000#  5013#  5018#  5022#  5024#
                5039#  5044#  5048#  5050#  5063#  5068#  5072#  5075#  5094#  5101#  5114#  5119#  5123#  5125#  5140#
                5145#  5149#  5151#  5164#  5169#  5173#  5176#  5196#  5203#  5216#  5221#  5225#  5227#  5242#  5247#
                5251#  5253#  5266#  5271#  5275#  5278#  5297#  5304#  5317#  5322#  5326#  5328#  5343#  5348#  5352#
                5354#  5367#  5372#  5376#  5379#  5396#  5440#  5449#  5457#  5476#  5508#  5516#  5526#  5569#  5626#
                5661#  5709#  5717#  5757#  5814#  5849#  5859#  5874#  5889#  5895#  5908#  5914#  5925#  5932#  5948#
                5963#  5969#  5981#  5987#  5998#  6005#  6035#  6052#  6068#  6078#  6094#  6111#  6133#  6146#  6154#
                6163#  6176#  6188#  6314#  6319#  6345#  6367#  6372#  6393#  6398#  6404#  6417#  6465#  6473#  6489#
                6520#  6527#  6533#  6542#  6560#  6610#  6616#  6625#  6632#  6640#  6646#  6654#  6660#  6670#  6676#
                6682#  6687#  6694#  6699#  6706#  6712#  6734#  6769#  6788#  6796#  6819#  6824#  6833#  6842#  6859#
                6889#  6896#  6905#  6925#  6947#  6956#  6994#  7002#  7022#  7060#  7069#  7090#  7113#  7115#  7121#
                7139#  7145#  7162#  7169#  7172#  7185#  7196#
MSWORD            1#  1234#  1305#  1314  1368#  1370   1371   1372   1373   1374   1375   1376   1377   1378   1379
                1380   1381   1382   1383   1384   1385   1386   1387   1388   1389   1390   1391   1392   1393   1394
                1395   1396   1397   1398   1399   1400   1401   1402   1403   1404   1405   1406   1407   1408   1409
                1410   1411   1412   3005#  3281#  3282   3283   3284   3329#  3330   3331   3332   3349#  3350   3351
                3352   3370#  3388#  3389   3390   3391   3420#  3439#  3440   3441   3442   3473#  3495#  3496   3497
                3498   3518#  3519   3520   3521   3578#  3579   3580   3581   3640#  3641   3642   3643   3702#  3703
                3704   3705   3735#  3736   3737   3738   3764#  3800#  3801   3802   3803   3825#  3826   3827   3828
                3872#  3873   3874   3875   3894#  3917#  3918   3919   3920   3940#  3941   3942   3943   3959#  3960
                3961   3962   3986#  4009#  4010   4011   4012   4033#  4034   4035   4036   4057#  4058   4059   4060
                4088#  4108#  4109   4110   4111   4131#  4132   4133   4134   4154#  4155   4156   4157   4185#  4206#
                4207   4208   4209   4231#  4232   4233   4234   4256#  4257   4258   4259   4287#  4308#  4309   4310
                4311   4332#  4333   4334   4335   4356#  4357   4358   4359   4387#  4409#  4410   4411   4412   4433#
```

```
                4434    4435    4436    4457#   4458    4459    4460    4488#   4510#   4511    4512    4513    4534#   4535    4536
                4537    4558#   4559    4560    4561    4589#   4610#   4611    4612    4613    4634#   4635    4636    4637    4658#
                4659    4660    4661    4689#   4710#   4711    4712    4713    4734#   4735    4736    4737    4758#   4759    4760
                4761    4789#   4810#   4811    4812    4813    4836#   4837    4838    4839    4860#   4861    4862    4863    4891#
                4912#   4913    4914    4915    4938#   4939    4940    4941    4962#   4963    4964    4965    4993#   5013#   5014
                5015    5016    5039#   5040    5041    5042    5063#   5064    5065    5066    5094#   5114#   5115    5116    5117
                5140#   5141    5142    5143    5164#   5165    5166    5167    5196#   5216#   5217    5218    5219    5242#   5243
                5244    5245    5266#   5267    5268    5269    5297#   5317#   5318    5319    5320    5343#   5344    5345    5346
                5367#   5368    5369    5370    5396#   5440#   5441    5442    5443    5476#   5508#   5509    5510    5511    5569#
                5626#   5627    5628    5629    5709#   5710    5711    5712    5757#   5814#   5815    5816    5817    5874#   5889#
                5890    5891    5892    5908#   5909    5910    5911    5925#   5926    5927    5928    5948#   5963#   5964    5965
                5966    5981#   5982    5983    5984    5998#   5999    6000    6001    6035#   6036    6037    6038    6052#   6053
                6054    6055    6068#   6069    6070    6071    6111#   6112    6113    6114    6133#   6134    6135    6136    6146#
                6147    6148    6149    6154#   6163#   6164    6165    6166    6176#   6177    6178    6179    6314#   6315    6316
                6317    6345#   6346    6347    6348    6367#   6368    6369    6370    6393#   6394    6395    6396    6404#   6465#
                6466    6467    6468    6489#   6520#   6521    6522    6523    6533#   6560#   6610#   6611    6612    6613    6625#
                6626    6627    6628    6640#   6641    6642    6643    6654#   6655    6656    6657    6670#   6671    6672    6673
                6682#   6683    6684    6685    6694#   6695    6696    6697    6706#   6707    6708    6709    6712#   6769#   6788#
                6789    6790    6791    6819#   6820    6821    6822    6833#   6834    6835    6836    6859#   6889#   6890    6891
                6892    6925#   6947#   6948    6949    6950    6994#   6995    6996    6997    7022#   7060#   7061    7062    7063
                7115#   7116    7117    7118    7139#   7140    7141    7142    7162#   7163    7164    7165    7185#   7186    7187
                7188    7215#   7221#   7232#   7340    7341
MSXFER             1#   1234#
OPEN               1#   1234#
POINTE             1#   1234#   1262
POPSP2          1740#   6127
PRINTB             1#   1234#   2348    2354    2360    2368    2377    2383    2389    2397    2406    2412    2418    2426    2435
                2441    2447    2454    2463    2469    2475    2482    2491    2497    2503    2511    2520    2526    2532    2540
                2549    2555    2561    2568    2577    2583    2589    2596    2605    2611    2617    2624    2633    2639    2645
                2652    2661    2667    2673    2680    2689    2695    2701    2709    2718    2724    2730    2737    2748    2754
                2760    2769    2775    2781    2788    2797    2803    2809    2816    2826    2845    2862    2879    2899    2918
                2937    2943    2962    2982
PRINTF             1#   1234#   2838    2856    2873    2891    2910    2929    2954    2974
PRINTS             1#   1234#
PRINTX             1#   1234#
READBU             1#   1234#
READEF             1#   1234#   3040    3046    3052    3058
RFLAGS             1#   1234#
ROMCLK          1813#   1888    1891    1894    1904    1912    1920    1928    1936    1939    1942    1953    1976    1982    3314
                3317    3320    3338    3341    3557    3560    3564    3567    3570    3619    3622    3626    3629    3632    3681
                3684    3688    3691    3694    3718    3721    3724    3727    3773    3777    3779    3786    3790    3808    3812
                3852    3855    3858    3861    3901    3905    3925    3931    3944    3947    3950    5404    5407    5412    5428
                5431    5487    5490    5499    5593    5596    5603    5606    5685    5688    5691    5695    5781    5784    5791
                5794    5880    5898    5917    5954    5972    5990    6040    6210    6224    6228    6232    6246    6250    6254
                6261    6264    6268    6272    6276    6285    6288    6304    6337    6355    6359    6378    6381    6385    6433
                6436    6439    6442    6447    6451    6865    6868    6876    6880    6932    6936    6980    6987    7035    7039
                7042    7052    7093    7097    7100    7104    7126    7130    7149    7152    7176
SETPRI             1#   1234#   7088
SETVEC             1#   1234#
SKIP04          1838#   3983
SKIP06          1824#   1968    3487    3844    4822    4924    5025    5126    5228    5329    5617    5672    5805    6486    6557
SKIP07          1831#   1970    1993    2015    6856    6922    7019
SLASH              1#   1234#
SROMCL          1819#   1897    1945    1956    3792    3804    3998    4001    4022    4025    4046    4049    4097    4100    4120
                4123    4143    4146    4195    4198    4220    4223    4245    4248    4297    4300    4321    4324    4345    4348
                4398    4401    4422    4425    4446    4449    4499    4502    4523    4526    4547    4550    4599    4602    4623
```

CZDMQD.P11 12-JAN-82 09:50 CROSS REFERENCE TABLE -- MACRO NAMES

```
          4626    4647    4650    4699    4702    4723    4726    4747    4750    4799    4802    4825    4828    4849    4852
          4901    4904    4927    4930    4951    4954    5002    5005    5028    5031    5052    5055    5103    5106    5129
          5132    5153    5156    5205    5208    5231    5234    5255    5258    5306    5309    5332    5335    5356    5359
          5493    5599    5787    6871
STARS        1#    1234#
SVC          1#    1232#   1233
XFER         1#    1234#   3005#   3370#   3420#   3473#   3764#   3894#   3986#   4088#   4185#   4287#   4387#   4488#   4589#
          4689#   4789#   4891#   4993#   5094#   5196#   5297#   5396#   5476#   5569#   5757#   5874#   5948#   6154#   6404#
          6489#   6533#   6560#   6712#   6769#   6859#   6925#   7022#
XFERF        1#    1234#
XFERT        1#    1234#
$MD       2330#   2346    2375    2404    2433    2461    2489    2518    2547    2575    2603    2631    2659    2687    2716
          2746    2767    2795
```

. ABS. 030144 000

ERRORS DETECTED: 0

CZDMQD,CZDMQD/SOL/CRF=SVC34R.MLB,CZDMQD.P11
RUN-TIME: 41 49 5 SECONDS
RUN-TIME RATIO: 159/96=1.6
CORE USED: 21K (41 PAGES)