

DMP - 11
DMV - 11

DMP/V - 11 FCTNL TST #1
CZDMTCO

AH - E238C - MC
FICHE 1 OF 2

AUG 1981
COPYRIGHT © 80-81
MADE IN USA



The main body of the document is a microfiche card containing a grid of approximately 20 columns and 20 rows of data. Each cell in the grid contains a small, high-contrast image of a document page, which is a scan of a microfiche frame. The data is organized into a structured table format, with columns likely representing different categories or fields of information. The text within each frame is too small to be legible, but the overall layout is consistent across the entire grid.

DMP-11
DMV-11

DMP/V-11 FCTNL TST #1
CZDMTCO

AH-E238C-MC
FICHE 2 OF 2

AUG 1981
COPYRIGHT © 80-81
MADE IN USA



SVC.MLB SOURCE FILE MACY11 30A(1052) 25-MAR-81 08:36 PAGE 2
CZDMTC.P11 25-MAR-81 08:24

.TITLE CZDMTCO DMP/V-11 FCTNL TST #1
.REM 8

IDENTIFICATION

PRODUCT CODE: AC-E237C-MC
PRODUCT NAME: CZDMTCO DMP/V-11 FUNCTIONAL TEST #1
PRODUCT DATE: AUGUST 1981
MAINTAINER: DIAGNOSTICS MERRIMACK CC: 38P

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1980, 1981 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

CZDMTC.P11 25-MAR-81 08:24

HISTORY

REV	DATE	REASON
---	---	-----
A	18-AUG-80	INITIAL RELEASE DMP ONLY
B	14-JAN-81	DMP11 BUGS FIXED
C	1-APR-81	DMV11 SUPPORT ADDED

CZDMTC.P11 25-MAR-81 08:24

TABLE OF CONTENTS

- 1.0 INTRODUCTION
- 2.0 HARDWARE REQUIREMENTS
- 3.0 PRELIMINARY PROGRAM REQUIREMENTS
- 4.0 GENERAL PROGRAM CONSIDERATIONS
 - 4.1 DIAGNOSTIC SUPERVISOR .
 - 4.2 EXECUTION TIME
- 5.0 PROGRAM LOAD MEDIA
- 6.0 OPERATING INSTRUCTIONS
 - 6.1 LOADING AND STARTING PROCEDURES
 - 6.1.1 LOADING PROCEDURES
 - 6.1.2 STARTING PROCEDURES
 - 6.1.3 STEPS FOR QUICK AND SIMPLE EXECUTION
 - 6.2 INITIAL DIALOGUE
 - 6.3 PROGRAM OPTIONS
 - 6.3.1 START COMMAND
 - 6.3.2 RESTART COMMAND
 - 6.3.3 CONTINUE COMMAND
 - 6.3.4 PROCEED COMMAND
 - 6.3.5 ADD COMMAND
 - 6.3.6 DROP COMMAND
 - 6.3.7 PRINT COMMAND
 - 6.3.8 DISPLAY COMMAND
 - 6.3.9 FLAGS COMMAND
 - 6.3.1 ZFLAGS COMMAND
 - 6.3.1 CONTROL CHARACTERS
 - 6.3.1 HARDWARE PARAMETERS
 - 6.3.1 SOFTWARE PARAMETERS
 - 6.3.1 EXTENDED DISCUSSION OF P-TABLE DIALOGUE
- 7.0 TEST DESCRIPTIONS
- 8.0 ERROR INFORMATION
 - 8.1 ERROR REPORTING

CZDMTC.P11 25-MAR-81 08:24

1.0 INTROP N

THE DMP AND DMV OPTIONS ARE COMMUNICATION OPTIONS THAT IMPLEMENT THE DDCMP PROTOCOL IN A MULTIDROP ENVIRONMENT. THE DMP IS USED WITH UNIBUS SYSTEMS WHILE THE DMV IS A Q BUS OPTION. THE PURPOSE OF THIS FUNCTIONAL TEST IS TO VERIFY AND EXERCISE THE MICROCODE USED IN THIS OPTION. THIS IS DONE BY PERFORMING THE FOLLOWING TESTS.

CSR ADDRESSING TESTS, ROM VERIFICATION BY CRC TESTS, RUNNING MICRO DIAGNOSTICS, RUNNING INTERFACE DIAGS. (DMP ONLY), CHECKS FOR RDO AND RDI, CHECKS FOR VARIOUS PROCEDURE ERRORS, MODE DEFINITION CHECKS, TEST FOR ALL CONTROL IN COMMANDS AND TESTS FOR ALL CONTROL AND INFORMATION OUT COMMANDS, TRANSMIT, AND RECEIVE MESSAGE TESTS OF VARIOUS LENGTHS, TO AND FROM VARIOUS BUFFERS.

THE FUNCTIONAL DIAGNOSTIC TEST WILL PROVIDE EXTENSIVE TROUBLESHOOTING CAPABILITIES, SUCH AS TIGHT SCOPE LOOPS, SWITCH OPTIONS, AND ABILITY TO 'LOCK' ONTO INTERMITTENT ERRORS. IN ADDITION TESTS WILL BE DESIGNED AND STRUCTURED TO ACHIEVE MAXIMUM FAULT RESOLUTION AND FACILITATE REPLACEMENT OF THE SMALLEST FIELD REPLACEABLE UNIT.

THIS PROGRAM WILL BE IMPLEMENTED USING THE DIAGNOSTIC SUPERVISOR AND A STRUCTURED PROGRAMMING APPROACH. BECAUSE THE DESIGN WILL CONFORM TO THE SUPERVISOR (STANDALONE VERSION) THE PROGRAM WILL BE COMPATIBLE WITH ACT, APT, XXDP+, AND SLIDE.

THROUGH DIALOGUE WITH OPERATOR, THE PROGRAM WILL ALLOW MODIFICATION OF DEVICE PARAMETERS, SUCH AS UNIBUS ADDRESS, VECTOR ADDRESSES AND DEVICE PRIORITY. IN ADDITION, THE OPERATOR CAN SPECIFY PARTICULAR TESTS TO BE RUN AND A VARIETY OF LOOPING, RUNNING, AND REPORTING MODES

DEVICE ERRORS WILL BE REPORTED AS THEY OCCUR. THE REPORT WILL INCLUDE A TEST NUMBER AND DESCRIPTION OF THE ERROR, GOOD AND BAD TEST DATA, AND APPLICABLE DEVICE REGISTER CONTENTS.

CZDMTC.P11 25-MAR-81 08:24

2.0 HARDWARE REQUIREMENTS

THE FOLLOWING HARDWARE IS REQUIRED TO RUN THE DMP/DMV-11 FUNCTIONAL TESTS:

FOR DMP:
PDP-11/04,05,10,20,30,34,35,40,45,50,60, OR 70
DMP-11

FOR DMV:
LSI-11/03,23,23B
DMV-11

FOR BOTH:
16K MEMORY
CONSOLE TERMINAL

3.0 PRELIMINARY PROGRAM REQUIREMENTS

FOR DMP:
THE M8207 STATIC DIAGNOSTICS AND THE M8203 STATIC DIAGNOSTICS SHOULD BE RUN BEFORE RUNNING THIS FUNCTIONAL DIAG.

FOR DMV:
THE M8053/64 MICROCONTROL AND LINE UNIT STATIC LOGIC TESTS (5 PROGRAMS) SHOULD BE RUN BEFORE RUNNING THIS FUNCTIONAL DIAG.

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 DMP-11 FUNCTIONAL TESTS IS ABOUT 120 SECONDS PER PASS (DMP-11) OR ??? SECONDS (DMV-11) FOR EACH UNIT.

4.3 XXDP+

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

CZDMTC.P11 25-MAR-81 08:24

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

IT IS USED IN TX AND RX TESTS.

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 WITHOUT READING THE REMAINDER OF THIS DOCUMENT, AS FOLLOWS:

- A) LOAD AND START THE DIAGNOSTIC USING THE RUN COMMAND
- B) RECEIVE DIAGNOSTIC SUPERVISOR IDENTIFICATION 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 THE PROGRAM IS STARTED, THE FOLLOWING IDENTIFICATION IS TYPED:

```

DRS LOADED
DIAG. RUN-TIME SERVICES
CZDMT-C-0
DMP/V-11 FUNCTIONAL DIAG.
UNIT IS DMP-11 OR DMV-11
DR>

```

THE OPERATOR THEN PROCEEDS BY TYPING ONE OR MORE OF THE COMMANDS DESCRIBED IN THE FOLLOWING SECTION 6.3. (FOR MORE 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.

CZDMTC.P11 25-MAR-81 08:24

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 OCCURRENCE 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.

CZDMTC.P11 25-MAR-81 08:24

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 DIALOGUE. 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>
*****

```

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>

CZDMTC.P11 25-MAR-81 08:24

CZDMT..P11 25-MAR-81 08:24

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 SUPPRESSED FOR THE REMAINDER OF THE DIAGNOSTIC OR UNTIL ANOTHER O IS TYPED, WHICH RESTORES NORMAL TELETYPE OUTPUT.

CZDMTC.P11 25-MAR-81 08:24

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.

SELECT OPTION TYPE (0=8207'DMP',1=8053'DMV',2=8064'DMV): (0) 0 ?
 DEVICE CSR ADDRESS : (0) 160170 ?
 DEVICE VECTOR ADDRESS : (0) 300 ?
 DEVICE PRIORITY LEVEL : (0) 5 ?
 TURNAROUND TYPE -(0=H3254H3255,1=CABLE,2=MOD LOC,3=MOD REM,4=NONE) (0) 0 ?
 PLEASE SELECT BAUD RATE: TYPE '0' FOR 2, 4K; '1' FOR 4 8K; '2' FOR 9.6K; '3' FOR 19.2K; '4' FOR 56K; '5' FOR 250K; OR '6' FOR 500K BAUDS (0) 4 ?
 SELECT INTERFACE TYPE (1=INTEGRAL,2=EIA,3=V.35,4=422): (0) 2 ?

6.3.13 SOFTWARE PARAMETERS

NO SOFTWARE PARAMETER QUESTIONS ARE ASKED BY THE DMP/V-11 FUNCTIONAL TEST

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.

CZDMTC.P11 25-MAR-81 08:24

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.

CZDMTC.P11 25-MAR-81 08:24

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 EXPLICIT 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).

CZDMTC.P11 25-MAR-81 08:24

TEST DESCRIPTIONS

7.0

7.1 ADDRESS TEST (TEST-1)

VERIFIES THAT ALL ADDRESSES IN THE MCPU RESPOND.
THIS TEST IS USED TO VERIFY THAT THE OPTION
IS AT THE ADDRESS THE USER THINKS IT IS ON.

7.2 ROM VERIFICATION TESTS (TESTS 2-9) (2-7 DMP ONLY+++8-9 DMV ONLY)

THIS SERIES OF TESTS VERIFIES THAT ALL ROMS
ARE IN PLACE AND THAT THE THE CONTENTS ARE
CORRECT BY DOING A CRC CALCULATION ON THE
ROM CONTENTS. THE TEST ALSO PRINTS THE REV AND
ROM NUMBER OF THE ROM ON THE FIRST PASS OF THE
TEST.

7.3 INITIALIZATION TEST (TEST 10)

THIS TEST DOES A MASTER CLEAR TO THE DEVICE
AND WAITS FOR THE MICRO-DIAGNOSTICS TO COMPLETE
IF MICRO DIAGS FAIL TO COMPLETE THEN A TIME
OUT ERROR WILL BE REPORTED.

7.4 INTERFACE DIAGNOSTICS (TEST 11) (DMP ONLY)

THIS TEST RUNS ADDITIONAL MICRODIAGNOSTIC CODE IN THE
DMP THAT CHECKS OUT THE INTERRUPT LOGIC AND THE
NPR LOGIC.

7.5 RDI REMAINS SET TEST (TEST 12)

THIS TEST SETS RQI, WAITS FOR RDI TO SET, ISSUES
A 'NO REQUEST' CONTROL IN AND LOOKS FOR RDI TO
REMAIN SET.

7.6 TEST FOR RDO SETTING (TEST 13)

THIS TEST DOES A CONTROL IN COMMAND OF 'READ MODEM'
AND EXPECTS RDO TO SET WITH AN INFORMATION OUT CODE
OF RETURN MODEM STATUS.

7.7 CHECK FOR PROCEDURE ERROR 100 (TEST 14)

THIS TEST ISSUES A MASTER CLEAR WAITS FOR RUN TO
SET THEN ISSUES A CONTROL IN COMMAND AND EXPECTS
A PROCEDURE ERROR OF 100 'NON MODE DEFINITION COMMAND
AFTER A MASTER CLEAR'. UNLESS MODE HAS BEEN DEFINED
IN THE SWITCHES THEN LOOK FOR INFORMATION OUT.

7.8 CHECK FOR PROCEDURE ERROR 104 (TEST 15)

CZDMTC.P11 25-MAR-81 08:24

THIS TEST ISSUES A MASTER CLEAR , MODE DEFINITION, FOLLOWED BY A MODE DEF. COMMAND DEFINING A DIFFERENT TYPE OF MODE. THE TEST LOOKS FOR A PROCEDURE ERROR OF 104 'ILLEGAL MODE CHANGE'.

7.9 TEST MODE CHANGE OF DUPLEX PORTION OF MODE (TEST 16)

THIS TEST ISSUES A MASTER CLEAR,MODE DEFINITION SEQUENCE(CONTROL STATION/FULL DUPLEX). THE TEST THEN ISSUES A MODE DEF. COMMAND TO CHANGE TO HALF DUPLEX. THEN THE TESTS WAITS AND MAKES SURE NO PROCEDURE ERROR OCCURS.

7.10 TEST FOR MAX TRIBS TO BE ESTABLISHED. (TEST 17)

THIS TEST ESTABLISHES MAX TRIBS THEN ATTEMPTS TO ESTABLISH MAX+1 TRIBS AND CHECKS FOR A PROCEDURE ERROR 114, 'ATTEMPT TO ESTABLISH MORE THEN MAXIMUM NUMBER OF TRIBS'. THE TEST THEN TRIES TO ESTABLISH A TRIB THAT HAS ALREADY BEEN ESTABLISHED AND CHECKS FOR A PROCEDURE ERROR OF 116 'ATTEMPT TO ESTABLISH ALREADY ESTABLISHED TRIB'.

NOTE: MAX TRIBS FOR DMP = 32
MAX TRIBS FOR DMV = 12

7.11 READ/WRITE TRIBUTARY STATUS SLOTS TEST (TEST 18)

THIS TEST WRITES EACH TSS SLOT WITH VARIOUS DATA PATTERNS THEN READS THAT SLOT TO BE SURE THAT THE CORRECT OUTPUT COMMAND AND DATA IS RETURNED. THE SLOTS THAT ARE WRITTEN ARE TRIB STATUS SLOTS 30 THRU 37. THE DATA PATTERNS USED ARE:0,125252,052525,0,-1,377,177400,562:OCTAL.

7.12 TESTS FOR PROCEDURE ERROR 132 (TEST 19-20)

THESE TESTS CHECK THAT A PROCEDURE ERROR OF 132 'ATTEMPT TO WRITE INTO A RESERVED AREA OF THE TRIBUTARY STATUS SLOTS' IS PRODUCED WHEN A WRITE TSS COMMAND IS ISSUED FOR ADDRESS 4. A READ/CLEAR TSS COMMAND IS ISSUED FOR ADDRESS 6.

7.13 TEST FOR READ/CLEAR COMMAND (TEST 21)

THIS TEST ISSUES A READ CLEAR COMMAND TO TRIBUTARY STATUS SLOT 7 AND MAKES SURE THAT NO ERRORS OCCUR.

7.14 TESTS FOR GLOBAL STATUS SLOTS (TEST 22)

THIS TEST READS ALL THE GLOBAL STATUS SLOTS THEN WRITES ALL THE GLOBAL SLOTS USING THE ADDRESSES AS DATA THEN READS THEM BACK AND MAKES SURE THE DATA IS CORRECT. THIS TEST ALSO CHECKS FOR THE LIMITS ON THE WRITE TSS COMMAND BY MAKING SURE A

CZDMTC.P11 25-MAR-81 08:24

PROCEDURE ERROR OCCURES WHEN THE LIMITS ARE EXCEEDED. THIS TEST ALSO CHECKS THE READ/CLEAR COMMAND TO A GLOBAL STATUS SLOT.

7.15 HALT TRIB COMMAND TESTS (TEST 23)

THIS TEST CHECKS THE HALT TRIB COMMAND BY DOING THE FOLLOWING: MASTER CLEAR;MODE DEF;ESTABLISH TRIB;ISTRIB;QUE UP REC BUFFER; ISSUE HALT TRIB COMMAND;CHECK FOR OUTPUT OF REC BUFFER UNUSED;CHECK FOR SECOND OUTPUT OF BUFFER RETURNED COMPLETE. THE TEST THEN ISSUES A SECOND HALT TRIB COMMAND AND CHECKS THAT AFTER A DELAY NO CONTROL OUT OCCURS

7.16 KILL TRIB COMMAND TESTS (TEST 24)

THIS TEST CHECKS THE KILL TRIB COMMAND BY DOING THE FOLLOWING: MASTER CLEAR; MODE DEF.; ESTABLISH TRIB; READ TSS SLOT 1 AND COMPARE FOR GOOD ADDRESS; PUT TRIB IN MAINT STATE;ISSUE KILL TRIB; CHECK FOR PROCEDURE ERROR 112 'KILL TO UNHALTED TRIB'; HALT TRIB; KILL TRIB;READ TSS SLOT 1 AND CHECK FOR PROCEDURE ERROR 106 'NON GLOBAL CONTROL IN COMMAND TO UNESTABLISHED TP'IB'.

7.17 CHECK FOR PROCEDURE ERROR 102 (TEST 25)

THIS TEST ISSUES ILLEGAL TYPE CODES OF 7 6 5 AND 3 AND CHECKS THAT EACH ONE PRODUCES A PROCEDURE ERROR 102 'ILLEGAL TYPE CODE USED IN AN INPUT COMMAND'.

7.18 CHECK FOR PROCEDURE ERROR OF 110 (TEST 26)

THIS TEST ISSUES A MASTER CLEAR; MODE DEF; FOLLOWED BY AN ISTRIB TO TRIB ADDRESS OF ZERO. IT THEN CHECKS FOR A 'PROCEDURE ERROR OF 110 'ATTEMPT TO PERFORM A NON-GLOBAL COMMAND FOR TRIBUTARY ADDRESS OF 0'.

7.19 CHECK FOR PROCEDURE ERROR OF 120 (TEST 27)

THIS TEST ISSUES A CONTROL IN WITH A REQUEST KEY OF 7 AND ALSO A CONTROL IN WITH A REQUEST KEY OF 17 THEN IT CHECKS THAT BOTH CASES GIVE PROCEDURE ERROR 120 'ILLEGAL REQUEST KEY ON CONTROL IN.'

7.20 CHECK FOR PROCEDURE ERROR OF 134 (TEST 28)

THIS TEST ISSUES A MASTER CLEAR , MODE DEF, AND ESTABLISH TRIB SEQUENCE, FOLLOWED BY AN ATTEMPT TO USE A RESERVED BIT IN BSEL 7 THEN CHECKS THAT THIS PRODUCES A PROCEDURE ERROR OF 134 'ATTEMPT TO USE RESERVED BIT IN BS'. 7 ON CONTROL IN ''

7.21 LATCH/UNLATCH POLL CHECK (TEST 29)

THIS TEST CHECKS THE LATCH AND UNLATCH POLL COMMANDS BY DJING THE FOLLOWING SEQUENCE OF COMMANDS:

CZDMTC.P11 25-MAR-81 08:24

MASTER CLEAR; MODE DEF; ESTABLISH TRIB; LATCH POLL TO DEAD STATE; READ TSS SLOT 2 AND CHECK THAT DEAD BIT IS ON; UNLATCH POLL; READ TSS SLOT 2; CHECK THAT ACTIVE BIT IS ON.

7.22 SHORT MESSAGE SENDING TEST (TEST 30)

THIS TEST SENDS A 4 BYTE MESSAGE FROM AN EVEN TRANSMIT BUFFER TO AN EVEN REC BUFFER IN DDCMP FORMAT CONFIGURED AS A MULTIPOINT CONTROL STATION FULL DUPLEX. THE TEST CHECKS THAT REC BUFFERS ARE RETURNED AND DATA IS CORRECT AND THAT THE NEXT OUTPUT IS TRANSMIT BUFFER RETURNED. THIS TEST IS ALWAYS DONE IN TTL LOOPBACK MODE.

7.23 CHECK FOR PROCEDURE ERROR 122 (TEST 31)

THIS TEST CHECKS FOR A PROCEDURE ERROR OF 122 BY PERFORMING THE FOLLOWING: MASTER CLEAR ; MODE DEF; ESTABLISH BUFFER; CHECK FOR ERROR 122 'ATTEMPT TO ASSIGN A BUFFER FOR AN UNESTABLISHED TRIB'.

7.24 CHECK FOR PROCEDURE ERROR 124 (TEST 32)

THIS TEST CHECKS FOR A PROCEDURE ERROR OF 124 BY PERFORMING THE FOLLOWING: MASTER CLEAR; MODE DEF; ESTABLISH TRIB; ESTABLISH BUFFER; CHECK FOR ERROR 124 'ATTEMPT TO ASSIGN A BUFFER FOR A HALTED TRIB'.

7.25 CHECK FOR PROCEDURE ERROR 126 (TEST 33)

THIS TEST CHECKS FOR A PROCEDURE ERROR OF 126 BY PERFORMING THE FOLLOWING: MASTER CLEAR; MODE DEF ESTABLISH TRIB ; PUT TRIB IN MAINT STATE; ESTABLISH BUFFER WITH 0 BYTE COUNT; LOOK FOR ERROR 126 'ATTEMPT TO ASSIGN A BUFFER WITH A BYTE COUNT OF 0'.

7.26 CHECK FOR PROCEDURE ERROR 130 (TEST 34)

THIS TEST CHECKS FOR A PROCEDURE ERROR OF 130 BY PERFORMING THE FOLLOWING: MASTER CLEAR; MODE DEF; ESTABLISH TX BUFFER TO TRIB; CHECK FOR ERROR 130 'ATTEMPT TO ASSIGN A TRANSMIT BUFFER FOR TRIB 0'.

7.27 TRANSMIT/RECEIVE 256 BYTES, PTP, DDCMP (TEST 35)

THIS TEST WILL TRANSMIT A BUFFER OF 256 BYTES, STARTING ON AN EVEN BYTE BOUNDARY TO A REC BUFFER STARTING ON AN EVEN BYTE BOUNDARY. THE MODE DEFINED IS POINT TO POINT FULL DUPLEX, DDCMP PROTOCOL. IF THERE IS EXTERNAL LOOP BACK THEN THE TEST WILL BE DONE OVER THAT LOOPBACK; ELSE THE LOOPBACK WILL BE SET TO INTERNAL (TTL).

7.28 DMV Q22 MODE TX/RX 256 BYTES, DDCMP (TEST 36) * DMV ONLY *

THIS TEST WILL TRANSMIT A BUFFER OF 256 BYTES, STARTING

CZDMTC.P11 25-MAR-81 08:24

ON AN EVEN BYTE BOUNDARY TO A REC BUFFER STARTING ON AN EVEN BOUNDARY. THE MODE DEFINED IS Q22 FORMAT, POINT TO POINT FULL DUPLEX,DDCMP PROTOCOL. IF THERE IS EXTERNAL LOOPBACK THEN THE TEST WILL USE IT, OTHERWISE THE LOOPBACK WILL BE SET TO INTERNAL(TTL).

7.29 TRANSMIT/RECEIVE 255 BYTES,MTP,DDCMP (TEST 37)

THIS TEST WILL TRANSMIT A BUFFER OF 255 BYTES STARTING ON AN EVEN BYTE ADDRESS TO A REC BUFFER STARTING ON AN ODD BYTE ADDRESS. THE MODE IS FULL DUPLEX CONTROL STATION MULTIPOINT,DDCMP PROTOCOL. THE DATA IS COMPARED FOR CORRECTNESS. THE TEST IS DONE WITH INTERNAL LOOPBACK.

7.30 READ/WRITE MODEM REGISTER TESTS (TEST 38) (DMP ONLY)

THIS TEST WRITES THE MODEM REGS OVER THE VARIOUS INTERFACES WITH A PATTERN OF 100. THE MODEM REGS ARE THEN READ AND COMPARED FOR CORRECTNESS.

 * N O T E ----- THIS TEST ONLY RUNS IF LOOPBACK CONNECTORS
 ***** ARE ATTACHED

7.31 TEST OF MEM EXTENSION BITS. (TESTS 39-41)

THESE THREE TESTS CHECK THE ABILITY OF THE DEVICE TO DO TRANSFERS TO UPPER MEMORY (IF IT EXISTS). THE TRANSFERS ARE DONE BY TRANSMITTING AND RECEIVING A MESSAGE. (TTL LOOPBACK ONLY). THE THREE TESTS ARE DONE FOR BIT 16, BIT 17 AND BITS 16 AND 17 (DMV SET FOR Q18 MODE).

 * N O T E ----- THIS TEST USES MEMORY ONLY IF IT EXISTS

7.32 TEST FOR TX/RX 257 BYTES (TEST 42)

THIS TEST TRANSMITS A MESSAGE OF 257 BYTES FROM A TRANSMIT BUFFER STARTING WITH AN ODD BYTE BOUNDARY TO A RECEIVE BUFFER STARTING ON AN ODD BYTE BOUNDARY IN DDCMP MODE,POINT TO POINT. IF THERE IS EXTERNAL LOOPBACK THEN THE TEST WILL BE DONE OVER THAT LOOPBACK;ELSE THE LOOPBACK WILL BE SET TO INTERNAL(TTL).

7.33 TEST FOR TX/RX 1 BYTE (TEST 43)

THIS TEST TX'S AND REC'S A 1 BYTE MESSAGE FROM AN ODD TX BUFFER TO AN EVEN RX BUFFER IN MAINT MODE,MULTIPOINT CONTROL STATION.

7.34 POLLING STATE TESTS (TEST 44)

THIS TEST CHECKS THE DEGRADING OF THE POLLING STATES FROM ACTIVE TO INACTIVE TO POTENTIALLY DEAD TO DEAD THE SEQUENCE THAT IS EXECUTED IS AS FOLLOWS:

CZDMTC.P11

25-MAR-81 08:24

MASTER CLEAR, MODE DEF(FULL DUP CONTROL STATION),
SET POLL DELAY(GSS ADD 37), ESTABLISH TRIB, SET SELECTION
TIMER(TSS ADD 36), SET NUMBER OF NO DATA MESSAGES TO
INACTIVE TO 10 AND THE NUMBER OF TIME OUTS TO POTENTIALLY
DEAD TO 4, ISTRT TRIB, WAIT FOR RUN STATE, READ TSS (ADD 2),
CHECK FOR INACTIVE BIT, LOOP UNTIL INACTIVE OR TIME OUT, READ
THE SELECTION TIMER(TSS 11), COMPARE IT TO 10, CHANGE MODE
TO HALF DUPLEX, WAIT FOR TSS SLOT 2 TO INDICATE POT. DEAD,
READ SELECTION TIMER(TSS 16), COMPARE IT TO 4, WAIT FOR
CONTROL OUT INDICATING DEAD TRIB, READ SELECTION TIMER(TSS 16)
COMPARE IT TO 10.

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:

CZDMT DVC FTL ERR 00024 ON UNIT 00 TST 004 SUB 000 PC: 016170
ERROR IN ROM E04 READ = 177777 ; CALCULATED = 017327

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

&

CZDMTC.P11 25-MAR-81 08:24

```

1092
1093      002000      ;      .=2000
1094
1095
1096
1097      000200      DRUN== 200
1098
1099
1100
1101
1102
1103

```

```

1104      000001
1105      000001
1106      000000
1107      000000
1108      000000
1109      000000
1110      000000
1111
1112
1113
1114
1115
1116
1117

```

```

$LSTIN= 1
$LSTTAG= 1
SVCINS= 0      ; LIST INSTRUCTIONS, SHIFTED RIGHT
SVCTST= 0      ; LIST TEST TAGS, SHIFTED RIGHT
SVCSUB= 0      ; LIST SUBTEST TAGS, SHIFTED RIGHT
SVCGBL= 0      ; LIST GLOBAL TAGS, SHIFTED RIGHT
SVCTAG= 0      ; LIST OTHER TAGS, SHIFTED RIGHT

:      CHANGE THE VALUES OF THE SVC... SYMBOLS TO BE ZERO IF YOU WISH
:      TO ALIGN THE MACRO CALLS AND THEIR EXPANSIONS.  CHANGE THE
:      SYMBOLS TO BE MINUS-ONE TO NOT LIST THE EXPANSIONS.  YOU MAY
:      CHANGE THE SYMBOLS AT ANY POINT IN YOUR PROGRAM.

```

CZDMTC.P11 25-MAR-81 08:24

PROGRAM HEADER

```

1118
1119
1120
1121
1122
1123
1124
1125
1126 002000
1127 002000 103
1128 002001 132
1129 002002 104
1130 002003 115
1131 002004 124
1132 002005 000
1133 002006 000
1134 002007 000
1135 002010
1136 002010 103
1137 002011
1138 002011 060
1139 002012
1140 002012 000000
1141 002014
1142 002014 003410
1143 002016
1144 002016 032534
1145 002020
1146 002020 000000
1147 002022
1148 002022 002264
1149 002024
1150 002024 000000
1151 002026
1152 002026 037660
1153 002030
1154 002030 000000
1155 002032
1156 002032 000000
1157 002034
1158 002034 000000
1159 002036
1160 002036 000000
1161 002040
1162 002040 002132
1163 002042
1164 002042 000340
1165 002044
1166 002044 000000
1167 002046
1168 002046 000000
1169 002050
1170 002050 003
1171 002051 003
1172 002052
1173 002052 000000

```

```

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

L$NAME::          ;DIAGNOSTIC NAME
                  .ASCII /C/
                  .ASCII /Z/
                  .ASCII /D/
                  .ASCII /M/
                  .ASCII /T/
                  .BYTE 0
                  .BYTE 0
                  .BYTE 0

L$REV::           ;REVISION LEVEL
                  .ASCII /C/

L$DEPO::          ;0
                  .ASCII /O/

L$UNIT::          ;NUMBER OF UNITS
                  .WORD 0

L$TIML::          ;LONGEST TEST TIME
                  .WORD 1800.

L$HPCP::          ;POINTER TO H.W. QUES.
                  .WORD L$HARD

L$SPCP::          ;POINTER TO S.W. QUES.
                  .WORD 0

L$HPTP::          ;PTR. TO DEF. H.W. PTABLE
                  .WORD L$HW

L$SPTP::          ;PTR. TO S.W. PTABLE
                  .WORD 0

L$LADP::          ;DIAG. END ADDRESS
                  .WORD L$LAST

L$STA::           ;RESERVED FOR APT STATS
                  .WORD 0

L$CO::            ;DIAGNOSTIC TYPE
                  .WORD 0

L$DTYP::          ;APT EXPANSION
                  .WORD 0

L$APT::           ;PTR. TO DISPATCH TABLE
                  .WORD 0

L$DTP::           ;DIAGNOSTIC RUN PRIORITY
                  .WORD L$DISPATCH

L$PRIO::          ;FLAGS DESCRIBE HOW IT WAS SETUP
                  .WORD #PRI07

L$ENVI::          ;EXPANSION WORD
                  .WORD 0

L$EXP1::          ;SVC REV AND EDIT #
                  .WORD 0

L$MREV::          ;DIAG. EVENT FLAGS
                  .BYTE C$REVISION
                  .BYTE C$EDIT
                  .WORD 0

L$EF::            ;DIAG. EVENT FLAGS
                  .WORD 0

```

CZDMTC.P11 25-MAR-81 08:24

PROGRAM HEADER

1174	002054	000000	L\$SPC::	.WORD	0	
1175	002056		L\$DEVP::	.WORD	0	
1176	002056	000000				; POINTER TO DEVICE TYPE LIST
1177	002060		L\$REPP::	.WORD	L\$DVTYP	
1178	002060	002564				;PTR. TO REPORT CODE
1179	002062		L\$EXP4::	.WORD	0	
1180	002062	000000	L\$EXP5::	.WORD	0	
1181	002064		L\$AUT::	.WORD	0	;PTR. TO ADD UNIT CODE
1182	002064	000000	L\$DUT::	.WORD	L\$AU	;PTR. TO DROP UNIT CODE
1183	002066		L\$SLUN::	.WORD	L\$DU	;LUN FOR EXERCISERS TO FILL
1184	002066	000000	L\$DESP::	.WORD	0	;POINTER TO DIAG. DESCRIPTION
1185	002070		L\$LOAD::	.WORD	L\$DESC	;GENERATE SPECIAL AUTOLOAD EMT
1186	002070	014502	L\$SETP::	EMT	E\$LOAD	;PTR. TO ERRtbl
1187	002072		L\$ICP::	.WORD	0	;PTR. TO INIT CODE
1188	002072	014476	L\$CCP::	.WORD	L\$INIT	;PTR. TO CLEAN-UP CODE
1189	002074		L\$ACP::	.WORD	L\$CLEAN	;PTR. TO AUTO CODE
1190	002074	000000	L\$PRT::	.WORD	L\$AUTO	;PTR. TO PROTECT TABLE
1191	002076		L\$TEST::	.WORD	L\$PROT	;TEST NUMBER
1192	002076	002606	L\$DLY::	.WORD	0	;DELAY COUNT
1193	002100		L\$HIME::	.WORD	0	;PTR. TO HIGH MEM
1194	002100	104035				
1195	002102		L\$PROT::	.WORD	-1	
1196	002102	000000				
1197	002104					
1198	002104	013774				
1199	002106					
1200	002106	014472				
1201	002110					
1202	002110	014430				
1203	002112					
1204	002112	002122				
1205	002114					
1206	002114	000000				
1207	002116					
1208	002116	000000				
1209	002120					
1210	002120	000000				
1211						
1212	002122					
1213	002122	177777				
1214	002124	177777				
1215	002126	177777				
1216						

CZDMTC.P11 25-MAR-81 08:24

DISPATCH TABLE

.SBTTL DISPATCH TABLE

```

:////////////////////
:/ THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
:/ IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
:////////////////////

```

```

1217
1218
1219
1220
1221
1222
1223
1224 002130 000054
1225 002132
1226 002132 014504
1227 002134 014664
1228 002136 015216
1229 002140 015550
1230 002142 016102
1231 002144 016434
1232 002146 016766
1233 002150 017320
1234 002152 017460
1235 002154 017620
1236 002156 020004
1237 002160 021136
1238 002162 021250
1239 002164 021460
1240 002166 021724
1241 002170 022134
1242 002172 022366
1243 002174 022620
1244 002176 023104
1245 002200 023212
1246 002202 023320
1247 002204 023434
1248 002206 024162
1249 002210 024610
1250 002212 025236
1251 002214 025352
1252 002216 025436
1253 002220 025664
1254 002222 026112
1255 002224 026440
1256 002226 026534
1257 002230 026654
1258 002232 027022
1259 002234 027210
1260 002236 027326
1261 002240 027450
1262 002242 027612
1263 002244 027706
1264 002246 030334
1265 002250 030576
1266 002252 031040
1267 002254 031302
1268 002256 031420
1269 002260 031516
1270
1271
1272

```

```

        .WORD 44
L$DISPATCH:
        .WORD T1
        .WORD T2
        .WORD T3
        .WORD T4
        .WORD T5
        .WORD T6
        .WORD T7
        .WORD T8
        .WORD T9
        .WORD T10
        .WORD T11
        .WORD T12
        .WORD T13
        .WORD T14
        .WORD T15
        .WORD T16
        .WORD T17
        .WORD T18
        .WORD T19
        .WORD T20
        .WORD T21
        .WORD T22
        .WORD T23
        .WORD T24
        .WORD T25
        .WORD T26
        .WORD T27
        .WORD T28
        .WORD T29
        .WORD T30
        .WORD T31
        .WORD T32
        .WORD T33
        .WORD T34
        .WORD T35
        .WORD T36
        .WORD T37
        .WORD T38
        .WORD T39
        .WORD T40
        .WORD T41
        .WORD T42
        .WORD T43
        .WORD T44

```

CZDMTC.P11

25-MAR-81 08:24

DISPATCH TABLE

1273
1274
1275

CZDMTC.P11 25-MAR-81 08:24

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.
:////////////////////

```

```

1276
1277
1278
1279
1280
1281
1282
1283
1284 002262 000014
1285 002264
1286 002264
1287
1288 002264 000000
1289 002266 160170
1290 002270 000300
1291 002272 005000
1292 002274 000003
1293 002276 000056
1294 002300 000000
1295 002302 000000
1296 002304 000004
1297
1298
1299 002306 000004
1300
1301
1302 002310 000000
1303 002312 000002
1304
1305 002314
1306
1307
1308
1309
1310

```

```

        .WORD  L10001-L$HW/2
L$HW::
DFPTBL::

```

```

        .WORD  0                :HARDWARE TYPE
        .WORD  160170           :M8200,4,7 CSR UNIBUS ADDRESS
        .WORD   300             :M8200,4,7 INTERRUPT VECTOR
        .WORD  5000             :M8200,4,7 INTERRUPT PRIORITY LEVEL = 5
        .WORD   3               :LINE UNIT = M8203
        .WORD   056             :SWITCH PACK #1 (REG 11)
        .WORD   000             :SWITCH PACK #2 (REG 15)
        .WORD   000             :SWITCH PACK #3 (REG 16)
        .WORD    4              :H3251&H3252 USED
                                :0= LOOPBACK CABLE,2= TEST CONNECTOR
                                :4= NONE
                                :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
                                :DUMMY WORD FOR RUN
                                :1=INTEGRAL ;2=EIA;3=V.35;4=422

```

L10001:

CZDMTC.P11 25-MAR-81 08:24

SOFTWARE P-TABLE

.SBTTL SOFTWARE P-TABLE

1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329

002314 000000
002316
002316
002316

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

.WORD L10002-LSSW/2
LSSW::
SFPTBL::

L10002:

CZDMTC.P11 25-MAR-81 08:24

SOFTWARE P-TABLE

1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385

100000
040000
020000
010000
004000
002000
001000
000400
000200
000100
000040
000020
000010
000004
000002
000001

001000
000400
000200
000100
000040
000020
000010
000004
000002
000001

000040
000037
000036
000035
000034

000340
000300
000240

.SBTTL GLOBAL EQUATES SECTION

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

: BIT DIFINITIONS

BIT15== 100000
BIT14== 40000
BIT13== 20000
BIT12== 10000
BIT11== 4000
BIT10== 2000
BIT09== 1000
BIT08== 400
BIT07== 200
BIT06== 100
BIT05== 40
BIT04== 20
BIT03== 10
BIT02== 4
BIT01== 2
BIT00== 1

BIT9== BIT09
BIT8== BIT08
BIT7== BIT07
BIT6== BIT06
BIT5== BIT05
BIT4== BIT04
BIT3== BIT03
BIT2== BIT02
BIT1== BIT01
BIT0== BIT00

: EVENT FLAG DEFINITIONS
: EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION

EF.START== 32. : START COMMAND WAS ISSUED
EF.RESTART== 31. : RESTART COMMAND WAS ISSUED
EF.CONTINUE== 30. : CONTINUE COMMAND WAS ISSUED
EF.NEW== 29. : A NEW PASS HAS BEEN STARTED
EF.PWR== 28. : A POWER-FAIL/POWER-UP OCCURRED

: PRIORITY LEVEL DEFINITIONS

PRI07== 340
PRI06== 300
PRI05== 240

CZDMTC.P11

25-MAR-81 08:24

GLOBAL EQUATES SECTION

1386 000200
 1387 000140
 1388 000100
 1389 000040
 1390 000000
 1391
 1392
 1393
 1394 000004
 1395 000010
 1396 000020
 1397 000040
 1398 000100
 1399 000200
 1400 000400
 1401 001000
 1402 002000
 1403 004000
 1404 010000
 1405 020000
 1406 040000
 1407 100000
 1408
 1409
 1410
 1411
 1412
 1413
 1414
 1415
 1416
 1417 000200
 1418 000100
 1419 000020
 1420 000010
 1421 000004
 1422 000002
 1423 000001
 1424
 1425
 1426
 1427
 1428 000010
 1429 000200
 1430 000020
 1431 000200
 1432
 1433

PRI04== 200
 PRI03== 140
 PRI02== 100
 PRI01== 40
 PRI00== 0
 ;
 ;OPERATOR FLAG BITS
 ;
 EVL== 4
 LOT== 10
 ADR== 20
 IDU== 40
 ISR== 100
 UAM== 200
 BOE== 400
 PNT== 1000
 PRI== 2000
 IXE== 4000
 IBE== 10000
 IER== 20000
 LOE== 40000
 HOE== 100000

 ; * PROGRAM EVENT FLAG DEFINITIONS
 ; *****

 ; * MAINTENANCE REGISTER - BSEL1
 ; *****

RUN = BIT7
 MCLR = BIT6
 STEPLU = BIT4
 LULoop = BIT3
 ROMO = BIT2
 ROMI = BIT1
 STEPMP = BIT0

 ; OTHER BIT DEFINITIONS
 ; *****

Q22BIT =BIT3
 RQI =200
 RDI =020
 RDO =200

CZDMTC.P11 25-MAR-81 08:24

GLOBAL DATA SECTION

1434
 1435
 1436
 1437
 1438
 1439
 1440
 1441
 1442
 1443
 1444 002316 000000
 1445 002320 000000
 1446 002322 000000
 1447 002324 000000
 1448 002326 000000
 1449 002330 000000
 1450 002332 000000
 1451 002334 000000
 1452 002336 000000
 1453 002340 000000
 1454 002342 000000
 1455 002344 000000
 1456 002346 000000
 1457 002350 000000
 1458 002352 000000
 1459 002354 000000
 1460 002356 000000
 1461 002360 000000
 1462 002362 000040
 1463 002364 000000
 1464
 1465 002366 000000
 1466 002370 000000
 1467 002372 000000
 1468 002374 177777
 1469 002376 000000
 1470 002400 000000
 1471 002402 000000
 1472 002404 000000
 1473 002406 000000
 1474 002410 000000
 1475 002412 000000
 1476 002414 003406
 1477 002416 002403
 1478 002420
 1479 002420 000000
 1480 002422 000000
 1481 002424 000000
 1482 002426 000000
 1483 002430 000000
 1484 002432 000000
 1485 002434 000000
 1486 002436 000000
 1487
 1488
 1489 002440

.SBTTL GLOBAL DATA SECTION

```

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

```

```

:*****
:* MISCELLANEOUS STORAGE
:*****

```

```

SAVE4: .WORD 0 ;SAVE LOC 4 HERE (ERROR TRAP VECTOR)
SAVE6: .WORD 0
PSTACK: .WORD 0
SUBRPC: .WORD 0
ERROR1: .WORD 0
FRSTIM: .WORD 0
LOGDEV: .WORD 0
IFLAG: .WORD 0
$GDDAT: .WORD 0 ;GOOD AND BAD DATA STORAGE
$BDDAT: .WORD 0
COUNT: .WORD 0
REG: .WORD 0
STARES: .WORD 0 ;INDICATES PASSES
DEVMAP: .WORD 0
DEVPTR: .WORD 0
FRSPAS: .WORD 0
MODINT: .WORD 0 ;MODEM INTERFACE SELECTION
TRIBN: .WORD 0 ;POINTS TO CURRENT TRIP NUMBER.
TRIBMX: .WORD 32. ;MAXIMUM NUMBER OF TRIBS
TRIBH: .WORD 0 ;VALUE OF HIGHEST TRIB USED

ROMN: .WORD 0 ;CURRENT ROM AND USED FOR TEST #
ROMN1: .WORD 0 ;CURRENT ROM NUMBER
WORDT: .WORD 0 ;CURRENT ROM CONTENTS.
CWORD: .WORD -1 ;CURRENT CRC CAL.
MODQ22: .WORD 0 ;DMV Q22 FORMAT FLAG (Q22 MODE)
EXLOOP: .WORD 0 ;DMV EXTERNAL LOOP FLAG
ERRWRD: .WORD 0 ;ERROR OCCURRED.
CADDR: .WORD 0 ;CURRENT ROM ADDR.
ERRADD: .WORD 0 ;PC OF ERROR
PERR: .WORD 0 ;PROCEDURE ERROR CHECKED
TSSADD: .WORD 0 ;WORD FOR TSS ADD
TYLST: .WORD 3406
        .WORD 2403

TYEND:
TXADD: .WORD 0 ;TX BUFF ADDRESS
RXADD: .WORD 0 ;RX BUFF ADDRESS
RXCC: .WORD 0 ;RX CHAR COUNT
TXCC: .WORD 0 ;TX CHAR COUNT
CODEW: .WORD 0 ;LOCATION FOR ERROR CODES
GENWRD: .WORD 0 ;USED FOR MAINT STATE AND EX MEM
CRCCAL: .WORD 0 ;TEMP FOR CPC
ROMADD: .WORD 0 ;ROM ADDRESS

```

```

:***** CURRENT DEVICE PARAMETERS *****

```

BSELO:

CZDMTC.P11 25-MAR-81 08:24

GLOBAL DATA SECTION

1490	002440		SELO:			
1491	002440	160170	MPCSR:	.WORD	160170	; POINTER TO M8200,4,7 CSR'S
1492	002442	160171	BSEL1:	.WORD	160171	; POINTER TO BSEL1
1493	002444		SEL2:			
1494	002444	160172	BSEL2:	.WORD	160172	
1495	002446	160173	BSEL3:	.WORD	160173	
1496	002450		BSEL4:			
1497	002450	160174	SEL4:	.WORD	160174	; POINTER TO SEL4
1498	002452	160175	BSEL5:	.WORD	160175	
1499	002454		BSEL6:			
1500	002454	160176	SEL6:	.WORD	160176	
1501	002456	160177	BSEL7:	.WORD	160177	
1502	002460		BSEL10:			
1503	002460	160200	SEL10:	.WORD	160200	; POINTER TO SEL10 (REQ'D FOR DMV Q22 MODE)
1504	002462		KMRVEC:			
1505	002462	000300	MPIVEC:	.WORD	300	; M8200,4,7 INPUT INTERRUPT VECTOR
1506	002464		KMTVEC:			
1507	002464	000304	MPOVEC:	.WORD	304	; M8200,4,7 OUTPUT INTERRUPT VECTOR
1508	002466	000000	SPEEDM:	.WORD	0	; SPEED OF LINE UNIT
1509	002470		KMRLVL:			
1510	002470		KMTLVL:			
1511	002470	000240	MPRIOR:	.WORD	240	; M8200,4,7 DEVICE PRIORITY
1512	002472	000000	OPTYP:	.WORD	0	; OPTION TYPE
1513	002474	000000	IFTYP:	.WORD	0	; INTERFACE TYPE
1514	002476	000000	TSTCON:	.WORD	0	; TEST CONNECTOR INDICATOR
1515	002500	000000	RETADR:	.WORD	0	; SUBR ERROR RETURN ADDRESS
1516	002502	000000	REDBYT:	.WORD	0	; LO BYTE CONTAINS BYTE READ FROM LU REG
1517	002504	000000	WRIBYT:	.WORD	0	; LO BYTE CONTAINS BYTE TO LOAD INTO LU REG
1518	002506	000000	AXNUM:	.WORD	0	; NUMBER (0-7) OF EXTENDED REG BYTE BEING TESTED
1519	002510	000000	DISILO:	.WORD	0	; CONTAINS CURRENT STATE OF DISSI IN BITS
1520						
1521			;***** STORAGE FOR DATA READ IN ADDRESS TESTS *****			
1522	002512	000	REDDAT:	.BYTE	0	
1523	002513	000		.BYTE	0	
1524	002514	000		.BYTE	0	
1525	002515	000		.BYTE	0	
1526	002516	000		.BYTE	0	
1527	002517	000		.BYTE	0	
1528	002520	000		.BYTE	0	
1529	002521	000		.BYTE	0	
1530						
1531			;***** GENERAL PURPOSE SCRATCH STORAGE *****			
1532	002522	000000	REG0:	.WORD	0	
1533	002524	000000	REG1:	.WORD	0	
1534	002526	000000	REG2:	.WORD	0	
1535	002530	000000	REG3:	.WORD	0	
1536	002532	000000	REG4:	.WORD	0	
1537	002534	000000	REG5:	.WORD	0	
1538	002536	000000	REG6:	.WORD	0	
1539	002540	000000	REG7:	.WORD	0	
1540						
1541			;***** SCRATCH STORAGE FOR MESSAGE REPORTING *****			
1542	002542	000000	\$TMP0:	.WORD	0	
1543	002544	000000	TMP0:	.WORD	0	
1544	002546	000000	TMP1:	.WORD	0	
1545	002550	000000	TMP2:	.WORD	0	

CZDMTC.P11 25-MAR-81 08:24

GLOBAL DATA SECTION

1546	002552	000000
1547	002554	000000
1548	002556	000000
1549	002560	000000
1550	002562	000000
1551		

TMP3:	.WORD	0
TMP4:	.WORD	0
TMP5:	.WORD	0
TMP6:	.WORD	0
TMP7:	.WORD	0

CZDMTC.P11 25-MAR-81 08:24

GLOBAL DATA SECTION

1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591

002564			
002564	046504	026520	030461
002572	047440	020122	046504
002600	026526	030461	000
	002606		
002606			
002606	046504	020120	051117
002614	042040	053115	030455
002622	020061	052506	041516
002630	044524	047117	046101
002636	042040	040511	027107
002644	000		
	002646		

.SBTTL GLOBAL TEXT SECTION

```

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

```

```

:*****
:* NAMES OF DEVICES SUPPORTED BY PROGRAM
:*****

```

```

L$DVTYP::
      .ASCII /DMP-11 OR DMV-11/

      .EVEN

L$DESC::
      .ASCII /DMP OR DMV-11 FUNCTIONAL DIAG./

      .EVEN

```

```

:
: FORMAT STATEMENTS USED IN PRINT CALLS
:

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

.SBTTL GLOBAL SUBROUTINES

:/ THE GLOBAL SUBROUTINES ARE CALLED BY MORE THAN ONE TEST

1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646

FUNCTIONAL DESCRIPTION: WRDO.. WAITS FOR READY OUT
FIRST SAVE THE CALLING ADDRESS
IN ERRADD. THEN SEE IF TIME OUT OCCURED
IF TIME OUT EXIT ROUTINE..ELSE SEE IF
READY OUT SET IF READY OUT SET EXIT
ROUTINE. IF NOT THEN WAIT A WHILE
THEN SEE IF READY IN SET. IF READY IN
IS SET REPORT ERROR AND EXIT ROUTINE.
IF NOT READY IN THEN GO BACK AND CHECK
FOR TIME OUT.
NOTE: CAN BE ENTERED AT WRD01 IF CALLING
ADDRESS FROM R5 DOES NOT NEED TO BE SAVED.
INPUTS: R5=ADDRESS FROM WHERE ROUTINE WAS CALLED
OUTPUTS: ERRWRD= -1 IF ERROR OCCURED IN ROUTINE.
SUBORDINATE ROUTINES USED:
TOUT - TIME OUT ROUTINE
WAIT50 - SHORT DELAY ROUTINE
CALLING SEQUENCE:
JSR R5,WRDO

WRDO: MOV R5,ERRADD ;STORE ERROR ADD. AWAY
WRD01: JSR R5,TOUT ; GO TO TIME OUT ROUTINE
TST ERRWRD ;CHECK IF ERROR
BMI WRDOE ;EXIT NOW
BIT #RDO,@BSEL2 ;RDO SET?
BNE WRDOE ;EXIT IF RDO IS SET
JSR PC,WAIT50 ;ELSE DELAY A LITTLE
BIT #RDI,@BSEL2 ;THEN SEE IF RDI IS SET
BEQ WRD01 ;IF NOT THEN GO BACK TO START
; ERROR -UNEXPECTED RDI SET
TRAP C\$ERDF
.WORD 1
.WORD MEF14
.WORD ERR26
DEC ERRWRD
CLRB @BSEL2 ;CLEAR RDO
WRDOE: RTS R5 ;EXIT

002646 010537 002406
002652 004537 005070
002656 005737 002402
002662 100422
002664 032777 000200 177552
002672 001016
002674 004737 004244
002700 032777 000020 177536
002706 001761
002710 104455
002712 000001
002714 012021
002716 010266
002720 005337 002402
002724 105077 177514
002730 000205

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692

```

*****
: FUNCTIONAL DESCRIPTION: WRDI - WAIT FOR READY IN
: THIS ROUTINE FIRST SAVES THE CALLING ADDRESS
: IN ERRADD, UNLESS ENTERED AT WRDI1.
: THEN CHECK FOR TIME OUT IF TIME OUT REPORT
: ERROR AND EXIT. IF NOT TIME OUT CHECK FOR
: READY IN. IF READY IN EXIT IF NOT READY IN
: DELAY A LITTLE AND CHECK FOR READY OUT. IF
: READY OUT REPORT ERROR AND EXIT ROUTINE.
: IF NOT READY OUT GO BACK AND CHECK FOR TIME OUT.
:
: INPUTS: R5= CALLING ADDRESS
: OUTPUTS: ERRWRD= -1 IF ERROR OCCURED IN ROUTINE
: SUBORDINATE ROUTINES USED:
: TOUT- TIME OUT
: WAIT50- DELAY A LITTLE
: CALLING SEQUENCE:
: JSR R5,WRDI ;OR
: JSR R5,WRDI1
:
*****

```

```

002732 010537 002406
002736 004537 005070
002742 005737 002402
002746 100422
002750 032777 000020 177466
002756 001016
002760 004737 004244
002764 032777 000200 177452
002772 001761
002774 104455
002776 000002
003000 012066
003002 010266
003004 005337 002402
003010 105077 177430
003014 000205

```

```

WRDI:  MOV R5,ERRADD ;STORE AWAY ERROR ADD.
WRDI1: JSR R5,TOUT ;GO TO TIME OUT
      TST ERRWRD ;IF ERROR EXIT
      BMI WRDIE
1$:    BIT #20,@BSEL2 ;RDI SET?
      BNE WRDIE ;YES-EXIT
      JSR PC,WAIT50 ;SHORT DELAY
      BIT #200,@BSEL2 ;RDYO SET?
      BEQ WRDI1 ;NO-LOOP.
      ;RDO INSTEAD OF RDI
      TRAP C$ERDF
      .WORD 2
      .WORD MEF15
      .WORD ERR26
      DEC ERRWRD ;SET ERROR OCCURRED
      CLRB @BSEL2 ;CLEAR RDYO
WRDIE: RTS R5 ;EXIT.

```


CZDMTC.P11 25-MAR-81 08.24

GLOBAL SUBROUTINES

1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748

003016 010537 002406
003022 004537 002652
003026 005737 002402
003032 100452
003034 117737 177404 002340
003042 042737 177770 002340
003050 122737 000001 002340
003056 001415
003060 012737 000001 002336
003066 012737 012000 002430
003074 104455
003076 000003
003100 011506
003102 010324
003104 005337 002402
003110 000423
003112 117737 177336 002340
003120 123737 002410 002340
003126 001414
003130 013737 002410 002336
003136 012737 011762 002430
003144 104455
003146 000004
003150 011506
003152 010324
003154 005337 002402
003160
003160 000205

```

*****
: FUNCTIONAL DESCRIPTION: WFPE - WAIT FOR PROCEDURE ERROR
: FIRST SAVE CALLING ADDRESS IN ERRADD.
: THEN WAIT FOR READY OUT, IF ERROR FROM
: WRDO ROUTINE EXIT THIS ROUTINE. ELSE
: GET CONTROL KEY FROM BSEL2 IF NOT CONTROL
: OUT REPORT ERROR AND EXIT. ELSE CHECK THAT
: CONTROL OUT CODE IS SAME AS IN PERR. IF
: EQUAL THEN EXIT ELSE REPORT ERROR AND EXIT.
:
: INPUTS:      R5= CALLING ADDRESS
:              PERR = PROCEDURE ERROR EXPECTED.
:
: OUTPUTS:     ERRWRD= -1 IF ERROR OCCURED IN ROUTINE
:
: SUBORDINATE ROUTINES USED:
:              WRD01 - WAIT FOR READY OUT
:
: CALLING SEQUENCE:
:              JSR      R5,WFPE
*****

```

```

WFPE:  MOV      R5,ERRADD      ;STORE OFF ERROR ADDRESS
      JSR      R5,WRD01      ;WAIT FOR READY OUT
      TST     ERRWRD
      BMI     20$           ;IF ERROR OCCURRED IN SR
      ;EXIT THIS SR.
      MOVB   @BSEL2,$BDDAT
      BIC   #'C<7>,$BDDAT  ;STRIP DATA TO CONTROL KEY
      CMPB  #01,$BDDAT
      BEQ   10$           ;GO TO 10 IF CONTROL OUT
      ;ELSE REPORT ERROR
      MOV   #01,$GDDAT     ;SET GOOD DATA TO 01
      MOV   #M18F,COEW     ;SET UP CODE WORD
      ;ERROR NOT CONTROL OUT
      TRAP  C$ERDF
      .WORD 3
      .WORD EROIC
      .WORD ERR27
      DEC   ERRWRD        ;SET ERROR OCCURRED
      BR   20$           ;AND EXIT SUBROUTINE
10$:  MOVB   @BSEL6,$BDDAT  ;MOVE ERROR CODE TO BDDAT
      CMPB  PERR,$BDDAT   ;IS IT WHAT IT SHOULD BE
      BEQ   20$           ;IF SO GO TO 20
      MOV   PERR,$GDDAT   ;PUT EXPECTED IN GOOD DATA
      MOV   #M13F,COEW    ;SET UP ERROR WORD
      ;ERROR BAD ERROR CODE RETURNED
      TRAP  C$ERDF
      .WORD 4
      .WORD EROIC
      .WORD ERR27
      DEC   ERRWRD        ;SET ERROR INDICATOR
20$:  RTS     R5           ;RETURN TO CALLER

```

CZDMTC.P11

25-MAR-81 08:24

GLOBAL SUBROUTINES

1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783

```

*****
: FUNCTIONAL DESCRIPTION:   CONTIN - CONTROL IN ROUTINE
: THIS ROUTINE SAVES THE CALLING ADDRESS IN R5.
: THEN SETS RQI AND WAITS FOR RDI TO BE RETURNED BY
: THE DMP/V-11. IF WRDI REPORTS ERROR EXIT TEST. ELSE
: LOAD BSEL 3 WITH TRIB NUMBER FROM TRIBN,CLEAR THE RQI
: BIT,MOV DATA FROM R4 TO SEL4,DATA FROM R3 TO SEL6, AND
: THEN ISSURE CONTROL IN AND EXIT ROUTINF.
:
: INPUTS:                   R4 = SEL4 DATA
:                           R3 = SEL6 DATA
:                           TRIBN = TRIBUTARY NUMBER.
:                           R5 = CALLING ADDRESS
:
: OUTPUTS:                  ERRWRD = -1 IF ERROR REPORTED IN THIS OR ANY SUBODINATE
:                           SUB ROUTINE.
: SUBORDINATE ROUTINES USED:
:                           WRDI1 - WAIT FOR READY IN.
: CALLING SEQUENCE:
:                           JSR      R5,CONTIN
:-----*****

```

```

003162 010537 002406
003166 052777 000200 177244
003174 004537 002736
003200 005737 002402
003204 100415
003206 113777 002360 177232
003214 042777 000200 177216
003222 010477 177222
003226 010377 177222
003232 112777 000001 177204
003240 000205

```

```

CONTIN: MOV      R5,ERRADD      ;SET UP ERROR ADDRESS
        BIS      #RQI,@BSELO    ;SET REQUEST
        JSR      R5,WRDI1      ;GO WAIT FOR RDI
        TST      ERRWRD
        BMI      43$           ;EXIT IF ERROR OCCURRED
        MOVB     TRIBN,@BSEL3    ;SET TRIBN
        BIC      #RQI,@BSELO    ;CLEAR REQUEST
        MOV      R4,@BSEL4      ;SET DATA
        MOV      R3,@BSEL6      ;SET REQUEST TYPE
        MOVB     #01,@BSEL2     ;DO CONTROL IN
43$:    RTS      R5             ;RETURN TO CALLER

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836

003242 010537 002406
003246 004537 002652
003252 005737 002402
003256 100444
003260 117737 177160 002340
003266 042737 177770 002340
003274 023737 002340 002336
003302 001412
003304 012737 012007 002430
003312 104455
003314 000005
003316 011506
003320 010324
003322 005337 002402
003326 000420
003330 013737 002360 002336
003336 117737 177104 002340
003344 023737 002340 002336
003352 001406
003354 104455
003356 000006
003360 012242
003362 007600
003364 005337 002402
003370 000205

FUNCTIONAL DESCRIPTION: GETOUT - GET OUTPUT CODE

THIS SUB-ROUTINE WAITS FOR RDO(REPORTS ERROR IF RDI OR TIME OUT);CHECKS THAT OUTPUT COMMAND TYPE IS THE SAME AS THE VALUE IN \$GDDAT(REPORTS ERROR IF NOT);THEN CHECKS THAT TRIB NUMBER IN BSEL3 IS EQUAL TO THE VALUE IN TRIBN(REPORTS ERROR IF NOT THEN RETURNS TO CALLER.

INPUTS: \$GDDAT = OUTPUT COMMAND TYPE EXPECTED
TRIBN = TRIBUTARY ADDRESS EXPECTED
R5 = ADDRESS OF CALLING ROUTINE

OUTPUTS: ERRWRD = -1 IF ERROR OCCURED

SUBORDINATE ROUTINES USED:
WRD01 - WAIT FOR READY OUT

CALLING SEQUENCE:
JSR R5,GETOUT

GETOUT: MOV R5,ERRADD ;STORE OFF ERROR ADD.
JSR R5,WRD01 ;GO WAIT FOR READY OUT
TST ERRWRD
BMI 20\$;EXIT IF ERROR OCCURRED
MOVB @BSEL2,\$BDDAT ;GET COMMAND TYPE TO BDDAT
BIC #'C<7>,\$BDDAT ;STRIP IT TO JUST COMMAND TYPE
CMP \$BDDAT,\$GDDAT ;IS IT THE RIGHT VALUE??
BEQ 10\$;IF YES GO TO 10
;ELSE REPORT ERROR

MOV #M28F,CODEW
TRAP C\$ERDF
.WORD 5
.WORD EROIC
.WORD ERR27
DEC ERRWRD
BR 20\$;AND EXIT ON ERROR

10\$: MOV TRIBN,\$GDDAT ;SET UP GDDAT FOR GOOD TRIBN
MOVB @BSEL3,\$BDDAT ;GET TRIB NUMBER RETURNED
CMP \$BDDAT,\$GDDAT ;ARE THEY THE SAME??
BEQ 20\$;IF YES GO TO 20
;ELSE REPORT ERROR

TRAP C\$ERDF
.WORD 6
.WORD MEF18A
.WORD ERR18
DEC ERRWRD
20\$: RTS R5 ;RETURN TO CALLER

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852 003372 010537 002406
1853 003376 017737 177052 002340
1854 003404 042737 177600 002340
1855 003412 023737 002340 002336
1856 003420 001411
1857
1858 003422 012737 011747 002430
1859 003430 104455
1860 003432 000007
1861 003434 011506
1862 003436 010324
1863 003440 005337 002402
1864 003444 000205
1865

```

```

:*****
: FUNCTIONAL DESCRIPTION:   GETRKY - GET RETURN KEY VALUE
: THIS ROUTINE GETS THE VALUE OF THE RETURN KEY
: FROM BSEL6 ADN COMPARES IT TO THE VALUE IN
: $GDDAT. IF EQUAL EXIT IF NOT EQUAL REPORT
: ERROR AND EXIT.
:
: INPUTS:      R5      = ADDRESS OF CALLER
:              $GDDAT = VALUE OF EXPECTED RETURN KEY
:
: OUTPUTS:     ERRWRD = -1 IF ERROR OCCURS
: CALLING SEQUENCE:
:              JSR     R5,GETRKY
:*****
GETRKY: MOV     R5,ERRADD      ; STORE OFF ERROR ADDRESS
        MOV     @BSEL6,$BDDAT ; GET RETURN KEY FROM BSEL6
        BIC     #^C<177>,$BDDAT ; STRIP TO VALID BITS
        CMP     $BDDAT,$GDDAT ; ARE THE VALUES EQUAL
        BEQ     10$          ; IF YES GO TO 10
                          ; ELSE ERROR.....
        MOV     #M12F,CODEW   ; SET UP CODE WORD
        TRAP   C$ERDF
        .WORD  7
        .WORD  EROIC
        .WORD  ERR27
        DEC    ERRWRD        ; SET ERROR OCCURRED
10$:    RTS     R5           ; RETURN TO CALLER

```

CZDMTC.P11

25-MAR-81 08:24

GLOBAL SUBROUTINES

1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895

003446 010537 002406
003452 017737 176772 002340
003460 023737 002340 002336
003466 001411

003470 012737 012014 002430
003476 104455
003500 000010
003502 011506
003504 010324
003506 005337 002402
003512 000205

```
*****
: FUNCTIONAL DESCRIPTION:   GETDAT - GET DATA CODE
: THIS ROUTINE GETS THE DATA CODE FROM BSEL4
: AND COMPARES IIT TO THE VALUE IN $GDDAT
: IF EQUAL EXIT ELSE REPORT ERROR AND EXIT
:
: INPUTS:      R5      = ADDRESS OF CALLER
:              $GDDAT = VALUE OF EXPECTED DATA
:
: OUTPUTS:     ERRWRD = -1 IF ERROR OCCURED
: CALLING SEQUENCE:
:              JSR     R5,GETDAT
:*****
```

```
GETDAT: MOV     R5,ERRADD      ;STORE OFF ERROR ADDRESS
        MOV     @BSEL4,$BDDAT ;GET DATA
        CMP     $BDDAT,$GDDAT ;COMPARE GOOD AND BAD
        BEQ    10$           ;IF OK GO TO 10
                               ;ELSE ERROR
                               ;ERROR BAD DATA CODE
        MOV     #M30F,CODEW
        TRAP   C$ERDF
        .WORD  8
        .WORD  EROIC
        .WORD  ERR27
        DEC    ERRWRD        ;SET ERROR OCCURRED
10$:    RTS     R5           ;RETURN TO CALLER
```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929

003514 010537 002406
003520 017737 176730 002340
003526 005737 002376
003532 001403
003534 017737 176720 002340
003542 042737 140000 002340
003550 023737 002340 002336
003556 001406

003560 104455
003562 000011
003564 012133
003566 010114
003570 005337 002402
003574 000205

```

*****
: FUNCTIONAL DESCRIPTION:      GETCC - GET CHARACTER COUNT
: THIS ROUTINE GETS THE CHAR. COUNT FROM EITHER:
: (1) CSR6 IF DMP OR DMV IN Q18 MODE
: (2) CSR10 IF DMV IN Q22 MODE
: AND THEN COMPARES IT TO THE VALUE IN $GDDAT.
: IF EQUAL EXIT, ELSE REPORT ERROR AND EXIT
:
: INPUTS:      R5      = ADDRESS OF CALLER
:              $GDDAT = VALUE OF EXPECTED DATA
:
: OUTPUTS:     ERRWRD = -1 IF ERROR OCCURED
: CALLING SEQUENCE:
:              JSR     R5,GETCC
*****
GETCC:  MOV     R5,ERRADD      ;STORE OFF RETURN ADDRESS
        MOV     @SEL6,$BDDAT  ;GET CSR6 TO BDDAT
        TST    MODQ22        ;IS THIS DMV W/Q22 ?
        BEQ    1$
        MOV     @SEL10,$BDDAT ;IF YES: GET CSR10 INSTEAD
1$:     BIC     #140000,$BDDAT ;STRIP TO CHAR COUNT
        CMP     $BDDAT,$GDDAT ;COMPARE
        BEQ    10$           ;IF OK GO TO 10
                                ;ELSE REPORT ERROR : BAD CHAR COUNT
                                TRAP   C$ERDF
                                .WORD  9
                                .WORD  MEF16A
                                .WORD  ERR23
10$:    DEC     ERRWRD        ;SET ERROR OCCURRED
        RTS     R5           ;RETURN TO CALLER

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966

003576 010537 002406
003602 017737 176642 002340
003610 023737 002340 002336
003616 001015
003620 005737 002376
003624 001420
003626 012737 000000 002340
003634 017737 176614 002336
003642 023737 002340 002336
003650 001406

003652
003652 104455
003654 000012
003656 012176
003660 010114
003662 005337 002402
003666 000205

```
*****
: FUNCTIONAL DESCRIPTION:      GETBA - GET BUFFER ADDRESS
: THIS ROUTINE GETS THE BUFFER ADDRESS FROM SEL4
: AND THEN COMPARES IT TO THE VALUE IN $GDDAT
: IF EQUAL EXIT ELSE REPORT ERROR AND EXIT
: (IF DMV IN Q22 MODE: SEL6 IS CHECKED FOR 0).
:
: INPUTS:      R5      = ADDRESS OF CALLER
:              $GDDAT = VALUE OF EXPECTED BUFFER ADDRESS
:
: OUTPUTS:     ERRWRD = -1 IF ERROR OCCURED
: CALLING SEQUENCE:
:              JSR     R5,GETBA
:*****
```

```
GETBA:  MOV     R5,ERRADD      ;STORE OFF ERROR ADDRESS
        MOV     @SEL4,$BDDAT  ;GET ADDRESS OUTPUT
        CMP     $BDDAT,$GDDAT ;ARE THEY EQUAL ?
        BNE    1$           ;IF NOT: REPORT IN
        TST    MODQ22        ;* IS THIS Q22 MODE ?
        BEQ    10$          ;* IF NOT: EXIT
        MOV     #0,$BDDAT     ;* GET EXPECTED EXTENDED ADDRESS
        MOV     @SEL6,$GDDAT ;* GET ACTUAL EXTENDED ADDRESS
        CMP     $BDDAT,$GDDAT ;* ARE THEY EQUAL ?
        BEQ    10$          ;IF YES: EXIT
                                ;ELSE ERROR
1$:     -RAP    C$ERDF
        .JRD   10
        .WORD  MEF17A
        .WORD  ERR23
10$:   DEC    ERRWRD        ;SET ERROR OCCURRED
        RTS    R5          ;RETURN TO CALLER
```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995

003670 010537 002406
003674 017737 176554 002340
003702 042737 177400 002340
003710 023737 002340 002336
003716 001411
003720 012737 012000 002430
003726 104455
003730 000013
003732 011506
003734 010324
003736 005337 002402
003742 000205

```
*****
: FUNCTIONAL DESCRIPTION: GETOC - GET OUTPUT CODE
: THIS ROUTINE GETS THE OUTPUT CODE FROM
: BSEL6 AND THEN COMPARES IT TO
: THE VALUE IN $GDDAT
: IF EQUAL EXIT ELSE REPORT ERROR AND EXIT
:
: INPUTS: R5 = ADDRESS OF CALLER
: $GDDAT = VALUE OF EXPECTED OUTPUT CODE
:
: OUTPUTS: ERRWRD = -1 IF ERROR OCCURED
: CALLING SEQUENCE:
: JSR R5,GETOC
:*****
```

```
GETOC: MOV R5,ERRADD ;STORE OFF ERROR ADDRESS
MOV @BSEL6,$BDDAT ;GET OUTPUT FROM BSEL6
BIC #^C<377>,$BDDAT ;STRIP TO VALID BITS
CMP $BDDAT,$GDDAT ;ARE THEY EQUAL
BEQ 10$ ;IF SO GO TO 10
;ELSE ERROR
MOV #M18F,COEWD
TRAP C$ERDF
.WORD 11
.WORD EROIC
.WORD ERR27
DEC ERRWRD ;SET ERROR OCCURRED
10$: RTS R5 ;RETURN TO CALLER
```


CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018 003744
2019 003744 004537 002646
2020 003750 005737 002402
2021 003754 100524
2022 003756 117737 176462 002340
2023 003764 042737 177770 002340
2024 003772 122737 000000 002340
2025 004000 001450
2026 004002 122737 000001 002340
2027 004010 001012
2028 004012 012737 000302 002336
2029 004020 117737 176430 002340
2030 004026 123737 002336 002340
2031 004034 001412
2032 004036 012737 011762 002430
2033 004044 104455
2034 004046 000014
2035 004050 011506
2036 004052 010324
2037 004054 005337 002402
2038 004060 000462
2039 004062
2040
2041
2042 004062 005237 177572
2043 004066 012737 004216 000004
2044 004074 005737 120000
2045 004100 005037 177572
2046 004104 104455
2047 004106 000015
2048 004110 000000
2049 004112 007706
2050 004114 005337 002402
2051 004120 000442

```

*****
: FUNCTIONAL DESCRIPTION: MEMEX - MEMORY EXTENSION CODE
: THIS ROUTINE IS USED WITH THE
: MEMORY EXTENSION TESTS. THE ROUTINE FIRST
: CHECKS FOR A CONTROL OUT.. IF THE CONTROL
: OUT IS A REC COMPLETE IT COMPARES THE FIRST
: DATA WORD ON THE EXTENSIONS PAGE IF GOOD
: THEN EXIT IF BAD REPORT ERROR AND EXIT.
: IF CONTROL OUT IS NON EXISTENT MEMORY THEN
: CHECK TO BE SURE MEMORY IS NON-EXISTENT
: IF MEMORY EXIST THEN PRINT ERROR AND
: EXIT ROUTINE.
:
: INPUTS:      R5      = CALLING ADDRESS
:
: OUTPUTS:     ERRWRD = -1 IF ERROR OCCURED
:
: SUBORDINATE ROUTINES USED:
:   WRDO      = WAIT FOR READY OUT
:
: CALLING SEQUENCE:
:   JSR      R4, MEMEX
:
*****

```

```

MEMEX:
EXMEM:  JSR      R5, WRDO      ;WAIT FOR READY OUT
        TST      ERRWRD
        BMI      EXMEMX
        MOVB     @BSEL2, $BDDAT ;
        BIC      #'C<7>, $BDDAT ;STRIP TO TYPE CODE
        CMPB     #0, $BDDAT    ;IS IT REC COMP
        BEQ      EXMEMA       ; IF YES THEN GO TO A
        CMPB     #1, $BDDAT    ;IF NOT IS IT CONTROL OUT
        BNE      EXMEMB       ;IF NOT GO TO B
        MOV      #302, $GDDAT
        MOVB     @BSEL6, $BDDAT ;
        CMPB     $GDDAT, $BDDAT ;IS IT NON EXISTENT MEM?
        BEQ      EXMEMC       ;IF YES GO TO C
EXMEMB: MOV      #M13F, CODEW
        TRAP     C$ERDF
        .WORD    12
        .WORD    EROIC
        .WORD    ERR27
        DEC      ERRWRD       ;SET ERROR FLAG
        BR       EXMEMX      ;PRINT ERROR AND EXIT
EXMEMC: ;IF ERROR IS NON EXISTENT
        INC      177572      ;INCREMENT MM
        MOV      #METB, 4    ;SET UP TRAP
        TST      @#120000    ;TEST THE NON EXISTENT
        CLR      177572      ;CLEAR MM
        TRAP     C$ERDF
        .WORD    13
        .WORD    0
        .WORD    ERR20
        DEC      ERRWRD
        BR       EXMEMX      ;PRINT ERROR AND EXIT TEST

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2052
2053           ; GET HERE IF BUFFER RETURNED OK
2054
2055 004122 005237 177572      EXMEMA: INC 177572           ;ENABLE MM
2056 004126 012737 004202 000004  MOV #METC,4       ;SET UP TRAP
2057 004134 013737 033456 002336  MOV R1+2,$GDDAT  ;GET FIRST WORD FROM NEW PAGE
2058 004142 013737 120000 002340  MOV 120000,$BDDAT ;AND FIRST RX WORD
2059 004150 005037 177572      CLR 177572        ;DISABLE MM
2060 004154 023737 002336 002340  CMP $GDDAT,$BDDAT ;COMPARE DATA
2061 004162 001421              REQ EXMEMX         ;EXIT IF GOOD
2062 004164 104455              TRAP C$ERDF
2063 004166 000016              .WORD 14
2064 004170 000000              .WORD 0
2065 004172 007740              .WORD ERR21
2066 004174 005337 002402      DEC ERRWRD
2067 004200 000412              BR EXMEMX         ;AND EXIT
2068 004202 005037 177572      METC: CLR 177572   ;DISABLE MM
2069 004206 104455              TRAP C$ERDF
2070 004210 000017              .WORD 15
2071 004212 000000              .WORD 0
2072 004214 010022              .WORD ERR22
2073 004216 005037 177572      METB: CLR 177572   ;DISABLE MM
2074 004222 062716 000004      META: ADD #4,(SP)
2075 004226 013737 002316 000004  EXMEMX: MOV SAVE4,4
2076 004234 013737 002320 000006  MOV SAVE6,6       ;RESTORE TRAPS
2077 004242 000205              RTS R5
2078

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089 004244 010146
2090 004246 012701 000310
2091 004252 005737 002472
2092 004256 001402
2093 004260 062701 000620
2094 004264 005301
2095 004266 001376
2096 004270 012601
2097 004272 000207
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112 004274 005077 176140
2113 004300 113777 002405 176136
2114 004306 004537 004346
2115 004312 121053
2116 004314 042737 000377 004334
2117 004322 153737 002404 004334
2118 004330 004537 004346
2119 004334 100000
2120 004336 052777 002000 176074
2121 004344 000207
    
```

```

*****
: FUNCTIONAL DESCRIPTION:      WAIT50 - WAIT 50 MICRO SECONDS
:                             THIS ROUTINE COUNTS DOWN R1 FROM 200 TO 0
:                             IF DMP AND FROM 600 TO 0 IF DMV. THIS
:                             IS USED AS A DELAY ROUTINE
:
: CALLING SEQUENCE:
:                             JSR      PC,WAIT50
:-----*****
    
```

```

WAIT50: MOV      R1,-(SP)      ;SAVE R1
        MOV      #200.,R1    ;INIT COUNTER
        TST      OPTYP
        BEQ      3$         ;IF DMP GO TO 3
        ADD      #400.,R1    ;ELSE TRIPLE UP TIMER FOR DMV
3$:     DEC      R1          ;DECREMENT COUNTER
        BNE      3$         ;BR IF NOT DONE YET
        MOV      (SP)+,R1    ;RESTORE R1
        RTS      PC         ;RETURN
    
```

```

*****
: FUNCTIONAL DESCRIPTION:      GWORD - GET WORD
:                             THIS ROUTINE READS A WORD FROM THE M8207 ROM.
:
: INPUTS:      CADDR = ADDRESS TO BE READ
:
: OUTPUTS:     SEL6 = DATA READ
: SUBORDINATE ROUTINES USED:
:             ROMCLK - ROUTINE TO ISSUE CLOCKS TO ROM CIRCUIT
: CALLING SEQUENCE:
:             JSR      PC,GWORD
:-----*****
    
```

```

GWORD: CLR      @SELO        ;INIT
        MOVB     CADDR+1,@SEL2 ;NOW HIGH BYTE OF ADDRESS
        JSR      R5,ROMCLK
        .WORD    121053      ;MOV IBUS* 2 TO OBUS* 13
        BIC      #377,1$     ;STRIP ADDR FLIED.
        BISB     CADDR,1$    ;ADD IN IMM ADDR.
        JSR      R5,ROMCLK   ;GO DO BRANCH.
1$:     .WORD    100000      ;BRANCH EXT PUTS ADDR. IN PCREG.
        BIS      #2000,@SELO ;SET READ ENABLE.
        RTS     PC          ;EXIT.
    
```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136 004346
2137 004346 152777 000002 176066
2138 004354 012577 176074
2139 004360 152777 000003 176054
2140 004366 142777 000007 176046
2141 004374 000205
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158 004376 010146
2159 004400 010246
2160 004402 012702 000020
2161 004406 000241
2162 004410 006037 002374
2163 004414 006037 002372
2164 004420 102011
2165 004422 012701 102010
2166 004426 043701 002374
2167 004432 042737 102010 002374
2168 004440 050137 002374
2169 004444 005302
2170 004446 003357
2171 004450 012602
2172 004452 012601
2173 004454 000207
2174

```

```

:*****
: FUNCTIONAL DESCRIPTION: ROMCLK - ROM CLOCK ROUTINE
: THIS ROUTINE ISSUES A SINGE STEP TO THE
: M8207.
: INPUTS: R5 - POINTS TO INSTRUCTION TO BE STEPPED
: RETURN: RETURN IS TO WORD FOLLWOING INSTRUCTION
: CALLING SEQUENCE:
: JSR R5,ROMCLK
: .WORD INSTR ;INSTURCITON TO EXECUTE
:*****

```

```

ROMCLK:
BISB #2,@BSEL1 ;SET ROMI
MOV (R5)+,@BSEL6 ;SET INSTRUCTION.
BISB #3,@BSEL1 ;CLOCK INSTR.
BICB #7,@BSEL1 ;CLEAR.
RTS R5

```

```

:*****
: FUNCTIONAL DESCRIPTION: CRCR - CRC CALCULATE ROUTINE
: THIS ROUTINE TAKES 16 BITS OF DATA FROM WORDT
: AND CONVERTS THEM INTO PART OF THE SERIAL STREAM
: THAT IS BEING USED TO CALCULATE A CRC-CCITT WORD.
: INPUTS: WORDT - WORD TO CALCULATE ON
: IMPLICIT INPUTS:
: CWORD - MUST BE A -1 FIRST TIME CALLED
: OUTPUTS: CWORD - 16 BIT CALCULATED WORD
: CALLING SEQUENCE:
: JSR PC,CRCR
:*****

```

```

CRCR: MOV R1,-(SP)
MOV R2,-(SP)
MOV #16.,R2
10$: CLC
ROR CWORD
ROR WORDT
BVC 20$
MOV #102010,R1
BIC CWORD,R1
BIC #102010,CWORD
BIS R1,CWORD
20$: DEC R2
BGT 10$
MOV (SP)+,R2
MOV (SP)+,R1
RTS PC

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

2175
 2176
 2177
 2178
 2179
 2180
 2181
 2182
 2183
 2184
 2185
 2186
 2187 004456 112777 000100 175756
 2188 004464 022737 000004 002472
 2189 004472 001003
 2190 004474 112777 000200 175740
 2191 004502 000240
 2192 004504 004737 004244
 2193 004510 000207
 2194
 2195
 2196
 2197
 2198
 2199
 2200
 2201
 2202
 2203
 2204
 2205
 2206
 2207
 2208
 2209
 2210
 2211
 2212
 2213
 2214
 2215
 2216
 2217 004512 012737 000003 002506
 2218 004520 000403
 2219 004522 012737 000005 002506
 2220 004530 112777 000100 175704
 2221 004536 022737 000004 002472
 2222 004544 001003
 2223 004546 112777 000200 175666
 2224 004554 000240
 2225 004556 000240
 2226 004560 012737 004600 002406
 2227 004566 004537 005070
 2228 004572 005737 002402
 2229 004576 100533
 2230 004600

```

*****
: FUNCTIONAL DESCRIPTION:   MINITR - MASTER CLEAR ROUTINE
: THIS ROUTINE ISSUES A MASTER CLEAR TO THE DEVICE
: IF OPTION IS AN 8206 IT ALSO SETS THE RUN BIT.
:
: SUBORDINATE ROUTINES USED:
:   WAIT50 - SHORT DELAY ROUTINE
: CALLING SEQUENCE:
:   JSR    PC,MINITR
:-----*****

```

```

MINITR:  MOVB  #100,@BSEL1      ;SET MASTER CLEAR.
         CMP   #04,OPTYP       ;IS THIS 8206
         BNE  MIN1R            ;BRANCH IF NOT
         MOVB  #200,@BSEL1     ;SET RUN
MIN1R:   NOP
1$:      JSR   PC,WAIT50       ;SHORT DELAY.
         RTS   PC              ;RETURN.

```

```

*****
: FUNCTIONAL DESCRIPTION:   MINITS - MASTER CLEAR AND INIT
: THIS ROUTINE ISSUES A MASTER CLEAR, WAITS FOR THE
: RUN BIT TO SET, CHECKS FOR GOOD COMPLETION OF MICRO
: DIAGNOSTICS AND ISSUES THE MODE DEFINITION.
: IF ENTERED AT MINIT1 - SET MODE TO FULL DUPLEX POINT
: TO POINT
: IF ENTERED AT MINITS - SET MODE TO FULL DUPLEX CONTROL
: IF ENTERED AT MINTR - SET MODE TO VALUE IN AXNUM
:
: OUTPUTS:   ERRWRD = -1 IF ERROR OCCURS.
:
: IMPLICIT OUTPUTS:
:   DMP EXITS WITH MODE DEFINED
:
: SUBORDINATE ROUTINES USED:
:   TOUT - TIME OUT ROUTINE
:   WAIT50 - SHORT DELAY ROUTINE
:   WRDO - WAIT FOR READY OUT
:
: CALLING SEQUENCE:
:   JSR    PC,MINITS          ;OR MINIT1 OR MINTR
:-----*****

```

```

MINIT1:  MOV   #03,AXNUM
         BR   MINTR
MINITS:  MOV   #05,AXNUM
MINTR:   MOVB  #100,@BSEL1     ;SET MASTER CLEAR.
         CMP   #4,OPTYP       ;IS THIS 8206
         BNE  MIN2R           ;SKIP IF NOT
         MOVB  #200,@BSEL1    ;SET RUN BIT
MIN2R:   NOP
         NOP
         MOV  #ERLB7,ERRADD   ;SET ERROR ADDRESS
TLB7:    JSR   R5,TOUT
         TST  ERRWRD
         BMI  MINTE           ;EXIT IF ERROR
ERLB7:

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2231 004600 004737 004244      4$:   JSR   PC, WAIT50
2232 004604 005777 175630      TST   @BSEL0      ;NOW WAIT FOR RUN TO ACTUALLY SET.
2233 004610 100366      BPL   TLB7
2234 004612 012737 004632 002406  MOV   #ERLB8, ERRADD ;SET ERROR ADDRESS
2235 004620 004537 005070      JSR   R5, TOUT    ;CHECK TIME OUT
2236 004624 005737 002402      TST   ERRWRD
2237 004630 100516      BMI   MINTE      ;EXIT IF ERROR
2238 004632      ERLB8:
2239 004632 122777 000305 175614  5$:   CMPB  #305, @BSEL6 ;GOOD END TO MICRO DIAG?
2240 004640 001404      BEQ   2$
2241 004642 122777 000264 175604  CMPB  #264, @BSEL6 ;LINE UNIT FAILURE.?
2242 004650 001363      BNE   TLB8      ;NO-STAY IN LOOP.
2243      ;YES, CATCH THE PROBLEM LATER.
2244 004652 012737 000077 002336  2$:   MOV   #77, $GDDAT
2245 004660 032737 000003 002472  BIT   #3, OPTYP   ;IS THIS DMV
2246 004666 001403      BEQ   3$         ;IF NOT GO TO 3
2247 004670 012737 000033 002336  MOV   #33, $GDDAT
2248 004676 023777 002336 175544  3$:   CMP   $GDDAT, @BSEL4
2249 004704 001407      BEQ   4$         ;IF CORRECT OPTYP CONTINUE
2250 004706 104457      TRAP  C$ERSOFT
2251 004710 000025      .WORD 21
2252 004712 012566      .WORD MEF32
2253 004714 010422      .WORD ERR32
2254 004716 005337 002402      DEC   ERRWRD
2255 004722 000461      BR    MINTE      ;EXIT TEST
2256 004724 032737 000003 002472  4$:   BIT   #3, OPTYP   ;IS THIS DMV
2257 004732 001423      BEQ   6$         ;IF NOT GO TO 6
2258 004734 005737 002400      TST   EXLOOP    ;DMV EXTERNAL LOOP FLAG SET?
2259 004740 001020      BNE   6$         ;IF YES: THEN GO TO 6
2260 004742 112777 000301 175472  MOVB  #301, @BSEL1
2261 004750 004537 002646      JSR   R5, WRDO   ;WAIT FOR READY OUT
2262 004754 012777 000006 175462  MOV   #06, @BSEL2 ;DO MAINT LOOP FOR TTL LOOPBACK
2263 004762 004537 005070      JSR   R5, TOUT
2264 004766 005737 002402      TST   ERRWRD
2265 004772 100435      BMI   MINTE
2266 004774 005777 175440      TST   @SEL0
2267 005000 100370      BPL   60$
2268 005002 152777 000010 175432  6$:   BISB  #BIT3, @BSEL1 ;SET LINE UNIT LOOP.-
2269      ;THIS ALLOWS US TO SET THE MODE.
2270 005010 052777 000200 175422      BIS   #RQ1, @BSEL0
2271 005016 012737 005036 002406  MOV   #ERLB9, ERRADD ;SET ERROR ADDRESS
2272 005024 004537 005070      JSR   R5, TOUT
2273 005030 005737 002402      TST   ERRWRD
2274 005034 100414      BMI   MINTE      ;EXIT IF ERROR
2275 005036 032777 000020 175400  ERLB9: BIT   #20, @BSEL2 ;WAIT FOR RDI.
2276 005044 001767      BEQ   TLB9
2277 005046 013777 002506 175400  MOV   AXNUM, @BSEL6 ;:SET UP FOR F/D CONTROL STATION
2278 005054 105077 175360      CLRB  @BSEL0     ;NO MORE REQUESTS.
2279 005060 012777 000002 175356  MOV   #2, @BSEL2 ;START.
2280 005066 000207      MINTE: RTS      PC
2281

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

2282
 2283
 2284
 2285
 2286
 2287
 2288
 2289
 2290
 2291 005070 020537 005136
 2292 005074 001011
 2293 005076 005237 005134
 2294 005102 001012
 2295 005104 104455
 2296 005106 000020
 2297 005110 011414
 2298 005112 010266
 2299 005114 005337 002402
 2300 005120 005037 005134
 2301 005124 010537 005136
 2302 005130
 2303 005130 104422
 2304 005132 000205
 2305
 2306 005134 000000
 2307
 2308 005136 000000
 2309

```

*****
:*****
: FUNCTIONAL DESCRIPTION:      TOUT  - TIME OUT ROUTINE
:                             THIS ROUTINE INC COUNTT LOCATION EVERY
:                             TIME IT IS CALLED IF COUNTT OVERFLOWS THEN
:                             TIME OUT IS REPORTED AND THE ROUTINE IS EXITED.
: CALLING SEQUENCE:
:                             JSR    R5,TOUT
:*****
TOUT:  CMP    R5,LA5TR5
       BNE   TOUTE
       INC  COUNTT
       BNE  TOUTEX
       TRAP C$ERDF
       .WORD 16
       .WORD MEF7
       .WORD ERR26
TOUTE: DEC  ERRWRD
       CLR  COUNTT
       MOV  R5,LA5TR5      ;SAVE CURRENT PC.
TOUTEX: TRAP C$BRK
       RTS  R5            ;EXIT
COUNTT: 0 ;NUMBERS OF TIMES IN THIS ROUTINE FROM
          ;SAME CALLING LOCATION.
LA5TR5: 0 ;LAST CALLING LOCATION.

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324 005140
2325 005140 013746 002440
2326 005144 013746 002332
2327 005150 012746 005306
2328 005154 012746 000003
2329 005160 010600
2330 005162 104415
2331 005164 062706 000010
2332 005170 017746 175260
2333 005174 017746 175250
2334 005200 017746 175240
2335 005204 017746 175230
2336 005210 012746 005373
2337 005214 012746 000005
2338 005220 010600
2339 005222 104415
2340 005224 062706 000014
2341 005230 032737 000003 002472
2342 005236 001412
2343 005240 017746 175214
2344 005244 012746 005451
2345 005250 012746 000002
2346 005254 010600
2347 005256 104415
2348 005260 062706 000006
2349 005264
2350 005264 012746 005466
2351 005270 012746 000001
2352 005274 010600
2353 005276 104415
2354 005300 062706 000004
2355 005304 000207
2356 005306 047045 040445 040506
2357 005314 046111 047111 020107
2358 005322 041515 052520 044440
2359 005330 020123 047125 052111
2360 005336 021440 047445 022462
2361 005344 020101 044127 051517
2362 005352 020105 042101 051104
2363 005360 051505 020123 051511
2364 005366 022440 033117 000
2365 005373 045 022516 051501

```

```

*****
: FUNCTIONAL DESCRIPTION: STAND - PRINT STANDARD REGS
: THIS ROUTINE PRINTS THE UNIT NUMBER AND
: CSR ADDRESS OF THE FAILING UNIT AS WELL AS THE
: CONTENTS OF ALL THE CSR REGS.
: THE ERROR MSG ROUTINES USE THIS SUBROUTINE
:
: IMPLICIT INF'TS:
: CSRS' - THE CSR ARE EXPECTED TO CONTAIN USEFUL DATA
: LOGDEV - THE LOGICAL DEVICE NUMBER
: SELO - ADDRESS OF THIS UNIT
:
: CALLING SEQUENCE:
: JSR PC,STAND
:-----*****

```

```

STAND:
MOV SELO,-(SP)
MOV LOGDEV,-(SP)
MOV #CFM1,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #10,SP
MOV @SEL6,-(SP)
MOV @SEL4,-(SP)
MOV @SEL2,-(SP)
MOV @SELO,-(SP)
MOV #CFM2,-(SP)
MOV #5,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #14,SP
BIT #3,OPTYP ;IS THIS A DMV-11 ?
BEQ 1$ ;IF NOT: SKIP SEL10 PRINTOUT
MOV @SEL10,-(SP)
MOV #CFM3,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #6,SP

1$:
MOV #CFM4,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTX
ADD #4,SP
RTS PC
CFM1: .ASCIZ '%N%AFAILING MCPU IS UNIT #%02%A WHOSE ADDRESS IS %06%'

CFM2: .ASCIZ '%N%ASELO=%06%A SEL2=%06%A SEL4=%06%A SEL6=%06%'

```


CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

2366	005400	046105	036460	047445	
2367	005406	022466	020101	042523	
2368	005414	031114	022475	033117	
2369	005422	040445	051440	046105	
2370	005430	036464	047445	022466	
2371	005436	020101	042523	033114	
2372	005444	022475	033117	000	
2373	005451	045	020101	042523	CFM3: .ASCIZ '%A SEL10=%06''
2374	005456	030514	036460	047445	
2375	005464	000066			
2376	005466	047045	000		CFM4: .ASCIZ '%N''
2377		005472			.EVEN
2378					
2379					

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2380 :*****
2381 : FUNCTIONAL DESCRIPTION:      RMVRT - VERIFY ROM CONTENTS
2382 :           THIS ROUTINE READS DMV ROMS USING 256 BYTE READS
2383 :           AND CALCULATES/CHECKS THE CRC.
2384 : SUBORDINATE ROUTINES:
2385 :           WRDO - WAIT FOR READY OUT (MRDY)
2386 :           CRCR - CALCULATE CRC
2387 : IMPLICIT INPUTS:
2388 :           ROMADD - STARTING ADDRESS OF PARTICULAR ROM
2389 : CALLING SEQUENCE:
2390 :           JSR    R5,RMVRT
2391 :*****
2392
2393 005472 012737 177777 002374 RMVRT:  MOV    #-1,CWORD      ;INIT CRC WORD
2394 005500 112777 000301 174734      MOVB   #301,@BSEL1    ;*ENTER MAINTENANCE LOOP
2395 005506 004537 002646          JSR    R5,WRDO        ;*
2396 005512 005737 002402          TST   ERRWRD         ;*
2397 005516 100477          BMI   RMVEX         ;*EXIT IF ERROR
2398 005520 005003          CLR   R3            ;CLEAR BLOCK_NUMBER
2399
2400 00552. 012777 033734 174724 RMVXX:  MOV    #RECBU1,@SEL6   ;READ 256 BYTES USING M-LOOP
2401 005530 105077 174724          CLRB  @BSEL10        ;RCV BUFFER ADDR => BSR10:CSR6
2402 005534 013777 002436 174706          MOV   ROMADD,@SEL4   ;ROM ADDRESS => CSR4
2403 005542 112777 000003 174674          MOVB  #03,@BSEL2     ;*DO BLOCK READ OF 256 BYTES
2404 005550 004537 002646          JSR   R5,WRDO        ;*WAIT FOR MRDY BIT (RDO)
2405 005554 005737 002402          TST   ERRWRD         ;*
2406 005560 100456          BMI   RMVEX         ;*EXIT IF ERROR
2407
2408          CLR   R2            ;CLEAR WORD_INDEX
2409          TST   R3            ;IS THIS THE 1ST BLOCK?
2410          BNE   RMVYY        ;
2411          MOV   RECBU1,CRCCAL ;YES: SAVE 1ST WORD OF 1ST BLOCK
2412          TST   (R2)+        ;      (CRC CHARACTER)
2413          MOVB  RECBU1(R2),ROMNO ;SAVE ROM #
2414          INC   R2            ;
2415          MOVB  RECBU1(R2),REVNO ;SAVE REVISION #
2416          DEC   R2            ;ADJUST INDEX FOR ROM#
2417
2418          MOV   RECBU1(R2),WORDT ;GET INDEXED WORD FROM BUFFER
2419          JSR   PC,CRCR      ;CALCULATE CRC WORD
2420          ADD   #2,R2        ;(BUMP INDEX)
2421          CMP   #256.,R2    ;IS THIS THE LAST WORD ?
2422          BNE   RMVYY        ;IF NOT: GET NEXT WORD.
2423
2424          CMP   R3,#31.     ;IS THIS THE LAST 256 WORD BLOCK?
2425          BEQ   RMVBB        ;
2426          ADD   #256.,ROMADD ;NO: ADD 256 TO ADDRESS
2427          INC   R3            ;      AND BUMP BLOCK NUMBER
2428          BR   RMVXX        ;      AND GO GET SOME MORE
2429
2430          NOP                ;(COM CWORD=5137 2374)
2431          NOP                ;YES:
2432          MOV   CRCCAL,WORDT ;
2433          CMP   CWORD,WORDT ;
2434          BEQ   RMVEX        ; COMPARE CRC WORDS...
2435          TRAP  C$ERDF       ; AND REPORT ERROR IF NO MATCH.

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

2436 005710 000021
2437 005712 000000
2438 005714 010162
2439
2440 005716 000205
2441

.WORD 17
.WORD 0
.WORD ERR24

RMVEX: RTS R5

;RETURN TO CALLER

CZDMTC.P11

25-MAR-81 08:24

GLOBAL SUBROUTINES

2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
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
2490
2491
2492
2493
2494
2495
2496
2497

```

*****
:*****
: FUNCTIONAL DESCRIPTION:      TXRXSR - TRANSMIT RECEIVE SUBROUTINE
: THIS ROUTINE IS USED BY ALL TESTS THAT TRANSMIT
: AND RECIEVE DATA. THE FIRST PART OF THE ROUTINE VERIFIES
: THE OPERATOR INPUTS AND MAKES SURE THAT INTERFACE
: SELECTION CORRESPOND TO SELECTED BAUD RATES.
: THE SECOND PART OF THE ROUTINE FORMS 'TXRX3' AND DOES THE
: FOLLOWING. ESTABLISH TRIBUTARY. THEN EITHER DO ISTRT OR
: MAINT STATE DEPENDING ON FLAG. IF ISTRT THEN CHECK FOR
: RUN STATE IF MAINT STATE THEN GO TO NEXT STEP. NEXT QUE
: REC AND TRANSMIT BUFFERS THEN WAIT FOR OUTPUT. IF MEMORY
: MANAGEMENT EXIT TEST. IF ISTRT LOOK FOR REC COMPLETED
: FOLLOWED BY TX COMPLETED. IF MAINT LOOK FOR TRANSMIT FIRST.
:
: INPUTS:      TXADD - ADDRESS OF TRANSMIT BUFFER
:              TXCC  - CHAR COUNT OF TX
:              RXADD - ADDRESS OF REC BUFFER
:              RXCC  - CHAR COUNT OF REC BUFFER
:              GENWRD - FLAG WORD IF BIT 15 SET-MAINT MODE
:                  IF BIT 14 SET THEN MEMORY MGT.
:
: SUBORDINATE ROUTINES USED:
:              WAIT50 - SHORT DELAY
:              CONTIN - CONTROL IN ROUTINE
:              GETOUT - GET CONTROL OUT CODE
:              GETOC  - GET OUTPUT CODE
:              WRDI   - WAIT FOR READY IN
:              GETCC  - GET CHAR COUNT
:              GETBA  - GET BUFFER ADDRESS
:
: CALLING SEQUENCE:
:              JSR      R5,TXRXSR      ;IF NOT INTERFACE CHECK
:                               ;CALL AT JSR      R5,TXRX3
:*****

```

```

005720 022737 000004 002476 TXRXSR: CMP      #4,TSTCON
005726 001002          BNE      10$          ;IF INTERNAL LOOP
005730 000137 006346          JMP      TXRX3        ;GO TO 3
:
: JUMP TO TXRX 3 IF INTERNAL LOOP
005734 013737 002360 002506 10$:  MOV      TRIBN,AXNUM ;SAVE TRIBN.
005742 005037 002360          CLR      TRIBN      ;MAKE TRIBN 0
:DMV-11 WILL IGNORE THE FOLLOWING...
005746 032737 000002 002476 TXRX1: BIT      #BIT1,TSTCON ;IS IT MODEM LOOP BACK?
005754 001571          BEQ      TXRX2        ;IF NOT THEN GO TO 2
005756 012704 000110          MOV      #110,R4
005762 032737 000001 002476          BIT      #BIT0,TSTCON ;IS IT REMOTE MODEM
005770 001402          BEQ      TXRX1A       ;IF NOT THEN GO TO 1A
005772 012704 000104          MOV      #104,R4
:
TXRX1A: MOV      #21,R3
005776 012703 000021          JSR      R5,CONTIN    ; WRITE MODEM WITH CORRECT
006002 004537 003162          ; TYPE OF LOOP CODE
006006 005737 002402          TST      ERRWRD
006012 100002          BPL      15$

```


CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2610
2611 006374 012703 000003      MOV    #03,R3      ;ELSE;
2612 006400 004537 003162      JSR    R5,CONTIN  ;ISTR THE TRIB
2613                                ; TIME OUT OR READY ERRORS REPORT THIS PC
2614 006404 005737 002402      TST    ERRWRD
2615 006410 100432      BMI    TXRXA      ;EXIT IF ERROR OCCURRED
2616
2617                                ;CHECK FOR RUN STATE
2618
2619 006412 012737 000001 002336  MOV    #01,$GDDAT  ;CHECK FOR CONTROL OUT
2620 006420 004537 003242      JSR    R5,GETOUT  ; AND CORRECT TRIBN
2621 006424 005737 002402      TST    ERRWRD
2622 006430 100422      BMI    TXRXA      ;EXIT IF ERROR OCCURRED
2623 006432 012737 000024 002336  MOV    #24,$GDDAT  ;CHECK FOR RUN STATE
2624 006440 004537 003670      JSR    R5,GETOC   ; IN OUTPUT CODE
2625 006444 005737 002402      TST    ERRWRD
2626 006450 100412      BMI    TXRXA      ;EXIT IF ERROR OCCURRED
2627 006452 042777 000200 173764  BIC    #RDO,@BSEL2 ;CLEAR OUTPUT
2628 006460 000411      BR     TXRXC      ;AND GO TO 20
2629
2630                                ; PUT TRIB IN MAINT STATE
2631
2632 006462 012703 000004      TXRXB: MOV    #04,R3      ;PUT TRIB IN MAINT STATE
2633 006466 004537 003162      JSR    R5,CONTIN  ;
2634                                ; TIME OUT OR READY ERRORS REPORT THIS PC
2635 006472 005737 002402      TST    ERRWRD
2636 006476 100002      TXRXA: BPL    TXRXC
2637 006500 000137 007344      JMP    TXRXEN     ;EXIT IF ERROR OCCURRED
2638
2639                                ;QUEUE RECEIVE BUFFER
2640
2641 006504 052777 000200 173726  TXRXC: BIS    #RQI,@BSEL0 ;SET REQUEST
2642 006512 004537 002732      JSR    R5,WRDI
2643                                ; TIME OUT OR READY ERROR REPORT THIS PC
2644 006516 005737 002402      TST    ERRWRD
2645 006522 100002      BPL    5$
2646 006524 000137 007344      JMP    TXRXEN     ;EXIT IF ERROR OCCURRED
2647 006530 013777 002422 173712  5$:  MOV    RXADD,@BSEL4 ;SET ADDRESS
2648 006536 113777 002360 173702  MOVB   TRIBN,@BSEL3 ;SET TRIBN
2649
2650 006544 005737 002376      TST    MODQ22
2651 006550 001411      BEQ    1$
2652                                ;*IS THIS 'Q22 MODE' ?
2653                                ;*
2654 006552 013777 002424 173700  MOV    RXCC,@BSEL10 ;*YES: SET CHARACTER COUNT
2655 006560 005077 173670      CLR    @BSEL6     ;* CLEAR EXTENDED ADDR BITS
2656 006564 112777 000010 173652  MOVB   #10,@BSEL2 ;* SET RX BUFFER IN (+Q22 BIT)
2657 006572 000406      BR     2$
2658                                ;* AND CONTINUE...
2659 006574 013777 002424 173652  1$:  MOV    RXCC, @BSEL6 ;*NO: SET CHAR COUNT [+BA16/17]
2660 006602 112777 000000 173634  MOVB   #0, @BSEL2 ; SET RX BUFFER IN
2661
2662                                ;QUEUE TX BUFFER
2663 006610 004537 002732      2$:  JSR    R5,WRDI
2664                                ;WAIT FOR READY
2665 006614 005737 002402      TST    ERRWRD
2666                                ; TIME OUT OR READY ERROR REPORT THIS PC
2667                                ;

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2666 006620 100002          BPL      35$          ;*
2667 006622 000137 007344  JMP      TXRXEN      ;*EXIT IF ERROR OCCURRED
2668 006626 013777 002420 173614 35$:  MOV     TXADD,@BSEL4 ;SET TX ADD
2669 006634 113777 002360 173604  MOVB   TRIBN,@BSEL3 ;SET TRIBN
2670
2671 006642 005737 002376          TST     MODQ22      ;*IS THIS 'Q22 MODE' ?
2672 006646 001414          BEQ     3$          ;*
2673
2674 006650 013777 002426 173602  MOV     TXCC,@BSEL10 ;*YES: SET CHARACTER COUNT
2675 006656 005077 173572          CLR     @BSEL6      ;* CLEAR EXTENDED ADDR BITS
2676 006662 042777 000200 173550  BIC     #RQI,@BSEL0 ;* CLEAR REQUEST
2677 006670 112777 000014 173546  MOVB   #14,@BSEL2  ;* SET UP TX BUFFER (+Q22 BIT)
2678 006676 000411          BR      4$          ;* AND CONTINUE...
2679
2680 006700 013777 002426 173546 3$:  MOV     TXCC,@BSEL6 ;*NO: SET CHAR COUNT [+BA16/17]
2681 006706 042777 000200 173524  BIC     #RQI,@BSEL0 ; CLEAR REQUEST
2682 006714 112777 000004 173522  MOVB   #04,@BSEL2  ; SET UP TX BUFFER
2683
2684 006722 032737 040000 002432 4$:  BIT     #BIT14,GENWRD
2685 006730 001402          BEQ     20$         ;IF MM GO TO RETURN
2686 006732 000137 007344          JMP     TXRXEN      ;* CLEAR LOCAL_MAINT_FLAG
2687 006736 005037 007346 20$:  CLR     MFLG
2688 006742 005737 002432          TST     GENWRD
2689 006746 100052          BPL     TXRXG      ;GO AHEAD IF NOT MAINT STATE
2690
2691          ;CHECK FOR RX COMPLETED
2692
2693          ; NOTE: IF MAINT STATE THEN CHECK FOR RX AND TX BUFFERS
2694          ; RETURNED (ORDER NOT IMPORTANT).
2695          ; BUT: IF NOT MAINT STATE THEN RX BUFFER MUST BE RETURNED
2696          ; FIRST AND TX BUFFER SECOND (OR ERROR REPORTED).
2697          ; * IF WE ARE IN MAINT STATE ....
2698 006750 004537 002646          JSR     R5,WRDO    ;* WAIT FOR READY_OUT
2699 006754 005737 002402          TST     ERRWRD
2700 006760 100571          BMI     TXRXEN      ;* EXIT IF ERROR
2701 006762 117704 173456          MOVB   @BSEL2,R4   ;* GET COMMAND TYPE
2702 006766 042704 177770          BIC     #177770,R4 ;* STRIP EXCESS BITS
2703 006772 022704 000004          CMP     #4,R4      ;* CHECK FOR TX BUFFER RETURNED
2704 006776 001403          BEQ     40$        ;* IF YES: GO CHECK IT
2705 007000 005337 007346          DEC     MFLG
2706 007004 000433          BR      TXRXG      ;* NO: MFLG=(-1) TO INDICATE TXBUF
2707          ;* RETURNED FIRST
2708 007006 012737 000004 002336 40$:  MOV     #04,$GDDAT
2709 007014 004537 003242          JSR     R5,GETOUT  ;GET OUTPUT CODE
2710 007020 005737 002402          TST     ERRWRD
2711 007024 100547          BMI     TXRXEN      ;EXIT IF ERROR
2712 007026 013737 002426 002336  MOV     TXCC,$GDDAT ;*
2713 007034 004537 003514          JSR     R5,GETCC   ;*ELSE CHECK TX CHAR COUNT
2714 007040 005737 002402          TST     ERRWRD
2715 007044 100537          BMI     TXRXEN      ;*EXIT IF ERROR
2716 007046 013737 002420 002336  MOV     TXADD,$GDDAT
2717 007054 004537 003576          JSR     R5,GETBA   ;IS THE TX BUFFER ADDR RIGHT ?
2718 007060 005737 002402          TST     ERRWRD
2719 007064 100527          BMI     TXRXEN      ;EXIT IF ERROR OCCURED
2720
2721 007066 042777 000200 173350  BIC     #RDO,@BSEL2 ;CLEAR READY OUT

```


CZDMTC.P11

25-MAR-81 08:24

GLOBAL SUBROUTINES

```

2722 007074 012737 000000 002336 TXRXG: MOV #0, $GDDAT ;
2723 007102 004537 003242 JSR R5,GETOUT ;CHECK FOR RX RETURNED
2724 007106 005737 002402 TST ERRWRD ;
2725 007112 100514 BMI TXRXEN ;EXIT IF ERROR OCCURRED
2726 007114 013737 002426 002336 MOV TXCC,$GDDAT ;
2727 007122 004537 003514 JSR R5,GETCC ;IS THE CHAR COUNT CORRECT
2728 007126 005737 002402 TST ERRWRD ;
2729 007132 100504 BMI TXRXEN ;EXIT IF ERROR OCCURRED
2730 007134 013737 002422 002336 MOV RXADD,$GDDAT ;
2731 007142 004537 003576 JSR R5,GETBA ;IS THE BUFFER ADD RIGHT
2732 007146 005737 002402 TST ERRWRD ;
2733 007152 100474 BMI TXRXEN ;EXIT IF ERROR OCCURRED
2734 ;*****
2735 ;DATA CHECK....
2736 007154 013703 002422 MOV RXADD,R3
2737 007160 013701 002420 MOV TXADD,R1 ;SET UP ADDRESS
2738 007164 005004 CLR R4 ;CLEAR R4
2739 007166 25$:
2740 007166 112337 002340 26$: MOV (R3)+,$BDDAT ;GET BYTE OF RX
2741 007172 112137 002336 28$: MOV (R1)+,$GDDAT ;GET BYTE OF TX
2742 007176 123737 002340 002336 CMPB $BDDAT,$GDDAT ;ARE THEY THE SAME
2743 007204 001411 BEQ 30$ ;IF SO GO TO 30
2744 007206 005204 INC R4 ;MAKE COUNT RIGHT
2745 007210 104455 TRAP C$ERDF
2746 007212 000022 .WORD 18
2747 007214 012337 .WORD MEF19A
2748 007216 007646 .WORD ERR19
2749 007220 005337 002402 DEC ERRWRD
2750 007224 005304 DEC R4 ;MAKE COUNT RIGHT
2751 007226 000446 BR TXRXEN ;EXIT IF ERROR
2752 007230 005204 30$: INC R4 ;BUMP TO NEXT BYTE
2753 007232 020437 002426 CMP R4,TXCC ;ARE WE DONE?
2754 007236 103753 BLO 25$ ;IF NOT GO BACK
2755 ;*****
2756 007240 005737 007346 TST MFLG ;* CHECK LOCAL_MAINT_FLAG
2757 007244 001401 BEQ 31$ ;* IF CLEARED: GOTO 31$
2758 007246 000403 BR 32$ ;* SET: CHECK EXPECTED TXBUFF RETURN
2759
2760 007250 005737 002432 31$: TST GENWRD ;TEST FOR MAINT STATE
2761 007254 100433 BMI TXRXEN ;RETURN TO CALLER IF MAINT STATE
2762 007256 042777 000200 173160 32$: BIC #RDO,@BSEL2 ;CLEAR OUTPUT
2763 007264 012737 000004 002336 MOV #4,$GDDAT
2764 007272 004537 003242 JSR R5,GETOUT ;CHECK FOR TX BUFF COMP
2765 007276 005737 002402 TST ERRWRD ;
2766 007302 100420 BMI TXRXEN ;*IF ERROR: THEN EXIT
2767 007304 013737 002426 002336 MOV TXCC,$GDDAT ;
2768 007312 004537 003514 JSR R5,GETCC ;*ELSE CHECK TX CHAR COUNT
2769 007316 005737 002402 TST ERRWRD ;
2770 007322 100410 BMI TXRXEN ;*EXIT IF ERROR
2771 007324 013737 002420 002336 MOV TXADD,$GDDAT
2772 007332 004537 003576 JSR R5,GETBA ;IS THE TX BUFFER ADDR RIGHT ?
2773 007336 005737 002402 TST ERRWRD ;
2774 007342 100400 BMI TXRXEN ;EXIT IF ERROR OCCURED
2775 007344 000205 TXRXEN: RTS R5
2776
2777 007346 000000 MFLG: 0 ;* LOCAL_MAINT_FLAG

```

CZDMTC.P11

25-MAR-81 08:24

GLOBAL SUBROUTINES

2778
2779

:* (IF MAINT MODE + MTFLG SET THEN TXBUF
:* RETURN EXPECTED AFTER RXBUF RETURN)

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

.SBTTL GLOBAL ERROR REPORT SECTION

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

2780
2781
2782
2783
2784
2785
2786
2787 007350
2788 007350 013746 002440
2789 007354 012746 010675
2790 007360 012746 000002
2791 007364 010600
2792 007366 104414
2793 007370 062706 000006
2794 007374
2795 007374 104423
2796
2797
2798
2799
2800 007376
2801 007376 017746 173052
2802 007402 012746 011105
2803 007406 012746 000002
2804 007412 010600
2805 007414 104414
2806 007416 062706 000006
2807 007422 004737 005140
2808 007426
2809 007426 104423
2810
2811 007430
2812 007430 013746 002340
2813 007434 013746 002336
2814 007440 012746 011243
2815 007444 012746 000003
2816 007450 010600
2817 007452 104414
2818 007454 062706 000010
2819 007460 004737 005140
2820 007464
2821 007464 104423
2822
2823 007466
2824 007466 013746 002340
2825 007472 013746 002336
2826 007476 012746 011336
2827 007502 012746 000003
2828 007506 010600
2829 007510 104414
2830 007512 062706 000010
2831 007514 004737 005140
2832 007522
2833 007522 104423
2834
2835

ERR1::
MOV SELO,-(SP)
MOV #MEF1,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #6,SP

L10003:
TRAP C\$MSG

;FAILING CODE

ERR3::
MOV @BSEL6,-(SP)
MOV #MEF3,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #6,SP
JSR PC,STAND

L10004:
TRAP C\$MSG

ERR5::
MOV \$BDDAT,-(SP)
MOV \$GDDAT,-(SP)
MOV #MEF5,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #10,SP
JSR PC,STAND

L10005:
TRAP C\$MSG

ERR6::
MOV \$BDDAT,-(SP)
MOV \$GDDAT,-(SP)
MOV #MEF6,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #10,SP
JSR PC,STAND

L10006:
TRAP C\$MSG

;PRINT FAILED TO SET RDI

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

```

2836
2837 007524
2838 007524 012746 011660
2839 007530 012746 000001
2840 007534 010600
2841 007536 104414
2842 007540 062706 000004
2843 007544 004737 005140
2844 007550
2845 007550 104423
2846
2847
2848
2849 007552
2850 007552 012746 011704
2851 007556 012746 000001
2852 007562 010600
2853 007564 104414
2854 007566 062706 000004
2855 007572 004737 005140
2856 007576
2857 007576 104423
2858
2859
2860
2861
2862
2863 007600
2864 007600 013746 002406
2865 007604 012746 011611
2866 007610 013746 002340
2867 007614 013746 002336
2868 007620 012746 012301
2869 007624 012746 000005
2870 007630 010600
2871 007632 104414
2872 007634 062706 000014
2873 007640 004737 005140
2874 007644
2875 007644 104423
2876
2877
2878
2879 007646
2880 007646 013746 002340
2881 007652 013746 002336
2882 007656 010446
2883 007660 012746 012372
2884 007664 012746 000004
2885 007670 010600
2886 007672 104414
2887 007674 062706 000012
2888 007700 004737 005140
2889 007704
2890 007704 104423
2891

```

```

ERR9::
      MOV    #MRDI,-(SP)
      MOV    #1,-(SP)
      MOV    SP,R0
      TRAP   C$PNTB
      ADD    #4,SP
      JSR    PC,STAND
L10007:
      TRAP   C$MSG
      ;PRINT FAILED TO SET RDO

ERR10::
      MOV    #MRDO,-(SP)
      MOV    #1,-(SP)
      MOV    SP,R0
      TRAP   C$PNTB
      ADD    #4,SP
      JSR    PC,STAND
L10010:
      TRAP   C$MSG
      ;PRINTS GOOD AND BAD DATA (BYTES) AND
      ;FAILING PC ADDRS AND STANDARD REGS

ERR18::
      MOV    ERRADD,-(SP)
      MOV    #MFPC,-(SP)
      MOV    $BDDAT,-(SP)
      MOV    $GDDAT,-(SP)
      MOV    #MEF18,-(SP)
      MOV    #5,-(SP)
      MOV    SP,R0
      TRAP   C$PNTB
      ADD    #14,SP
      JSR    PC,STAND
L10011:
      TRAP   C$MSG
      ;DATA COMPARE ERROR

ERR19::
      MOV    $BDDAT,-(SP)
      MOV    $GDDAT,-(SP)
      MOV    R4,-(SP)
      MOV    #MEF19,-(SP)
      MOV    #4,-(SP)
      MOV    SP,R0
      TRAP   C$PNTB
      ADD    #12,SP
      JSR    PC,STAND
L10012:
      TRAP   C$MSG

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

;NON-EXISTENT MEMORY ERROR

2892
 2893
 2894 007706
 2895 007706 013746 002542
 2896 007712 012746 012633
 2897 007716 012746 000002
 2898 007722 010600
 2899 007724 104414
 2900 007726 062706 000006
 2901 007732 004737 005140
 2902 007736
 2903 007736 104423
 2904
 2905 007740
 2906 007740 012746 012756
 2907 007744 012746 000001
 2908 007750 010600
 2909 007752 104414
 2910 007754 062706 000004
 2911 007760 013746 002340
 2912 007764 013746 002336
 2913 007770 013746 002542
 2914 007774 012746 013045
 2915 010000 012746 000004
 2916 010004 010600
 2917 010006 104414
 2918 010010 062706 000012
 2919 010014 004737 005140
 2920 010020
 2921 010020 104423
 2922
 2923 010022
 2924 010022 012746 013126
 2925 010026 012746 000001
 2926 010032 010600
 2927 010034 104414
 2928 010036 062706 000004
 2929 010042 013746 002542
 2930 010046 012746 013214
 2931 010052 012746 000002
 2932 010056 010600
 2933 010060 104414
 2934 010062 062706 000006
 2935 010066 012746 013303
 2936 010072 012746 000001
 2937 010076 010600
 2938 010100 104414
 2939 010102 062706 000004
 2940 010106 004737 005140
 2941 010112
 2942 010112 104423
 2943
 2944
 2945
 2946
 2947

ERR20::
 MC / \$TMP0,-(SP)
 MOV #TFM20,-(SP)
 MOV #2,-(SP)
 MOV SP,R0
 TRAP C\$PNTB
 ADD #6,SP
 JSR PC,STAND
 L10013:
 TRAP C\$MSG
 ERR21::
 MOV #TFM21,-(SP)
 MOV #1,-(SP)
 MOV SP,R0
 TRAP C\$PN 3
 ADD #4,SP
 MOV \$BDDAT,-(SP)
 MOV \$GDDAT,-(SP)
 MOV \$TMP0,-(SP)
 MOV #TFM2A,-(SP)
 MOV #4,-(SP)
 MOV SP,R0
 TRAP C\$PNTB
 ADD #12,SP
 JSR PC,STAND
 L10014:
 TRAP C\$MSG
 ERR22::
 MOV #TFM22,-(SP)
 MOV #1,-(SP)
 MOV SP,R0
 TRAP C\$PNTB
 ADD #4,SP
 MOV \$TMP0,-(SP)
 MOV #TFM22A,-(SP)
 MOV #2,-(SP)
 MOV SP,R0
 TRAP C\$PNTB
 ADD #6,SP
 MOV #TFM22B,-(SP)
 MOV #1,-(SP)
 MOV SP,R0
 TRAP C\$PNTB
 ADD #4,SP
 JSR PC,STAND
 L10015:
 TRAP C\$MSG

;PRINTS GOOD AND BAD DATA (WORDS) AND
 ;FAILING PC ADDRS AND STANDARD REGS

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

```

2948 010114
2949 010114 013746 002406
2950 010120 012746 011611
2951 010124 013746 002340
2952 010130 013746 002336
2953 010134 012746 012460
2954 010140 012746 000005
2955 010144 010600
2956 010146 104414
2957 010150 062706 000014
2958 010154 004737 005140
2959 010160
2960 010160 104423
2961
2962 010162
2963 010162 013746 002374
2964 010166 013746 002372
2965 010172 013746 002366
2966 010176 012746 013367
2967 010202 012746 000004
2968 010206 010600
2969 010210 104414
2970 010212 062706 000012
2971 010216 004737 005140
2972 010222
2973 010222 104423
2974
2975 010224
2976 010224 013746 002372
2977 010230 013746 002370
2978 010234 013746 002366
2979 010240 012746 013461
2980 010244 012746 000004
2981 010250 010600
2982 010252 104414
2983 010254 062706 000012
2984 010260 004737 005140
2985 010264
2986 010264 104423
2987
2988
2989
2990 010266
2991 010266 013746 002406
2992 010272 012746 011611
2993 010276 012746 011603
2994 010302 012746 000003
2995 010306 010600
2996 010310 104414
2997 010312 062706 000010
2998 010316 004737 005140
2999 010322
3000 010322 104423
3001
3002
3003

```

```

ERR23::
MOV ERRADD,-(SP)
MOV #MFPC,-(SP)
MOV $BDDAT,-(SP)
MOV $GDDAT,-(SP)
MOV #MEF23,-(SP)
MOV #5,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #14,SP
JSR PC,STAND

```

```

L10016: TRAP C$MSG

```

```

ERR24::
MOV CWORD,-(SP)
MOV WORDT,-(SP)
MOV ROMN,-(SP)
MOV #TFM24,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #12,SP
JSR PC,STAND

```

```

L10017: TRAP C$MSG

```

```

ERR25::
MOV WORDT,-(SP)
MOV ROMN1,-(SP)
MOV ROMN,-(SP)
MOV #TFM25,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #12,SP
JSR PC,STAND

```

```

L10020: TRAP C$MSG

```

```

;PRINTS FAILING PC ADDRESS AND STANARD REGS

```

```

ERR26::
MOV ERRADD,-(SP)
MOV #MFPC,-(SP)
MOV #MEF1A,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #10,SP
JSR PC,STAND

```

```

L10021: TRAP C$MSG

```

```

;PRINTS FAILING PC ADDRESS AND
;CODE IN ERROR FROM CODEW AND

```

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

```

3004
3005
3006 010324
3007 010324 013746 002406
3008 010330 012746 011611
3009 010334 012746 011603
3010 010340 012746 000003
3011 010344 010600
3012 010346 104414
3013 010350 062706 000010
3014 010354 013746 002340
3015 010360 013746 002336
3016 010364 012746 011730
3017 010370 013746 002430
3018 010374 012746 011536
3019 010400 012746 000005
3020 010404 010600
3021 010406 104414
3022 010410 062706 000014
3023 010414 004737 005140
3024 010420
3025 010420 104423
3026
3027
3028
3029 010422
3030 010422 012746 010450
3031 010426 012746 000001
3032 010432 010600
3033 010434 104414
3034 010436 062706 000004
3035 010442 004737 005140
3036 010446
3037 010446 104423
3038
3039 010450 040445 040502 044523
3040 010456 020103 051105 047522
3041 010464 000122
3042 010466 047045 040445 044440
3043 010474 041516 051117 042522
3044 010502 052103 044440 052116
3045 010510 051105 040506 042503
3046 010516 043040 051117 047440
3047 010524 052120 047511 020116
3048 010532 042523 042514 052103
3049 010540 042105 000040
3050 010544 047045 040445 044440
3051 010552 041516 051117 042522
3052 010560 052103 041040 052501
3053 010566 020104 040522 042524
3054 010574 043040 051117 044440
3055 010602 052116 051105 040506
3056 010610 042503 051440 046105
3057 010616 041505 042524 000104
3058 010624 047045 040445 025040
3059 010632 025052 025052 020052

```

;STANDARD REGISTERS

ERR27::

```

MOV ERRADD,-(SP)
MOV #MFPC,-(SP)
MOV #MEF1A,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #10,SP
MOV $BDDAT,-(SP)
MOV $GDDAT,-(SP)
MOV #MGB,-(SP)
MOV CODEW,-(SP)
MOV #MEF11,-(SP)
MOV #5,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #14,SP
JSR PC,STAND

```

L10022:

TRAP C\$MSG

;PRINTS THE STANDARD REGS

ERR32::

```

MOV #BASER,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTB
ADD #4,SP
JSR PC,STAND

```

L10023:

TRAP C\$MSG

BASER: .ASCIZ '%ABASIC ERROR''

BADIFM: .ASCIZ '%N% INCORRECT INTERFACE FOR OPTION SELECTED ''

BADBRM: .ASCIZ '%N% INCORRECT BAUD RATE FOR INTERFACE SELECTED''

TESTAB: .ASCIZ '%N% ***** SUBTEST %02% ABORTED ***** ''

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

3060	010640	052523	052102	051505	
3061	010646	020124	047445	022462	
3062	010654	020101	041101	051117	
3063	010662	042524	020104	025052	
3064	010670	025052	020052	000	
3065					
3066	010675	045	020101	042101	MEF1: .ASCIZ '%A ADDRESSING PROBLEM UNIT ADDRESS %06%N'
3067	010702	051104	051505	044523	
3068	010710	043516	050040	047522	
3069	010716	046102	046505	052440	
3070	010724	044516	020124	042101	
3071	010732	051104	051505	020123	
3072	010740	047445	022466	000116	
3073	010746	042101	051104	051505	EMT0: .ASCIZ /ADDRESS ERROR -TRAP 4/
3074	010754	020123	051105	047522	
3075	010762	020122	052055	040522	
3076	010770	020120	000064		
3077	010774	051445	022463	041501	FMT0: .ASCIZ /%S3%ACSR (SEL%D1%A) DOES NOT RESPOND%N/
3078	011002	051123	024040	042523	
3079	011010	022514	030504	040445	
3080	011016	020051	047504	051505	
3081	011024	047040	052117	051040	
3082	011032	051505	047520	042116	
3083	011040	047045	000		
3084	011043	111	052116	051105	MEF3A: .ASCIZ /INTERNAL DMP-11 DIAGNOSTIC FAILED/
3085	011050	040516	020114	046504	
3086	011056	026520	030461	042040	
3087	011064	040511	047107	051517	
3088	011072	044524	020103	040506	
3089	011100	046111	042105	000	
3090	011105	045	052101	051505	MEF3: .ASCII '%ATEST CODE- %03''
3091	011112	020124	047503	042504	
3092	011120	020055	047445	063	
3093	011125	045	022516	044501	MEF4: .ASCIZ /%N%AINTERNAL DMP-11-LINE UNIT TEST FAILURE/
3094	011132	052116	051105	040516	
3095	011140	020114	046504	026520	
3096	011146	030461	046055	047111	
3097	011154	020105	047125	052111	
3098	011162	052040	051505	020124	
3099	011170	040506	046111	051125	
3100	011176	000105			
3101	011200	047111	042524	043122	MEFC: .ASCIZ /INTERFACE MICRO-DIAGNOSTIC FAILURE/
3102	011206	041501	020105	044515	
3103	011214	051103	026517	044504	
3104	011222	043501	047516	052123	
3105	011230	041511	043040	044501	
3106	011236	052514	042522	000	
3107	011243	045	020101	052515	MEF5: .ASCII '%A MULTIPORT RAM WRITE/READ ERROR%N'
3108	011250	052114	050111	051117	
3109	011256	020124	040522	020115	
3110	011264	051127	052111	027505	
3111	011272	042522	042101	042440	
3112	011300	051122	051117	047045	
3113	011306	040445	043440	047517	.ASCIZ '%A GOOD= %03%A BAD= %03''
3114	011314	036504	022440	031517	
3115	011322	040445	041040	042101	

CZDMTC.P11

25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

3116	011330	020075	047445	000063		
3117						
3118	011336	040445	047040	051120	MEF6:	.ASCII '%A NPR TRANSFER TEST'
3119	011344	052040	040522	051516		
3120	011352	042506	020122	042524		
3121	011360	052123				
3122	011362	047045	040445	043440		.ASCIZ '%N%A GOOD= %06%A BAD= %06''
3123	011370	047517	036504	022440		
3124	011376	033117	040445	041040		
3125	011404	042101	020075	047445		
3126	011412	000066				
3127						
3128	011414	047125	054105	042520	MEF7:	.ASCIZ /UNEXPECTED TEST LOOP HANG/
3129	011422	052103	042105	052040		
3130	011430	051505	020124	047514		
3131	011436	050117	044040	047101		
3132	011444	000107				
3133	011446	046504	020120	047111	MEF8:	.ASCIZ /DMP INTERRUPTED TO WRONG VECTOR/
3134	011454	042524	051122	050125		
3135	011462	042524	020104	047524		
3136	011470	053440	047522	043516		
3137	011476	053040	041505	047524		
3138	011504	000122				
3139	011506	047111	047503	051122	EROIC:	.ASCIZ /INCORRECT CODE RETURNED/
3140	011514	041505	020124	047503		
3141	011522	042504	051040	052105		
3142	011530	051125	042516	000104		
3143	011536	047045	052045	040445	MEF11:	.ASCIZ '%N%T%A CODE INCORRECT%N%T%N%06%S3%06'
3144	011544	041440	042117	020105		
3145	011552	047111	047503	051122		
3146	011560	041505	022524	022516		
3147	011566	022524	022516	033117		
3148	011574	051445	022463	033117		
3149	011602	000				
3150	011603	045	022524	033117	MEF1A:	.ASCIZ '%T%06'
3151	011610	000				
3152	011611	106	044501	052514	MFPC:	.ASCIZ 'FAILURE AT PC '
3153	011616	042522	040440	020124		
3154	011624	041520	000040			
3155						
3156	011630	047522	020115	052506	MRFT:	.ASCIZ 'ROM FUNCTION TEST ERROR'
3157	011636	041516	044524	047117		
3158	011644	052040	051505	020124		
3159	011652	051105	047522	000122		
3160	011660	040445	042122	020111	MRDI:	.ASCIZ '%ARDI FAILED TO SET'
3161	011666	040506	046111	042105		
3162	011674	052040	020117	042523		
3163	011702	000124				
3164	011704	040445	042122	020117	MRDO:	.ASCIZ '%ARDO FAILED TO SET'
3165	011712	040506	046111	042105		
3166	011720	052040	020117	042523		
3167	011726	000124				
3168	011730	047507	042117	020040	MGB:	.ASCIZ 'GOOD BAD'
3169	011736	020040	020040	041040		
3170	011744	042101	000			
3171	011747	122	052105	051125	M12F:	.ASCIZ 'RETURN KEY'

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

3172	011754	020116	042513	000131	
3173	011762	051105	047522	000122	M13F: .ASCIZ 'ERROR'
3174	011770	042122	000117		MFRO: .ASCIZ 'RDO'
3175	011774	042122	000111		MFRI: .ASCIZ 'RDI'
3176	012000	052517	050124	052125	M18F: .ASCIZ 'OUTPUT'
3177	012006	000			
3178	012007	124	050131	000105	M28F: .ASCIZ 'TYPE'
3179	012014	040504	040524	000	M30F: .ASCIZ 'DATA'
3180	012021	122	044504	051440	MEF14: .ASCIZ 'RDI SET WHEN EXPECTING RDO TO BE SET''
3181	012026	052105	053440	042510	
3182	012034	020116	054105	042520	
3183	012042	052103	047111	020107	
3184	012050	042122	020117	047524	
3185	012056	041040	020105	042523	
3186	012064	000124			
3187	012066	042122	020117	042523	MEF15: .ASCIZ 'RDO SET WHEN EXPECTING RDI TO BE SET''
3188	012074	020124	044127	047105	
3189	012102	042440	050130	041505	
3190	012110	044524	043516	051040	
3191	012116	044504	052040	020117	
3192	012124	042502	051440	052105	
3193	012132	000			
3194	012133	111	041516	051117	MEF16A: .ASCIZ '/INCORRECT CHARACTER COUNT RETURNED/'
3195	012140	042522	052103	041440	
3196	012146	040510	040522	052103	
3197	012154	051105	041440	052517	
3198	012162	052116	051040	052105	
3199	012170	051125	042516	000104	
3200	012176	047111	047503	051122	MEF17A: .ASCIZ '/INCORRECT REC BUFFER ADDR. RETURNED/'
3201	012204	041505	020124	042522	
3202	012212	020103	052502	043106	
3203	012220	051105	040440	042104	
3204	012226	027122	051040	052105	
3205	012234	051125	042516	000104	
3206	012242	047111	047503	051122	MEF18A: .ASCIZ '/INCORRECT TRIB NUMBER RETURNED/'
3207	012250	041505	020124	051124	
3208	012256	041111	047040	046525	
3209	012264	042502	020122	042522	
3210	012272	052524	047122	042105	
3211	012300	000			
3212	012301	045	043501	047517	MEF18: .ASCII '%AGOOD= %03%A BAD- %03''
3213	012306	036504	022440	031517	
3214	012314	040445	041040	042101	
3215	012322	020075	047445	063	
3216	012327	045	022516	022524	.ASCIZ '%N%T%06''
3217	012334	033117	000		
3218	012337	115	051505	040523	MEF19A: .ASCIZ '/MESSAGE DA' , COMPARE ERROR/'
3219	012344	042507	042040	052101	
3220	012352	020101	047503	050115	
3221	012360	051101	020105	051105	
3222	012366	047522	000122		
3223	012372	040445	044103	051101	MEF19: .ASCII '%A CHARACTER# %03%A SENT CODE''
3224	012400	041501	042524	021522	
3225	012406	022440	031517	040445	
3226	012414	051440	047105	020124	
3227	012422	047503	042504		

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

3228	012426	022440	031517	040445		.ASCIZ '' %03%A RECEIVED CODES %03''
3229	012434	051040	041505	044505		
3230	012442	042526	020104	047503		
3231	012450	042504	020123	047445		
3232	012456	000063				
3233	012460	040445	047507	042117	MEF23:	.ASCII '%AGOOD= %06%A BAD= %06''
3234	012466	020075	047445	022466		
3235	012474	020101	040502	036504		
3236	012502	022440	033117			
3237	012506	047045	052045	047445		.ASCIZ '%N%T%06''
3238	012514	000066				
3239	012516	042122	020117	046111	MEF30:	.ASCIZ /RDO ILLEGALLY SET/
3240	012524	042514	040507	046114		
3241	012532	020131	042523	000124		
3242	012540	047522	020115	042526	MEF31:	.ASCIZ /ROM VERSION INCORRECT/
3243	012546	051522	047511	020116		
3244	012554	047111	047503	051122		
3245	012562	041505	000124			
3246	012566	053440	047522	043516	MEF32:	.ASCIZ / WRONG OPTION TYPE SELECTED IN TABLE/
3247	012574	047440	052120	047511		
3248	012602	020116	054524	042520		
3249	012610	051440	046105	041505		
3250	012616	042524	020104	047111		
3251	012624	052040	041101	042514		
3252	012632	000				
3253	012633	045	052501	044516	TFM20:	.ASCII '%AUNIT RETURNED NON-EXISTENT MEM ERR FOR ADD'
3254	012640	020124	042522	052524		
3255	012646	047122	042105	047040		
3256	012654	047117	042455	044530		
3257	012662	052123	047105	020124		
3258	012670	042515	020115	051105		
3259	012676	020122	047506	020122		
3260	012704	042101	104			
3261	012707	040	047445	022462		.ASCIZ ' %02%A00000%N%A-MEMORY DOES NOT EXIST!'
3262	012714	030101	030060	030060		
3263	012722	047045	040445	046455		
3264	012730	046505	051117	020131		
3265	012736	047504	051505	047040		
3266	012744	052117	042440	044530		
3267	012752	052123	000041			
3268	012756	040445	040504	040524	TFM21:	.ASCIZ '%ADATA ERROR IN TRANSFER TO RECEIVE BUFFER AT ADDRESS '
3269	012764	042440	051122	051117		
3270	012772	044440	020116	051124		
3271	013000	047101	043123	051105		
3272	013006	052040	020117	042522		
3273	013014	042503	053111	020105		
3274	013022	052502	043106	051105		
3275	013030	040440	020124	042101		
3276	013036	051104	051505	020123		
3277	013044	000				
3278	013045	045	031117	040445	TFM2A:	.ASCII '%02%A00000%N%ADATA SENT = %06'
3279	013052	030060	030060	022460		
3280	013060	022516	042101	052101		
3281	013066	020101	042523	052116		
3282	013074	036440	022440	033117		
3283	013102	040445	020054	040504		.ASCIZ '%A, DATA RECD = %06'

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

3284	013110	040524	051040	041505	
3285	013116	020104	020075	047445	
3286	013124	000066			
3287	013126	040445	047125	052111	TFM22: .ASCIZ '%AUNIT FAILED TO DETECT NON-EXISTENT MEMORY, ADDRESS '
3288	013134	043040	044501	042514	
3289	013142	020104	047524	042040	
3290	013150	052105	041505	020124	
3291	013156	047516	026516	054105	
3292	013164	051511	042524	052116	
3293	013172	046440	046505	051117	
3294	013200	026131	040440	042104	
3295	013206	042522	051523	000040	TFM22A: .ASCIZ '%02%A00000%N%A W A R N I N G PARTS OF THIS DIAGNOSTIC'
3296	013214	047445	022462	030101	
3297	013222	030060	030060	047045	
3298	013230	040445	053440	040440	
3299	013236	051040	047040	044440	
3300	013244	047040	043440	020040	
3301	013252	040520	052122	020123	
3302	013260	043117	052040	044510	
3303	013266	020123	044504	043501	
3304	013274	047516	052123	041511	
3305	013302	000			
3306	013303	045	020101	040515	TFM22B: .ASCIZ '%A MAY HAVE BEEN %N%ADESTROYED BY THE NPR TRANSFER!'
3307	013310	020131	040510	042526	
3308	013316	041040	042505	020116	
3309	013324	047045	040445	042504	
3310	013332	052123	047522	042531	
3311	013340	020104	054502	052040	
3312	013346	042510	047040	051120	
3313	013354	052040	040522	051516	
3314	013362	042506	020522	000	
3315	013367	045	041501	041522	TFM24: .ASCIZ '%ACRC ERROR IN ROM E%02%A READ = %06%A ; CALCULATED = %06''
3316	013374	042440	051122	051117	
3317	013402	044440	020116	047522	
3318	013410	020115	022505	031117	
3319	013416	040445	051040	040505	
3320	013424	020104	020075	047445	
3321	013432	022466	020101	020073	
3322	013440	040503	041514	046125	
3323	013446	052101	042105	036440	
3324	013454	022440	033117	000	
3325	013461	045	042501	051122	TFM25: .ASCII '%AERROR IN ROM E%02%A, ''
3326	013466	051117	044440	020116	
3327	013474	047522	020115	022505	
3328	013502	031117	040445	020054	
3329	013510	044123	052517	042114	.ASCIZ ''SHOULD BE ROM NO.: %T%A, NO. READ IS: %T''
3330	013516	041040	020105	047522	
3331	013524	020115	047516	035056	
3332	013532	020040	052045	040445	
3333	013540	020054	047516	020056	
3334	013546	042522	042101	044440	
3335	013554	035123	022440	000124	
3336	013562	047045	052045	047045	DFMT4: .ASCIZ '/%N%T%N%T%N/'
3337	013570	052045	047045	000	
3338	013575	045	031517	051445	DFMT5: .ASCIZ '/%03%S5%03%S5%03%S5%03%N/'
3339	013602	022465	031517	051445	

CZDMTC.P11 25-MAR-81 08:24

GLOBAL ERROR REPORT SECTION

3340	013610	022465	031517	051445
3341	013616	022465	031517	047045
3342	013624	000		
3343	013625	045	032123	047445
3344	013632	022463	032523	047445
3345	013640	022463	032523	047445
3346	013646	022463	032523	047445
3347	013654	022463	000116	
3348	013660	052045	047045	000
3349	013665	045	022516	020101
3350	013672	047125	052111	047040
3351	013700	046525	042502	035122
3352	013706	022440	032504	040445
3353	013714	051040	046517	047040
3354	013722	046525	042502	020122
3355	013730	051511	020072	052045
3356	013736	040445	020040	042522
3357	013744	027126	047040	027117
3358	013752	044440	035123	022440
3359	013760	000124		
3360				
3361	013762	000	000	
3362	013764	000	000	
3363				

DFMT6: .ASCIZ /%S4%03%S5%03%S5%03%S5%03%N/

DFMT9: .ASCIZ /%T%N/

ROMMSG: .ASCIZ '%N%A UNIT NUMBER: %D5%A ROM NUMBER IS: %T%A REV. NO. IS: %T''

.EVEN
ROMNO: .BYTE 0,0
REVNO: .BYTE 0,0
.EVEN

CZDMTC.P11 25-MAR-81 08:24

REPORT CODING SECTION

.SBTTL REPORT CODING SECTION

3364
 3365
 3366
 3367
 3368
 3369
 3370
 3371
 3372 013766
 3373
 3374
 3375 013766 000167
 3376 013770 000000
 3377
 3378
 3379 013772
 3380 013772 104425
 3381
 3382
 3383
 3384
 3385

 : THE REPORT CODING SECTION CONTAINS THE
 : 'PRINTS' CALLS THAT GENERATE STATISTICAL REPORTS.
 :--

L\$RPT::

.WORD JSJMP
 .WORD L10024-2-

L10024:

TRAP C\$RPT

CZDMTC.P11 25-MAR-81 08:24

INITIALIZE SECTION

3386
3387
3388
3389
3390
3391
3392
3393 013774
3394
3395 013774 010637 002322
3396 014000 005037 002324
3397 014004 005037 002326
3398 014010 005037 002376
3399 014014 005037 002400
3400 014020 005737 002330
3401 014024 001007
3402 014026 013737 000004 002316
3403 014034 013737 000006 002320
3404 014042 000406
3405 014044 013737 002316 000004 6\$:
3406 014052 013737 002320 000006
3407 014060 012737 000001 002330 9\$:
3408
3409 014066 012700 000040
3410 014072 104447
3411 014074 103415
3412
3413 014076 012700 000037
3414 014102 104447
3415 014104 103411
3416
3417 014106 012700 000035
3418 014112 104447
3419 014114 103411
3420
3421 014116 012700 000036
3422 014122 104447
3423 014124 103540
3424 014126 000416
3425 014130
3426 014130 005037 002346
3427
3428 014134 005037 002350
3429 014140
3430 014140 012737 177777 002332
3431 014146 005237 002354
3432 014152 005237 002346
3433 014156 012737 000001 002352
3434
3435
3436 014164
3437 014164 005237 002332
3438 014170 023737 002332 002012
3439 014176 002360
3440 014200 013700 002332
3441 014204 104442

```

.SBTTL INITIALIZE SECTION
://////
:// THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
:// AT THE BEGINNING OF THE TEST SEQUENCE ON THE NEXT UNIT.
://////
L$INIT::
      MOV      SP,PSTACK      ;SAVE BASE-LEVEL STACK POINTER
      CLR      SUBRPC         ;CLEAR SUBR CALL PC
      CLR      ERROR1         ;CLEAR ERROR FLAGS
      CLR      MODQ22         ;CLEAR DMV Q22 FORMAT FLAG.
      CLR      EXLOOP         ;CLEAR DMV EXTERNAL LOOP FLAG
      TST      FRSTIM         ;SEE IF FIRST TIME THROUGH AFTER LOAD
      BNE      6$             ;BR IF NOT
      MOV      @#4,SAVE4      ;SAVE ERROR TRAP VECTOR
      MOV      @#6,SAVE6
      BR       9$
6$:   MOV      SAVE4,@#4      ;RESTORE ERROR TRAP VECTOR
      MOV      SAVE6,@#6
9$:   MOV      #1,FRSTIM      ;MARK FLAG FOR NEXT TIME THROUGH
:SEE IF PROGRAM JUST STARTED, BR IF YES
      MOV      #EF.START,R0
      TRAP     C$REFG
      BCS      STARST
:SEE IF PROGRAM JUST RESTARTED, BR IF YES
      MOV      #EF.RESTART,R0
      TRAP     C$REFG
      BCS      STARST
:SEE IF THIS IS A NEW PASS, BR IF YES
      MOV      #EF.NEW,R0
      TRAP     C$REFG
      BCS      NEWST
:SEE IF PROGRAM WAS JUST CONTINUED
      MOV      #EF.CONTINUE,R0
      TRAP     C$REFG
      BCS      ENDIT
      BR       GETPRM
STARST:
      CLR      STARES         ;CLEAR FLAG TO SHOW JUST HAD STA OR RES
:CLEAR DEVICE MAP
      CLR      DEVMAP
NEWST:
      MOV      #-1,LOGDEV     ;RESET LOGICAL DEVICE TO -1
      INC      FRSPAS         ;INCREMENT NO. OF PASSES AFTER LOAD
      INC      STARES         ;INCREMENT NO. OF PASSES SINCE STA OR RES
      MOV      #BIT0,DEVPTR   ;INIT DEVICE MAP BIT POINTER
: GET UNIBUS ADDRESS, VECTOR, PRIORITY LEVEL, SWITCH PACKS, TEST
: CONNECTOR INFORMATION FOR THIS LOGICAL DEVICE
GETPRM:
      INC      LOGDEV         ;INCREMENT LOGICAL DEVICE NUMBER
      CMP      LOGDEV,L$UNIT  ;SEE IF MAXIMUM UNIT NO. EXCEEDED
      BGE      NEWST         ;BR IF YES....
      MOV      LOGDEV,R0
      TRAP     C$GPHRD

```

CZDMTC.P11 25-MAR-81 08:24

INITIALIZE SECTION

```

3442 014206 010001      MOV      R0,R1
3443 014210 103403      BCS      10$
3444 014212 006337 002352  ASL      DEVPTR      ;SHIFT DEVICE MAP BIT POINTER
3445 014216 000762      BR       GETPRM      ;SKIP THIS DEVICE
3446 014220
3447 014220 053737 002352 002350 10$:  BIS      DEVPTR,DEVMAP ;SHIFT DEVICE MAP BIT POINTER
3448 014226 006337 002352      ASL      DEVPTR      ;
3449 014232 012137 002472      MOV      (R1)+,OPTYP  ;SET THE OPTION TYPE
3450 014236 011137 002440      MOV      (R1),MPCSR  ;STORE POINTER TO M8200,4,7 CSR'S
3451 014242 011137 002442      MOV      (R1),BSEL1
3452 014246 005237 002442      INC      BSEL1      ;GET POINTER TO BSEL1 (MAINTENANCE REGISTER)
3453 014252 011137 002450      MOV      (R1),SEL4
3454 014256 062737 000004 002450  ADD      #4,SEL4      ;GET POINTER TO SEL4
3455 014264 011137 002444      MOV      (R1),SEL2
3456 014270 062737 000002 002444  ADD      #2,SEL2
3457 014276 011137 002446      MOV      (R1),BSEL3
3458 014302 062737 000003 002446  ADD      #3,BSEL3
3459 014310 011137 002452      MOV      (R1),BSEL5
3460 014314 062737 000005 002452  ADD      #5,BSEL5
3461 014322 011137 002456      MOV      (R1),BSEL7
3462 014326 062737 000007 002456  ADD      #7,BSEL7
3463 014334 011137 002460      MOV      (R1),BSEL10 ;FOR DMV
3464 014340 062737 000010 002460  ADD      #10,BSEL10
3465 014346 012137 002454      MOV      (R1)+,SEL6
3466 014352 062737 000006 002454  ADD      #6,SEL6      ;STORE POINTER TO SEL6
3467 014360 011137 002462      MOV      (R1),MPIVEC ;GET M8200,4,7 INPUT INTRPT VECTOR
3468 014364 012137 002464      MOV      (R1)+,MPOVEC
3469 014370 062737 000004 002464  ADD      #4,MPOVEC    ;GET M8200,4,7 OUTPUT INTRPT VECTOR
3470 014376 012137 002470      MOV      (R1)+,MPRIOR ;GET M8200,4,7 DEVICE PRIORITY
3471 014402 062701 000010      ADD      #10,R1      ;POINT TO TEST CON
3472 014406 012137 002476      MOV      (R1)+,TSTCON ;GET TEST CONNECTOR INDICATOR
3473 014412 012137 002466      MOV      (R1)+,SPEEDM ;GET SPEED
3474 014416 012137 002474      MOV      (R1)+,IFTYP
3475 014422 012137 002474      MOV      (R1)+,IFTYP ;FIRST TIME SKIP RUN WORD;THEN LOAD
3476                                     ;INTERFACE TYPE
3477 014426
3478 014426
3479 014426 104411      ENDIT:
L10025: TRAP      CS$INIT
3480
3481
3482 014430      L$AUTO::
3483 014430 004737 004456      JSR      PC,MINITR   ;INITIALIZE
3484 014434 005037 002342      CLR      CNT         ;+COUNTER
3485
3486 014440 004737 004244      1$:  JSR      PC,WAIT50   ;STALL
3487 014444 005777 165770      TST      @SELO      ;HAS IT STARTED?
3488 014450 100406      BMI      4$
3489 014452 105337 002342      DECB    COUNT      ;TIME UP?
3490 014456 001370      BNE     1$         ;NO LOOP
3491 014460 013700 002332      MOV      LOGDEV,R0
3492 014464 104451      TRAP    CS$DODU
3493
3494 014466
3495 014466 104433      4$:  TRAP    CS$RESET
3496 014470      L10026:
3497 014470 104461      TRAP    CS$AUTO

```


CZDMTC.P11

25-MAR-81 08:24

INITIALIZE SECTION

' 3498

CZDMTC.P11 25-MAR-81 08:24

CLEANUP CODING SECTION

.SBTTL CLEANUP CODING SECTION

3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515

:/

```
:/ THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
:/ AT THE END OF THE TEST SEQUENCE ON A PARTICULAR UNIT.
:/
```

014472
014472 104433
014474
014474 104412

L\$CLEAN::
TRAP C\$RESET
L10027:
TRAP C\$CLEAN

CZDMTC.P11 25-MAR-81 08:24

DROP UNIT SECTION

3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532

.SBTTL DROP UNIT SECTION

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

014476

L\$DU::
:ISSUE UNIBUS RESET TO CLEAN UP
TRAP C\$RESET

014476 104433

014500

L10030:
TRAP C\$DU

014500 104453

CZDMTC.P11 25-MAR-81 08:24

ADD UNIT SECTION

3533
 3534
 3535
 3536
 3537
 3538
 3539
 3540
 3541 014502
 3542 014502
 3543 014502 104452
 3544

.SBTTL ADD UNIT SECTION

```

:////////////////////
:/ THE ADD-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
:/ TO BE (A) TESTED FOR THE FIRST TIME, OR (B) RESUMED IN TESTING. IF
:/ 'EF.AUNIT' IS SET, THE UNIT WILL BE TESTED AS A NEW UNIT.
:////////////////////

```

```

L$AU::
L10031:
      TRAP   C$AU

```

CZDMTC.P11

25-MAR-81 08:24

ADD UNIT SECTION

3545
3546
3547
3548
3549

CZDMTC.P11 25-MAR-81 08:24

HARDWARE TESTS

.SBTTL HARDWARE TESTS

```

3550
3551
3552
3553
3554
3555
3556
3557
3558 014504
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570 014504
3571 014504 012746 000340
3572 014510 012746 014616
3573 014514 012746 000004
3574 014520 012746 000003
3575 014524 104437
3576 014526 062706 000010
3577 014532 005037 014614
3578 014536 005001
3579 014540 005777 165674
3580 014544 012701 000002
3581 014550 005777 165670
3582 014554 012701 000004
3583 014560 005777 165664
3584 014564 012701 000006
3585 014570 005777 165660
3586 014574 005737 014614
3587 014600 001401
3588 014602 104444
3589 014604
3590 014604 012700 000004
3591 014610 104436
3592
3593 014612
3594 014612 104401
3595
3596 014614 000000
3597
3598 014616
3599 014616 005737 014614
3600 014622 001006
3601 014624 104455
3602 014626 000023
3603 014630 010746
3604 014632 007350
3605 014634 005237 014614

```

```

.SBTTL :***** TEST 1 *****
.SBTTL * ADDRES TEST-VERIFY THAT ALL MCPU ADDRESSES RESPOND
ZZ
:*ECB
:*
:* THIS IS THE VERY FIRST TEST IN NORMAL SEQUENCE
:* IT IS USED TO VERIFY THAT DMP OR DMV-11 UNDER TEST, RESPONDS
:* TO THE ADDRESS THAT YOU THINK IT IS AT. ON DMP FAILURE CHECK
:* ADDRESS SWITCHES ON THE M8207 MICRO-CPU. WITH LITTLE
:* DOUBT, THIS FAILURE CAN ONLY BE ATTRIBUTED TO THE M8207 BOARD.
:* NOTE:8207 IS DMP ONLY.....
:*
:*-
.SBTTL :***** TEST 1 *****
T1::
MOV #PRI07,-(SP)
MOV #ECBINT,-(SP)
MOV #4,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP
CLR JMO ;CLEAR FLAG
CLR R1
TST @SELO ;TEST CSR 0
MOV #2,R1 ;SAVE OFFSET FOR NEXT CSR
TST @SEL2 ;TEST CSR 2
MOV #4,R1 ;SAVE OFFSET
TST @SEL4 ;TEST CSR 4
MOV #6,R1 ;SAVE OFFSET
TST @SEL6 ;TEST CSR 6
TST JMO ;WAS THERE A NXM TRAP
BEQ 10$ ;IF NOT EXIT CLEANLY
TRAP C$DCLN
10$:
MOV #4,R0
TRAP C$CVEC
L10032:
TRAP C$ETST
JMO: .WORD 0 ;FLAG FOR O'CONNOR CODE
ECBINT::
TST JMO ;HAVE WE HAD AT LEAST 1 TRAP
BNE 10$
TRAP C$ERDF
.WORD 19
.WORD EMT0
.WORD ERR1
INC JMO ;SET FLAG

```

CZDMTC.P11 25-MAR-81 08:24

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

3606	014640		
3607	014640	010146	
3608	014642	012746	010774
3609	014646	012746	000002
3610	014652	010600	
3611	014654	104415	
3612	014656	062706	000006
3613	014662		
3614	014662	000002	
3615			
3616			
3617			

```

10$:
      MOV     R1,-(SP)
      MOV     #FMT0,-(SP)
      MOV     #2,-(SP)
      MOV     SP,R0
      TRAP   C$PNTX
      ADD     #6,SP
L10033:
      RTI

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 2 *****

```

3618
3619
3620 014664
3621
3622
3623
3624
3625
3626
3627 014664
3628 014664 022737 000000 002472
3629 014672 001150
3630 014674 012737 000003 002366
3631 014702 012737 000000 002404
3632
3633 014710 012737 177777 002374
3634
3635 014716 004737 004274
3636 014722 117737 165526 002372
3637 014730 005237 002404
3638 014734 004737 004274
3639 014740 117737 165510 002373
3640 014746 005237 002404
3641 014752 023727 002404 004000
3642 014760 001403
3643
3644 014762 004737 004376
3645 014766 000753
3646
3647 014770 005137 002374
3648 014774 023737 002374 002372
3649 015002 001404
3650
3651
3652 015004 104455
3653 015006 000024
3654 015010 000000
3655 015012 010162
3656 015014 012737 003775 002404
3657 015022 012737 000060 002370
3658 015030 004737 004274
3659 015034 117737 165414 002372
3660 015042 123737 002370 002372
3661 015050 001404
3662 015052 104455
3663 015054 000025
3664 015056 000000
3665 015060 010224
3666 015062
3667 015062 022737 000001 002346
3668 015070 001031
3669 015072 113737 002372 013762
3670 015100 012737 003774 002404
3671 015106 004737 004274
3672 015112 117737 165336 013764
3673 015120 012746 013764

```

```

.SBTTL ***** TEST 2 *****
.SBTTL *DMP ONLY VERIFY CONTENTS OF ROM 3
ZZ
;* THIS TEST DONE FOR DMP ONLY
;*
;* IN THIS TEST WE'LL VERIFY THE CONTENTS OF ROM 3
;*
.SBTTL ***** TEST 2 *****
;:-CROMT-
T2::
CMP #0,OPTYP ;IS THIS AN 8207 DMP
BNE 60$ ;IF NOT END.....
MVC #3,ROMN ;ROM NUMBER
MOV #0,CADDR ;GET STARTING ADDR.
MOV #-1,CWORD ;INIT CRC WORD.
10$: JSR PC,GWORD ;GET FIRST BYTE.
MOVB @SEL6,WORDT ;STORE FIRST BYTE.
INC CADDR ;UPDATE ADDR.
JSR PC,GWORD ;GET NEXT BYTE.
MOVB @SEL6,WORDT+1 ;STORE IN HIGH BYTE OF WORDT
INC CADDR ;UPDATE ADDR.
CMP CADDR,#3777+1 ;AT END?
BEQ 20$ ;YES,EXIT LOOP.
JSR PC,CRCR ;NO-CALCULATE CRC ON THIS WORD.
BR 10$ ;LOOP.
20$: COM CWORD ;STORED CRC WORD IS COMPLEMENT.
CMP CWORD,WORDT ;EQUAL?
BEQ 30$ ;ROM CRC WORD BAD.
TRAP C$ERDF
.WORD 20
.WORD 0
.WORD ERR24
30$: MOV #3777-2,CADDR ;SET ROM NUMBER ADDRESS
MOV #60,ROMN1 ;ROM NUMBER
JSR PC,GWORD ;READ ROM NUMBER
MOVB @SEL6,WORDT ;STORE BYTE
CMPB ROMN1,WORDT ;GOOD?
BEQ 40$
TRAP C$ERDF
.WORD 21
.WORD 0
.WORD ERR25
40$: CMP #1,STARES ;IS THIS FIRST PASS
BNE 50$ ;IF NOT THEN GO TO 50
MOVB WORDT,ROMNO ;PUT ROM NO IN PRINT CONDITION
MOV #3777-3,CADDR
JSR PC,GWORD ;READ REV NO.
MOVB @SEL6,REVNO ;STORE BYTE
MOV #REVNO,-(SP)

```


CZDMTC.P11 25-MAR-81 08:24

***** TEST 2 *****

3674	015124	012746	013762			MOV	#ROMNO,-(SP)	
3675	015130	013746	002332			MOV	LOGDEV,-(SP)	
3676	015134	012746	013665			MOV	#ROMMSG,-(SP)	
3677	015140	012746	000004			MOV	#4,-(SP)	
3678	015144	010600				MOV	SP,R0	
3679	015146	104417				TRAP	C\$PNTF	
3680	015150	062706	000012			ADD	#12,SP	
3681	015154	012737	003773	002404	50\$:	MOV	#3777-4,CADDR	:GET VERSION
3682	015162	004737	004274			JSR	PC,GWORD	:READ IT
3683	015166	117737	165262	002372		MOVB	@SEL6,WORDT	
3684	015174	122737	000131	002372		CMPB	#131,WORDT	
3685	015202	001404				BEQ	60\$	
3686	015204	104455				TRAP	C\$ERDF	
3687	015206	000026				.WORD	22	
3688	015210	012540				.WORD	MEF31	
3689	015212	010422				.WORD	ERR32	
3690	015214				60\$:			
3691	015214				L10034:			
3692	015214	104401				TRAP	C\$ETST	

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 3 *****

3693
3694
3695 015216
3696
3697
3698
3699
3700
3701
3702 015216
3703 015216 022737 000000 002472
3704 015224 001150
3705 015226 012737 000002 002366
3706 015234 012737 000000 002404
3707
3708 015242 012737 177777 002374
3709
3710 015250 004737 004274
3711 015254 117737 165176 002372
3712 015262 005237 002404
3713 015266 004737 004274
3714 015272 117737 165160 002373
3715 015300 005237 002404
3716 015304 023727 002404 004000
3717 015312 001403
3718
3719 015314 004737 004376
3720 015320 000753
3721
3722 015322 005137 002374
3723 015326 023737 002374 002372
3724 015334 001404
3725
3726
3727 015336 104455
3728 015340 000024
3729 015342 000000
3730 015344 010162
3731 015346 012737 003775 002404 30\$:
3732 015354 012737 000061 002370
3733 015362 004737 004274
3734 015366 117737 165064 002372
3735 015374 123737 002370 002372
3736 015402 001404
3737 015404 104455
3738 015406 000025
3739 015410 000000
3740 015412 010224
3741 015414
3742 015414 022737 000001 002346 40\$:
3743 015422 001031
3744 015424 113737 002372 013762
3745 015432 012737 003774 002404
3746 015440 004737 004274
3747 015444 117737 165006 013764
3748 015452 012746 013764

```

.SBTTL :***** TEST 3 *****
.SBTTL *DMP ONLY VERIFY CONTENTS OF ROM 2
ZZ
;* THIS TEST DONE FOR DMP ONLY
;*
;* IN THIS TEST WE'LL VERIFY THE CONTENTS OF ROM 2
;*
.SBTTL :***** TEST 3 *****
;-CROMT-
T3::
CMP #0,OPTYP ;IS THIS AN 8207 DMP
BNE 60$ ;IF NOT END.....
MOV #2,ROMN ;ROM NUMBER
MOV #0,CADDR ;GET STARTING ADDR.
MOV #-1,CWORD ;INIT CRC WORD.
10$: JSR PC,GWORD ;GET FIRST BYTE.
MOVB @BSEL7,WORDT ;STORE FIRST BYTE.
INC CADDR ;UPDATE ADDR.
JSR PC,GWORD ;GET NEXT BYTE.
MOVB @BSEL7,WORDT+1 ;STORE IN HIGH BYTE OF WORDT.
INC CADDR ;UPDATE ADDR.
CMP CADDR,#3777+1 ;AT END?
BEQ 20$ ;YES,EXIT LOOP.
JSR PC,CRCR ;NO-CALCULATE CRC ON THIS WORD.
BR 10$ ;LOOP.
20$: COM CWORD ;STORED CRC WORD IS COMPLEMENT.
CMP CWORD,WORDT ;EQUAL?
BEQ 30$ ;ROM CRC WORD BAD.
TRAP C$ERDF
.WORD 20
.WORD 0
.WORD ERR24
30$: MOV #3777-2,CADDR ;SET ROM NUMBER ADDRESS
MOV #61,ROMN1 ;ROM NUMBER
JSR PC,GWORD ;READ ROM NUMBER
MOVB @BSEL7,WORDT ;STORE BYTE
CMPB ROMN1,WORDT ;GOOD?
BEQ 40$
TRAP C$ERDF
.WORD 21
.WORD 0
.WORD ERR25
40$: CMP #1,STARES ;IS THIS FIRST PASS
BNE 50$ ;IF NOT THEN GO TO 50
MOVB WORDT,ROMNO ;PUT ROM NO IN PRINT CONDITION
MOV #3777-3,CADDR
JSR PC,GWORD ;READ REV NO.
MOVB @BSEL7,REVNO ;STORE BYTE
MOV #REVNO,-(SP)

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 3 *****

```

3749 015456 012746 013762      MOV    #ROMNO,-(SP)
3750 015462 013746 002332      MOV    LOGDEV,-(SP)
3751 015466 012746 013665      MOV    #ROMMSG,-(SP)
3752 015472 012746 000004      MOV    #4,-(SP)
3753 015476 010600      MOV    SP,R0
3754 015500 104417      TRAP   C$PNTF
3755 015502 062706 000012      ADD    #12,SP
3756 015506 012737 003773 002404 50$:      MOV    #3777-4,CADDR      ;GET VERSION
3757 015514 004737 004274      JSR    PC,GWORD          ;READ IT
3758 015520 117737 164732 002372      MOVB  @BSEL7,WORDT
3759 015526 122737 000131 002372      CMPB  #131,WORDT
3760 015534 001404      BEQ    60$
3761 015536 104455      TRAP   C$ERDF
3762 015540 000026      .WORD 22
3763 015542 012540      .WORD MEF31
3764 015544 010422      .WORD ERR32
3765 015546      60$:
3766 015546      L10035:
3767 015546 104401      TRAP   C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 4 *****

3768
3769
3770 015550
3771
3772
3773
3774
3775
3776
3777 015550
3778 015550 022737 000000 002472
3779 015556 001150
3780 015560 012737 000004 002366
3781 015566 012737 004000 002404
3782
3783 015574 012737 177777 002374
3784
3785 015602 004737 004274
3786 015606 117737 164642 002372
3787 015614 005237 002404
3788 015620 004737 004274
3789 015624 117737 164624 002373
3790 015632 005237 002404
3791 015636 023727 002404 010000
3792 015644 001403
3793
3794 015646 004737 004376
3795 015652 000753
3796
3797 015654 005137 002374
3798 015660 023737 002374 002372
3799 015666 001404
3800
3801
3802 015670 104455
3803 015672 000024
3804 015674 000000
3805 015676 010162
3806 015700 012737 007775 002404
3807 015706 012737 000062 002370
3808 015714 004737 004274
3809 015720 117737 164530 002372
3810 015726 123737 002370 002372
3811 015734 001404
3812 015736 104455
3813 015740 000025
3814 015742 000000
3815 015744 010224
3816 015746
3817 015746 022737 000001 002346
3818 015754 001031
3819 015756 113737 002372 013762
3820 015764 012737 007774 002404
3821 015772 004737 004274
3822 015776 117737 164452 013764
3823 016004 012746 013764

```

.SBTTL :***** TEST 4 *****
.SBTTL *DMP ONLY VERIFY CONTENTS OF ROM 4
ZZ
:* THIS TEST DONE FOR DMP ONLY
:*
:* IN THIS TEST WE'LL VERIFY THE CONTENTS OF ROM 4
:*
.SBTTL :***** TEST 4 *****
      :-CROMT-
T4::
      CMP      #0,OPTYP      ;IS THIS AN 8207 DMP
      BNE      60$          ;IF NOT END.....
      MOV      #4,ROMN      ;ROM NUMBER
      MOV      #4000,CADDR   ;GET STARTING ADDR.
      MOV      #-1,CWORD    ;INIT CRC WORD.
10$:
      JSR      PC,GWORD     ;GET FIRST BYTE.
      MOVB    @SEL6,WORDT   ;STORE FIRST BYTE.
      INC     CADDR        ;UPDATE ADDR.
      JSR      F,GWORD     ;GET NEXT BYTE.
      MOVB    @SEL6,WORDT+1 ;STORE IN HIGH BYTE OF WORDT
      INC     CADDR        ;UPDATE ADDR.
      CMP     CADDR,#7777+1 ;AT END?
      BEQ     20$          ;YES,EXIT LOOP.
      JSR      PC,CRCR     ;NO-CALCULATE CRC ON THIS WORD.
      BR      10$         ;LOOP.
20$:
      COM     CWORD        ;STORED CRC WORD IS COMPLEMENT.
      CMP     CWORD,WORDT  ;EQUAL?
      BEQ     30$
      ;ROM CRC WORD BAD.
30$:
      TRAP    C$ERDF
      .WORD  20
      .WORD  0
      .WORD  ERR24
      MOV     #7777-2,CADDR ;SET ROM NUMBER ADDRESS
      MOV     #62,ROMN1    ;ROM NUMBER
      JSR     PC,GWORD     ;READ ROM NUMBER
      MOVB   @SEL6,WORDT   ;STORE BYTE
      CMPB   ROMN1,WORDT  ;GOOD?
      BEQ    40$
      TRAP    C$ERDF
      .WORD  21
      .WORD  0
      .WORD  ERR25
40$:
      CMP     #1,STARES    ;IS THIS FIRST PASS
      BNE     50$          ;IF NOT THEN GO TO 50
      MOVB   WORDT,ROMNO  ;PUT ROM NO IN PRINT CONDITION
      MOV     #7777-3,CADDR
      JSR     PC,GWORD     ;READ REV NO.
      MOVB   @SEL6,REVNO  ;STORE BYTE
      MOV     #REVNO,-(SP)

```

CZDMTC.P11 25-MAR-81 08:24

;***** TEST 4 *****

3824	016010	012746	013762			MOV	#ROMNO,-(SP)		
3825	016014	013746	002332			MOV	LOGDEV,-(SP)		
3826	016020	012746	013665			MOV	#ROMMSG,-(SP)		
3827	016024	012746	000004			MOV	#4,-(SP)		
3828	016030	010600				MOV	SP,R0		
3829	016032	104417				TRAP	C\$PNTF		
3830	016034	062706	000012			ADD	#12,SP		
3831	016040	012737	007773	002404	50\$:	MOV	#7777-4,CADDR		;GET VERSION
3832	016046	004737	004274			JSR	PC,GWORD		;READ IT
3833	016052	117737	164376	002372		MOVB	@SEL6,WORDT		
3834	016060	122737	000131	002372		CMPB	#131,WORDT		
3835	016066	001404				BEQ	60\$		
3836	016070	104455				TRAP	C\$ERDF		
3837	016072	000026				.WORD	22		
3838	016074	012540				.WORD	MEF31		
3839	016076	010422				.WORD	ERR32		
3840	016100				60\$:				
3841	016100				L10036:				
3842	016100	104401				TRAP	C\$ETST		

CZDMTC.P11 25-MAR-81 08:24

***** TEST 5 *****

3843
3844
3845 016102
3846
3847
3848
3849
3850
3851
3852 016102
3853 016102 022737 000000 002472
3854 016110 001150
3855 016112 012737 000001 002366
3856 016120 012737 004000 002404
3857
3858 016126 012737 177777 002374
3859
3860 016134 004737 004274
3861 016140 117737 164312 002372
3862 016146 005237 002404
3863 016152 004737 004274
3864 016156 117737 164274 002373
3865 016164 005237 002404
3866 016170 023727 002404 010000
3867 016176 001403
3868
3869 016200 004737 004376
3870 016204 000753
3871
3872 016206 005137 002374
3873 016212 023737 002374 002372
3874 016220 001404
3875
3876
3877 016222 104455
3878 016224 000024
3879 016226 000000
3880 016230 010162
3881 016232 012737 007775 002404
3882 016240 012737 000063 002370
3883 016246 004737 004274
3884 016252 117737 164200 002372
3885 016260 123737 002370 002372
3886 016266 001404
3887 016270 104455
3888 016272 000025
3889 016274 000000
3890 016276 010224
3891 016300
3892 016300 022737 000001 002346
3893 016306 001031
3894 016310 113737 002372 013762
3895 016316 012737 007774 002404
3896 016324 004737 004274
3897 016330 117737 164122 013764
3898 016336 012746 013764

```

.SBTTL :***** TEST 5 *****
.SBTTL *DMP ONLY VERIFY CONTENTS OF ROM 1
ZZ
:* THIS TEST DONE FOR DMP ONLY
:*
:* IN THIS TEST WE'LL VERIFY THE CONTENTS OF ROM 1
:*
.SBTTL :***** TEST 5 *****
      ; -CROMT-
T5::
      CMP      #0,OPTYP      ;IS THIS AN 8207 DMP
      BNE      60$          ;IF NOT END.....
      MOV      #1,ROMN      ;ROM NUMBER
      MOV      #4000,CADDR   ;GET STARTING ADDR.
      MOV      #71,CWORD    ;INIT CRC WORD.
10$:
      JSR      PC,GWORD     ;GET FIRST BYTE.
      MOV      @BSEL7,WORDT ;STORE FIRST BYTE.
      INC      CADDR        ;UPDATE ADDR.
      JSR      PC,GWORD     ;GET NEXT BYTE.
      MOV      @BSEL7,WORDT+1 ;STORE IN HIGH BYTE OF WORDT.
      INC      CADDR        ;UPDATE ADDR.
      CMP      CADDR,#7777+1 ;AT END?
      BEQ      20$         ;YES,EXIT LOOP.
      JSR      PC,CRCR      ;NO-CALCULATE CRC ON THIS WORD.
      BR      10$          ;LOOP.
20$:
      COM      CWORD        ;STORED CRC WORD IS COMPLEMENT.
      CMP      CWORD,WORDT  ;EQUAL?
      BEQ      30$
      ;ROM CRC WORD BAD.
30$:
      TRAP     C$ERDF
      .WORD   20
      .WORD   0
      .WORD   ERR24
      MOV      #7777-2,CADDR ;SET ROM NUMBER ADDRESS
      MOV      #63,ROMN1    ;ROM NUMBER
      JSR      PC,GWORD     ;READ ROM NUMBER
      MOV      @BSEL7,WORDT ;STORE BYTE
      CMP      ROMN1,WORDT  ;GOOD?
      BEQ      40$
      TRAP     C$ERDF
      .WORD   21
      .WORD   0
      .WORD   ERR25
40$:
      CMP      #1,STARES    ;IS THIS FIRST PASS
      BNE      50$          ;IF NOT THEN GO TO 50
      MOV      WORDT,ROMNO  ;PUT ROM NO IN PRINT CONDITION
      MOV      #7777-3,CADDR
      JSR      PC,GWORD     ;READ REV NO.
      MOV      @BSEL7,REVNO ;STORE BYTE
      MOV      #REVNO,-(SP)

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 5 *****

3899 016342 012746 013762
 3900 016346 013746 002332
 3901 016352 012746 013665
 3902 016356 012746 000004
 3903 016362 010600
 3904 016364 104417
 3905 016366 062706 000012
 3906 016372 012737 007773 002404 50\$:
 3907 016400 004737 004274
 3908 016404 117737 164046 002372
 3909 016412 122737 000131 002372
 3910 016420 001404
 3911 016422 104455
 3912 016424 000026
 3913 016426 012540
 3914 016430 010422
 3915 016432
 3916 016432
 3917 016432 104401

MOV #ROMNO,-(SP)
 MOV LOGDEV,-(SP)
 MOV #ROMMSG,-(SP)
 MOV #4,-(SP)
 MOV SP,R0
 TRAP C\$PNTF
 ADD #12,SP
 MOV #7777-4,CADDR ;GET VERSION
 JSR PC,GWORD ;READ IT
 MOVB @BSEL7,WORDT
 CMPB #131,WORDT
 BEQ 60\$
 TRAP C\$ERDF
 .WORD 22
 .WORD MEF31
 .WORD ERR32
 60\$:
 L10037:
 TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 6 *****

```

3918
3919
3920 016434
3921
3922
3923
3924
3925
3926
3927 016434
3928 016434 022737 000000 002472
3929 016442 001150
3930 016444 012737 000005 002366
3931 016452 012737 010000 002404
3932
3933 016460 012737 177777 002374
3934
3935 016466 004737 004274
3936 016472 117737 163756 002372
3937 016500 005237 002404
3938 016504 004737 004274
3939 016510 117737 163740 002373
3940 016516 005237 002404
3941 016522 023727 002404 014000
3942 016530 001403
3943
3944 016532 004737 004376
3945 016536 000753
3946
3947 016540 005137 002374
3948 016544 023737 002374 002372
3949 016552 001404
3950
3951
3952 016554 104455
3953 016556 000024
3954 016560 000000
3955 016562 010162
3956 016564 012737 013775 002404
3957 016572 012737 000064 002370
3958 016600 004737 004274
3959 016604 117737 163644 002372
3960 016612 123737 002370 002372
3961 016620 001404
3962 016622 104455
3963 016624 000025
3964 016626 000000
3965 016630 010224
3966 016632
3967 016632 022737 000001 002346
3968 016640 001031
3969 016642 113737 002372 013762
3970 016650 012737 013774 002404
3971 016656 004737 004274
3972 016662 117737 163566 013764
3973 016670 012746 013764

```

```

.SBTTL ***** TEST 6 *****
.SBTTL *DMP ONLY VERIFY CONTENTS OF ROM 5
ZZ
;* THIS TEST DONE FOR DMP ONLY
;*
;* IN THIS TEST WE'LL VERIFY THE CONTENTS OF ROM 5
.SBTTL ***** TEST 6 *****
;--CROMT-
T6::
CMP #0,OPTYP ;IS THIS AN 8207 DMP
BNE 60$ ;IF NOT END.....
MOV #5,ROMN ;ROM NUMBER
MOV #10000,CADDR ;GET STARTING ADDR.
MOV #-1,CWORD ;INIT CRC WORD.
10$: JSR PC,GWORD ;GET FIRST BYTE.
MOVB @SEL6,WORDT ;STORE FIRST BYTE.
INC CADDR ;UPDATE ADDR.
JSR PC,GWORD ;GET NEXT BYTE.
MOVB @SEL6,WORDT+1 ;STORE IN HIGH BYTE OF WORDT
INC CADDR ;UPDATE ADDR.
CMP CADDR,#13777+1 ;AT END?
BEQ 20$ ;YES,EXIT LOOP.
JSR PC,CRCR ;NO-CALCULATE CRC ON THIS WORD.
BR 10$ ;LOOP.
20$: COM CWORD ;STORED CRC WORD IS COMPLEMENT.
CMP CWORD,WORDT ;EQUAL?
BEQ 30$
;ROM CRC WORD BAD.
TRAP C$ERDF
.WORD 20
.WORD 0
.WORD ERR24
30$: MOV #13777-2,CADDR ;SET ROM NUMBER ADDRESS
MOV #64,ROMN1 ;ROM NUMBER
JSR PC,GWORD ;READ ROM NUMBER
MOVB @SEL6,WORDT ;STORE BYTE
CMPB ROMN1,WORDT ;GOOD?
BEQ 40$
TRAP C$ERDF
.WORD 21
.WORD 0
.WORD ERR25
40$: CMP #1,STARES ;IS THIS FIRST PASS
BNE 50$ ;IF NOT THEN GO TO 50
MOVB WORDT,ROMNO ;PUT ROM NO IN PRINT CONDITION
MOV #13777-3,CADDR
JSR PC,GWORD ;READ REV NO.
MOVB @SEL6,REVNO ;STORE BYTE
MOV #REVNO,-(SP)

```


CZDMTC.P11 25-MAR-81 08:24

***** TEST 6 *****

3974 016674 012746 013762
 3975 016700 013746 002332
 3976 016704 012746 013665
 3977 016710 012746 000004
 3978 016714 010600
 3979 016716 104417
 3980 016720 062706 000012
 3981 016724 012737 013773 002404 50\$:
 3982 016732 004737 004274
 3983 016736 117737 163512 002372
 3984 016744 122737 000131 002372
 3985 016752 001404
 3986 016754 104455
 3987 016756 000026
 3988 016760 012540
 3989 016762 010422
 3990 016764
 3991 016764
 3992 016764 104401

MOV #ROMNO,-(SP)
 MOV LOGDEV,-(SP)
 MOV #ROMMSG,-(SP)
 MOV #4,-(SP)
 MOV SP,R0
 TRAP C\$PNTF
 ADD #12,SP
 MOV #13777-4,CADDR ;GET VERSION
 JSR PC,GWORD ;READ IT
 MOVB @SEL6,WORDT
 CMPB #131,WORDT
 BEQ 60\$
 TRAP C\$ERDF
 .WORD 22
 .WORD MEF31
 .WORD ERR32
 60\$:
 L10040:
 TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 7 *****

```

3993
3994
3995 016766
3996
3997
3998
3999
4000
4001
4002 016766
4003 016766 022737 000000 002472
4004 016774 001150
4005 016776 012737 000014 002366
4006 017004 012737 010000 002404
4007
4008 017012 012737 177777 002374
4009
4010 017020 004737 004274
4011 017024 117737 163426 002372
4012 017032 005237 002404
4013 017036 004737 004274
4014 017042 117737 163410 002373
4015 017050 005237 002404
4016 017054 023727 002404 014000
4017 017062 001403
4018
4019 017064 004737 004376
4020 017070 000753
4021
4022 017072 005137 002374
4023 017076 023737 002374 002372
4024 017104 001404
4025
4026
4027 017106 104455
4028 017110 000024
4029 017112 000000
4030 017114 010162
4031 017116 012737 013775 002404
4032 017124 012737 000065 002370
4033 017132 004737 004274
4034 017136 117737 163314 002372
4035 017144 123737 002370 002372
4036 017152 001404
4037 017154 104455
4038 017156 000025
4039 017160 000000
4040 017162 010224
4041 017164
4042 017164 022737 000001 002346
4043 017172 001031
4044 017174 113737 002372 013762
4045 017202 012737 013774 002404
4046 017210 004737 004274
4047 017214 117737 163236 013764
4048 017222 012746 013764

```

```

.SBTTL ***** TEST 7 *****
.SBTTL *DMP ONLY VERIFY CONTENTS OF ROM 14
ZZ
;* THIS TEST DONE FOR DMP ONLY
;*
;* IN THIS TEST WE'LL VERIFY THE CONTENTS OF ROM 14
.SBTTL ***** TEST 7 *****
;:-CROMT-
T7::
CMP #0,OPTYP ;IS THIS AN 8207 DMP
BNE 60$ ;IF NOT END.....
MOV #14,ROMN ;ROM NUMBER
MOV #10000,CADDR ;GET STARTING ADDR.
MOV #-1,CWORD ;INIT CRC WORD.
10$: JSR PC,GWORD ;GET FIRST BYTE.
MOVB @BSEL7,WORDT ;STORE FIRST BYTE.
INC CADDR ;UPDATE ADDR.
JSR PC,GWORD ;GET NEXT BYTE.
MOVB @BSEL7,WORDT+1 ;STORE IN HIGH BYTE OF WORDT.
INC CADDR ;UPDATE ADDR.
CMP CADDR,#13777+1 ;AT END?
BEQ 20$ ;YES,EXIT LOOP.
JSR PC,CRCR ;NO-CALCULATE CRC ON THIS WORD.
BR 10$ ;LOOP.
20$: COM CWORD ;STORED CRC WORD IS COMPLEMENT.
CMP CWORD,WORDT ;EQUAL?
BEQ 30$ ;ROM CRC WORD BAD.
TRAP C$ERDF
.WORD 20
.WORD 0
.WORD ERR24
30$: MOV #13777-2,CADDR ;SET ROM NUMBER ADDRESS
MOV #65,ROMN1 ;ROM NUMBER
JSR PC,GWORD ;READ ROM NUMBER
MOVB @BSEL7,WORDT ;STORE BYTE
CMPB ROMN1,WORDT ;GOOD?
BEQ 40$
TRAP C$ERDF
.WORD 21
.WORD 0
.WORD ERR25
40$: CMP #1,STARES ;IS THIS FIRST PASS
BNE 50$ ;IF NOT THEN GO TO 50
MOVB WORDT,ROMNO ;PUT ROM NO IN PRINT CONDITION
MOV #13777-3,CADDR
JSR PC,GWORD ;READ REV NO.
MOVB @BSEL7,REVNO ;STORE BYTE
MOV #REVNO,-(SP)

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 7 *****

4049 017226 012746 013762
 4050 017232 013746 002332
 4051 017236 012746 013665
 4052 017242 012746 000004
 4053 017246 010600
 4054 017250 104417
 4055 017252 062706 000012
 4056 017256 012737 013773 002404 50\$:
 4057 017264 004737 004274
 4058 017270 117737 163162 002372
 4059 017276 122737 000131 002372
 4060 017304 001404
 4061 017306 104455
 4062 017310 000026
 4063 017312 012540
 4064 017314 010422
 4065 017316
 4066 017316
 4067 017316 104401

```

MOV #ROMNO,-(SP)
MOV LOGDEV,-(SP)
MOV #ROMMSG,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #12,SP
MOV #13777-4,CADDR ;GET VERSION
JSR PC,GWORD ;READ IT
MOVB @BSEL7,WORDT
CMPB #131,WORDT
BEQ 60$
TRAP C$ERDF
.WORD 22
.WORD MEF31
.WORD ERR32
60$:
L10041: TRAP C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 8 *****

4068
4069
4070 017320
4071
4072
4073
4074
4075
4076
4077
4078 017320
4079 017320 032737 000003 002472
4080 017326 001453
4081 017330 012737 000001 002366
4082 017336 012737 000060 002370
4083 017344 012737 140000 002436
4084 017352 004537 005472
4085 017356 005037 002402
4086 017362 104410
4087 017364 000072
4088 017366 022737 000001 002346
4089 017374 001030
4090
4091 017376 023737 013762 002370
4092 017404 001406
4093 017406 104455
4094 017410 000027
4095 017412 000000
4096 017414 010224
4097
4098 017416 104410
4099 017420 000036
4100 017422
4101 017422 012746 013764
4102 017426 012746 013762
4103 017432 013746 002332
4104 017436 012746 013665
4105 017442 012746 000004
4106 017446 010600
4107 017450 104417
4108 017452 062706 000012
4109
4110
4111
4112 017456
4113 017456
4114 017456 104401

```

.SBTTL ***** TEST 8 *****
.SBTTL * ROM VERIFY ROM 1 DMV
ZZ
:*
:*
:*
:* THIS TEST IS USED TO VERIFY THE CONTENTS OF ROM 1
:* THIS TEST IS NOT DONE FOR DMP
:*
.SBTTL ***** TEST 8 *****
T8::
BIT #3,OPTYP ;IS THIS DMV
BEQ RDVEX ;IF NOT EXIT
MOV #1,ROMN
MOV #60,ROMN1 ;SET UP ROM NUMBER(ASCII 0)
MOV #140000,ROMADD ;SET UP 1ST ROM ADDRESS
JSR R5,RMVRT ;GO CHECK ROM CRC
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10042-
CMP #1,STARES ;IS IT FIRST PASS
BNE RDVEX ;IF NOT EXIT

CMP ROMNO,ROMN1 ;COMPARE ROM NUMBER
BEQ 10$
TRAP C$ERDF
.WORD 23
.WORD 0
.WORD ERR25

TRAP C$ESCAPE
.WORD L10042-

10$:
MOV #REVNO,-(SP)
MOV #ROMNO,-(SP)
MOV LOGDEV,-(SP)
MOV #ROMMSG,-(SP)
MOV #4,-(SP)
MOV SP,RC
TRAP C$PNTF
ADD #12,SP

RDVEX:
L10042:
TRAP C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 9 *****

4115
4116
4117 017460
4118
4119
4120
4121
4122
4123
4124 017460
4125 017460 032737 000003 002472
4126 017466 001453
4127 017470 012737 000002 002366
4128 017476 012737 000061 002370
4129 017504 012737 160000 002436
4130 017512 004537 005472
4131 017516 005037 002402
4132 017522 104410
4133 017524 000072
4134 017526 022737 000001 002346
4135 017534 001030
4136
4137 017536 123737 013762 002370
4138 017544 001406
4139 017546 104455
4140 017550 000030
4141 017552 000000
4142 017554 010224
4143 017556 104410
4144 017560 000036
4145 017562
4146 017562 012746 013764
4147 017566 012746 013762
4148 017572 013746 002332
4149 017576 012746 013665
4150 017602 012746 000004
4151 017606 010600
4152 017610 104417
4153 017612 062706 000012
4154
4155
4156 017616
4157 017616
4158 017616 104401
4159

```

.SBTTL ***** TEST 9 *****
.SBTTL * ROM VERIFY ROM 2 DMV ONLY
ZZ
:
:
: * THIS IS THE TEST THAT VERIFIES THE CONTENTS OF ROM 2
: * OF THE DMV OPTION. THIS TEST IS NOT RUN FOR DMP
:
.SBTTL ***** TEST 9 *****
T9::
BIT #3,OPTYP ;IS THIS DMV
BEQ RDVEX2 ;IF NOT EXIT
MOV #2,ROMN
MOV #61,ROMN1 ;SETUP ROM NUMBER(ASCII 1)
MOV #160000,ROMADD ;SETUP 1ST ROM ADDRESS
JSR R5,RMVRT ;GO CHECK ROM CRC
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10043-
CMP #1,STARES
BNE RDVEX2 ;IF NOT FIRST PASS EXIT

CMPB ROMNO,ROMN1 ;CHECK ROM #
BEQ 10$
TRAP C$ERDF
.WORD 24
.WORD 0
.WORD ERR25
TRAP C$ESCAPE
.WORD L10043-

10$:
MOV #REVNO,-(SP)
MOV #ROMNO,-(SP)
MOV LOGDEV,-(SP)
MOV #ROMMSG,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #12,SP

RDVEX2:
L10043:
TRAP C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 10 *****

4160
 4161
 4162 017620
 4163
 4164
 4165
 4166
 4167
 4168
 4169
 4170
 4171
 4172
 4173
 4174
 4175
 4176
 4177
 4178
 4179
 4180
 4181
 4182
 4183
 4184
 4185
 4186
 4187
 4188
 4189
 4190
 4191
 4192
 4193
 4194
 4195
 4196
 4197
 4198
 4199
 4200
 4201
 4202
 4203
 4204
 4205
 4206
 4207
 4208
 4209
 4210
 4211
 4212 017620
 4213
 4214 017620 012737 000000 002342
 4215 017626 004737 004456

.SBTTL ***** TEST 10 *****
 .SBTTL *INITIALIZATION TEST (INTERNAL DIAGNOSTICS)

ZZ
 *
 * IN THIS TEST WE'LL START OUT BY SETTING THE MASTER CLEAR BIT (BIT 14 OF SELO)
 * THE LOGIC CLEARS AND STARTS THE MICRO DIAGNOSTICS. IF THE MICRO-DIAGNOSTICS
 * PASS, THE RUN BIT (BIT15 OF SELO) WILL SET.
 * IF THE RUN BIT FAILS TO SET WITHIN 300 MILLI-SEC, IT
 * PROBABLY MEANS THAT MICRO DIAGNOSTICS HAVE DETECTED AN
 * ERROR AND THE TEST CODE IS IN BSEL6
 *
 * DMP

TEST CODE	TEST ENTERED
143	BRANCH TEST
135	BRANCH EXTENDED TESTS
125,252,0	IBUS/OBUS TESTS
123	SCRATCH PAD TEST
151	ALU TESTS
222	MAIN MEMORY DATA TEST
132	MAIN MEMORY DUAL ADDRESS TEST
264	LINE UNIT TESTS
305	TESTS COMPLETE
	DMV TEST
101	BRANCH TEST
102	INTERNAL REG TEST
103	LOAD AND STORE INSTR.
104	COMPARE INSTR. TEST
105	INC/DEC INSTR.
106	SHIFT AND ROTATE INSTR.
107	LOGIC INSTR.
110	ADC,SBC,SED,CLD INSTRU.
111	STACK PUSH,PULL INSTR.
112	SUBROUTINE INSTR.
113	SCRATCH PAD,CSR,AND NPR
114	
115	FALSE INT TEST
116	RAM DATA AND ADDRESS
117	RAM ALTERNATING TEST
120	INDEX INDIRECT TEST
121	LINE UNIT TEST

* NOTE THE RUN BIT WILL BE SET EVEN IF THE LINE UNIT
 * TEST FAILS. TEST CODE MUST BE CHECKED TO FIND ERROR.
 * THESE CODES ARE SET UPON ENTRY OF EACH TEST
 * ONE SHOULD NOT BE DEPENDENT ON A BAD DMP-DMV MODULE
 * TO PASS A CORRECT TEST CODE. IF THIS TEST FAILS, YOU
 * SHOUL ^ RUN THE REPAIR LEVEL DIAGNOSTIC

NOTE
 * IF THIS TEST FAILS, CHECK SW7 OF SP#1 TO SEE IF RUN IS ENABLED.

.SBTTL ***** TEST 10 *****

T10::

MOV #0,COUNT ;CLEAR COUNTER
 JSR PC,MINITR

CZDMTC.P11 25-MAR-81 08:24

;***** TEST 10 *****

4216 017632
 4217 017632 004737 004244
 4218 017636 005777 162576
 4219 017642 100411
 4220 017644 005337 002342
 4221 017650 001370
 4222
 4223
 4224
 4225 017652 104455
 4226 017654 000031
 4227 017656 011043
 4228 017660 007376
 4229 017662 104410
 4230 017664 000116
 4231 017666 122777 000305 162560 20\$:
 4232 017674 001420
 4233 017676 122777 000264 162550
 4234 017704 001406
 4235
 4236 017706 104455
 4237 017710 000032
 4238 017712 011043
 4239 017714 007376
 4240
 4241
 4242 017716 104410
 4243 017720 000062
 4244
 4245 017722
 4246 017722 104455
 4247 017724 000033
 4248 017726 011125
 4249 017730 010422
 4250 017732 104410
 4251 017734 000046
 4252
 4253 017736
 4254 017736 112777 000200 162474
 4255 017744 012737 017766 002406
 4256 017752
 4257 017752 004537 005070
 4258 017756 005037 002402
 4259 017762 104410
 4260 017764 000016
 4261 017766
 4262
 4263
 4264
 4265 017766 032777 000020 162450 47\$:
 4266 017774 001766
 4267 017776
 4268 017776 104410
 4269 020000 000002
 4270 020002
 4271 020002 104401

10\$:

```

JSR    PC, WAIT50
TST    @SELO           ;TEST DONE?
BMI    20$             ;YES TEST FOR ERROR
DEC    COUNT           ;UPDATE COUNT IF NOT TOO LONG
BNE    10$             ;IN THIS WAIT LOOP, GO BACK

```

;INTERNAL DIAG FAILED

```

TRAP   C$ERDF
.WORD  25
.WORD  MEF3A
.WORD  ERR3
TRAP   C$ESCAPE
.WORD  L10044-
CMPB   #305, @BSEL6
BEQ    40$
CMPB   #264, @BSEL6
BEQ    30$

```

;LEGAL TEST COMPLETE CODE?

;LINE UNIT TEST FAILURE?

```

TRAP   C$ERDF
.WORD  26
.WORD  MEF3A
.WORD  ERR3

```

;UNKNOWN ERROR WHILE INITIALLING
;DMP-11

```

TRAP   C$ESCAPE
.WORD  L10044-

```

30\$:

```

TRAP   C$ERDF
.WORD  27
.WORD  MEF4
.WORD  ERR32
TRAP   C$ESCAPE
.WORD  L10044-

```

;MODULE FAULT

40\$:

```

MOVB   #RQI, @BSELO
MOV    #ERLB1, ERRADD

```

;SET RQI AND THEN WAIT FOR RDI TO SET.
;SET UP ERROR ADD.

TLB1:

```

JSR    R5, TOUT
CLR    ERRWRD
TRAP   C$ESCAPE
.WORD  L10044-

```

ERLB1:

.....
; TIME OUT ERROR REPORTS THIS ADDRESS
.....

47\$:

```

BIT    #RDI, @BSEL2
BEQ    TLB1

```

;DID RDI SET?

50\$:

```

TRAP   C$ESCAPE
.WORD  L10044-

```

L10044:

```
TRAP   C$SETST
```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 10 *****

4272
4273
4274
4275
4276 020004
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290
4291 020004
4292
4293 020004 032737 000003 002472
4294 020012 001402
4295 020014 000137 021134
4296 020020
4297 020020
4298 020020 004737 004522
4299
4300
4301
4302 020024 005037 002402
4303 020030 104410
4304 020032 001102
4305 020034 105077 162416
4306 020040 112777 000200 162372
4307 020046
4308 020046 004537 002732
4309
4310 020052 005037 002402
4311 020056 104410
4312 020060 001054
4313
4314
4315
4316
4317 020062 105077 162352
4318 020066 112777 000022 162360
4319 020074 105077 162356
4320 020100 112777 000001 162336
4321
4322 020106
4323 020106 012737 020130 002406
4324 020114 004537 005070
4325 020120 005037 002402
4326 020124 104410
4327 020126 001006

```

.SBTTL ***** TEST 11 *****
.SBTTL * MICRO-DIAGNOSTIC-INTERFACE TESTING DMP
ZZ
;* DMP ONLY THIS TEST...
;* THIS TEST WILL EXERCISE THE MICRO-CPU' INTERFACE TO THE PDP-11
;* WE FIRST START THE MCPU. NEXT WE GIVE THE COMMAND THAT
;* TAKES US TO THE INTERFACE DIAGNOSTIC CODE. ONCE THIS CODE IS
;* STARTED, WE MUST GO THROUGH ALL TESTS. THEREFORE, YOU WILL NOTICE
;* FIVE DISTINCT TESTS PREFORMED
;* AT THE END OF THIS TEST, THE MICRO-CODE IS LISTED.
;* VARIOUS SCOPE POINTS DO EXIST IF YOU NEED THEM. IT IS NOT
;* COMMON PRACTICE TO USE THEM, HOWEVER, WHERE SOME USE OF THEM
;* COULD BE MADE, THEY ARE NOTED.
.SBTTL ***** TEST 11 *****

T11::
BIT #3,OPTYP ;IS THIS DMV
BEQ 10$ ;IF NOT GO TO 10;ELSE
JMP EXMDT ;EXIT TEST

10$:
T9BG:
JSR PC,MINITS
:****
: JUMP TO END OF TEST IF ERROR
:
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10045-.
CLRB @BSEL7
MOVB #DRUN,@BSEL0 ;REQUEST INTERFACE DIAGNOSTICS

25$:
JSR R5,WRDI ;WAIT FOR RDI TO SET

CLR ERRWRD ;CLEAR ERROR
TRAP C$ESCAPE
.WORD L10045-.
:
:
: TIME OUT OR READY ERROR REPORTS THIS
: ADDRESS AS FAILING PC
:
:
CLRB @BSEL0 ;NO MORE REQUESTS.
MOVB #22,@BSEL6 ;DIAGNOSTIC CODE.
CLRB @BSEL7 ;CLEAR BSEL7
MOVB #1,@BSEL2 ;START.!

TLB2:
MOV #ERLB2,ERRADD ;SET UP ERROR ADDRESS
JSR R5,TOUT
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10045-.

```


CZDMTC.P11 25-MAR-81 08:24

***** TEST 11 *****

```

4328 020130 ERLB2:
4329
4330      : TIME OUT ERROR REPORTS THIS ADDRESS
4331      :
4332 020130 122777 000377 162320 26$: CMPB #377,@BSEL7 ;LOOK FOR SYNC OF CODE 377 IN LAST REG
4333 020136 001363 BNE TLB2 ;IF 'HANG' OCCURS HERE THEN ITS POSSIBLE
4334 ;THAT EITHER THE DATA PATHS ARE BAD OR
4335 ;THAT THE MCPU FAILED TO START
4336
4337 020140 012737 000377 002336 MOV #377,$GDDAT ;EXPECT 377 BACK FROM ALL REGS
4338 020146 013701 002440 MOV BSEL0,R1 ;EXCEPT REG 1 (MAINTENANCE)
4339 020152 012737 000000 002344 MOV #0,REG
4340 020160 111137 002340 4$: MOVB (R1),$BDDAT ;READ REG, EXPECT 377
4341 020164 123737 002340 002336 CMPB $BDDAT,$GDDAT ;OK?
4342 020172 001412 BEQ 5$ ;YES-CONTINUE
4343 020174 022737 000001 002344 CMP #1,REG ;NO ERROR? (EXCEPT REG 1)
4344 020202 001406 BEQ 5$ ;IF REG 1, SKIP
4345
4346 020204 104455 TRAP C$ERDF
4347 020206 000034 .WORD 28
4348 020210 011200 .WORD MEFC
4349 020212 007430 .WORD ERR5
4350 020214 104410 TRAP C$ESCAPE
4351 020216 000716 .WORD L10045-.
4352 020220 005237 002344 5$: INC REG ;UPDATE REGISTER #
4353 020224 005201 INC R1 ;AND ADDRESS
4354 020226 023727 002344 000010 CMP REG,#10 ;DONE ALL REGS?
4355 020234 001351 BNE 4$
4356 020236 105077 162176 CLRB @BSEL0 ;CAUSES MCPU TO EXIT TSTA
4357
4358
4359 ;TEST B
4360
4361
4362 020242 TLB3:
4363 020242 012737 020264 002406 MOV #ERLB3,ERRADD ;SET ERROR ADDRESS
4364 020250 004537 005070 JSR R5,TOUT
4365 020254 005037 002402 CLR ERRWRD
4366 020260 104410 TRAP C$ESCAPE
4367 020262 000652 .WORD L10045-.
4368 020264 ERLB3:
4369
4370      : TIME OUT ERROR REPORTS THIS ADDRESS
4371      :
4372 020264 105777 162166 27$: TSTB @BSEL7 ;LOOK FOR A ZERO IN BSEL7
4373 020270 001364 BNE TLB3
4374
4375 020272 005037 002336 CLR $GDDAT ;EXPECT ALL ZEROS EXCEPT SBEL1
4376 020276 013701 002440 MOV BSEL0,R1 ;GET ADDR OF MCPU.
4377 020302 012737 000000 002344 MOV #0,REG
4378 020310 005037 002340 CLR $BDDAT
4379 020314 111137 002340 7$: MOVB (R1),$BDDAT ;READ REG
4380 020320 001412 BEQ 8$ ;IF ZERO-CONTINUE
4381 020322 022737 000001 002344 CMP #1,REG ;IF REG #1 CONTINUE
4382 020330 001406 BEQ 8$
4383

```

C:DMTC.P11 25-MAR-81 08:24

:***** TEST 11 *****

```

4384 020332 104455 TRAP C$ERDF
4385 020334 000035 .WORD 29
4386 020336 011200 .WORD MEFC
4387 020340 007430 .WORD ERR5
4388 020342 104410 TRAP C$ESCAPE
4389 020344 000570 .WORD L10045-.
4390 020346 005237 002344 8$: INC REG ;UPDATE REGISTER #
4391 020352 005201 INC R1 ;AND ADDRESS
4392 020354 122737 000010 002344 CMPB #10,REG ;DONE ALL REGS (0-7)?
4393 020362 001354 BNE 7$ ;NO-DO NEXT ONE
4394
4395 020364 000404 BR 9$ ;REPLACE THIS INSTRUCTION WITH CODE 240
4396 ;(NOP) IF YOU WITH TO COOP IN
4397 ;TESTS A&B
4398 020366 112777 000200 162044 MOVB #200,@BSEL0 ;ALL MICRO-CODE TO LOOP
4399 020374 000611 BR T9BG ;LOOP
4400
4401 020376 112777 000377 162034 9$: MOVB #377,@BSEL0 ;TELL MICRO-CODE TO EXIT TEST B,
4402 ;PROCEED TO TEST C.
4403
4404
4405 ;TEST C
4406
4407 020404 017746 162060 MOV @KMLVL,-(SP)
4408 020410 012746 021044 MOV #INTC0,-(SP)
4409 020414 013746 002462 MOV KMRVEC,-(SP)
4410 020420 012746 000003 MOV #3,-(SP)
4411 020424 104437 TRAP C$$SVEC
4412 020426 062706 000010 ADD #10,SP
4413 ;INTERRUPT VECTOR
4414
4415 020432 017746 162032 MOV @KMTLVL,-(SP)
4416 020436 012746 021060 MOV #INTC4,-(SP)
4417 020442 013746 002464 MOV KMTVEC,-(SP)
4418 020446 012746 000003 MOV #3,-(SP)
4419 020452 104437 TRAP C$$SVEC
4420 020454 062706 000010 ADD #10,SP
4421 ;ILLEGAL INTERRUPT TO WRONG VECTOR
4422
4423 020460 005037 002334 CLR IFLAG
4424 020464 112777 000377 161754 MOVB #377,@BSEL3 ;TELL MICRO-CODE TO FORCE INTERRUPT
4425 020472 012700 000000 MOV #0,R0
4426 020476 104441 TRAP C$$SPRI
4427
4428 020500 TLB4: MOV #ERLB4,ERRADD ;SET UP ERROR ADDRESS
4429 020500 012737 020522 002406 JSR R5,TOUT
4430 020506 004537 005070 CLR ERRWRD
4431 020512 005037 002402 TRAP C$ESCAPE
4432 020516 104410 .WORD L10045-.
4433 020520 000414
4434 020522 ERLB4:
4435 ;TIME OUT REPORTS THIS ADDRESS
4436
4437
4438 020522 005737 002334 28$: TST IFLAG ;IFLAG=1 SET BY INTERRUPT SERVICE ROUTINE
4439 020526 001764 BEQ TLB4 ;LOOP UNIT DONE

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 11 *****

;NOTE: IF HANGS HERE, MCPU FAILS TO
;GENERATE INTERRUPT TO PDP-11.

```

4440
4441
4442
4443
4444
4445
4446 020530 017746 161734      MOV    @KMLVL,-(SP)
4447 020534 012746 021100      MOV    #INTD0,-(SP)
4448 020540 013746 002462      MOV    KMRVEC,-(SP)
4449 020544 012746 000003      MOV    #3,-(SP)
4450 020550 104437                TRAP   C$SVEC
4451 020552 062706 000010      ADD    #10,SP
4452 020556 017746 161706      MOV    @KMTLVL,-(SP)
4453 020562 012746 021120      MOV    #INTD4,-(SP)
4454 020566 013746 002464      MOV    KMTVEC,-(SP)
4455 020572 012746 000003      MOV    #3,-(SP)
4456 020576 104437                TRAP   C$SVEC
4457 020600 062706 000010      ADD    #10,SP
4458
4459 020604 005037 002334      CLR    IFLAG                ;NO INTERRUPT INDICATOR
4460 020610 012700 000000      MOV    #0,R0
4461 020614 104441                TRAP   C$SPRI
4462 020616 105077 161624      CLR    @BSEL3                ;TELL MCPU TO INTERRUPT
4463
4464 020622                TLB5:
4465 020622 012737 020644 002406      MOV    #ERLB5,ERRADD        ;SET UP ERROR ADDRESS
4466 020630 004537 005070                JSR    R5,TOUT
4467 020634 005037 002402                CLR    ERRWRD
4468 020640 104410                TRAP   C$ESCAPE
4469 020642 000272                .WORD  L10045-
4470 020644                ERLB5:
4471                ;:~::~:
4472                ;:TIME OUT REPORTS THIS ADDRESS
4473                ;:~::~:
4474 020644 005737 002334      29$:  TST    IFLAG                ;DID MCPU INTERRUPT (IFLAG NOT 0)?
4475 020650 001764                BEQ    TLB5                    ;NO - LOOP
4476                ;NOTE: IF PROGRAM 'HANGS' HERE, MCPU
4477                ;FAILED TO INTERRUPT TO VECTOR XX4
4478                ;WE KNOW MCPU IS ABLE TO INTERRUPT
4479                ;TO XX0 (TEST C)
4480
4481 020652 013700 002462      MOV    KMRVEC,R0
4482 020656 104436                TRAP   C$CVEC
4483 020660 013700 002464      MOV    KMTVEC,R0
4484 020664 104436                TRAP   C$CVEC
4485
4486
4487                ;TEST E NPR TEST
4488
4489
4490 020666 012701 033714      MOV    #DATLST,R1            ;GET DATA LIST
4491 020672 152777 000010 161542      BIS    #BIT3,@BSEL1          ;SET INDICATOR THAT WE ARE STILL IN TEST.
4492
4493 020700 011137 002336      TLB6A: MOV    (R1),%GDDAT          ;GET NEXT PATTERN
4494 020704 010177 161540      MOV    R1,@BSEL4            ;SET NPR IN ADDR
4495 020710 012777 002340 161536      MOV    #%BDDAT,@BSEL6       ;SET NPR OUT ADDR

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 11 *****

```

4496 020716 105077 161516          CLRB    @BSELO          ;TELL MCPU TO DO NPRS
4497
4498 020722
4499 020722 012737 020744 002406  TLB6:  MOV    #ERLB6,ERRADD ;SET ERROR ADDRESS
4500 020730 004537 005070          JSR    R5,TOUT
4501 020734 005037 002402          CLR    ERRWRD
4502 020740 104410          TRAP  C$ESCAPE
4503 020742 000172          .WORD L10045-.
4504 020744          ERLB6:
4505          ;:TIME OUT ERROR REPORTS THIS ADDRESS
4506
4507          ;:
4508 020744 132777 000010 161470 30$:  BITB   #BIT3,@BSEL1    ;DID WE ACCIDENTILY ESCAPE THIS TEST???
4509 020752 001006          BNE   135$
4510 020754 104455          TRAP  C$ERDF
4511 020756 000036          .WORD 30
4512 020760 011414          .WORD MEF7
4513 020762 010422          .WORD ERR32
4514          ;UNKNOWN MCPU ERROR CAUSED ABORT OF TEST.
4515 020764 104410          TRAP  C$ESCAPE
4516 020766 000146          .WORD L10045-.
4517 020770          135$:
4518
4519 020770 122777 000377 161442      CMPB  #377,@BSELO     ;WHEN MCPU DONE, IT PUTS 377 INTO BSELO
4520 020776 001351          BNE  TLB6            ;IF WE 'HANG' HERE, MCPU FAILS TO DO
4521          ;EITHER NPR IN OR NPR OUT
4522
4523 021000 023737 002336 002340      CMP   $GDDAT,$BDDAT ;NPRED FRO PATTERN LIST TO $BDDAT
4524          ;DID XFER OCCUR SUCCESSFULLY?
4525 021006 001406          BEQ  14$
4526
4527 021010 104455          TRAP  C$ERDF
4528 021012 000037          .WORD 31
4529 021014 011200          .WORD MEF7
4530 021016 007466          .WORD ERR6
4531 021020 104410          TRAP  C$ESCAPE
4532 021022 000112          .WORD L10045-.
4533 021024 022721 000562          14$:  CMP   #562,(R1)+     ;IS IT THE LAST PATTERN (562) IS TERM)?
4534 021030 001323          BNE  TLB6A
4535 021032 112777 000200 161400      MOVB  #200,@BSELO    ;TELL MCPU TO EXIT TEST
4536 021040 104432          TRAP  C$EXIT
4537 021042 000072          .WORD L10045-.
4538
4539 021044          INTC0::
4540 021044 013700 000006          MOV   6,R0
4541 021050 104441          TRAP  C$SPRI
4542 021052 005237 002334          INC   IFLAG
4543 021056          L10046:
4544 021056 000002          RTI
4545
4546 021060          INTC4::
4547 021060 013700 000006          MOV   6,R0
4548 021064 104441          TRAP  C$SPRI
4549 021066 104455          TRAP  C$ERDF
4550 021070 000040          .WORD 32
4551 021072 011446          .WORD MEF8
    
```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 11 *****

4552 021074 010422
 4553 021076
 4554 021076 000002
 4555
 4556 021100
 4557 021100 013700 000006
 4558 021104 104441
 4559 021106 104455
 4560 021110 000041
 4561 021112 011446
 4562 021114 010422
 4563 021116
 4564 021116 000002
 4565
 4566 021120
 4567 021120 013700 000006
 4568 021124 104441
 4569 021126 005237 002334
 4570 021132
 4571 021132 000002
 4572
 4573 021134
 4574
 4575 021134
 4576 021134 104401
 4577
 4578

L10047: .WORD ERR32
 RTI
 INTD0::
 MOV 6,R0
 TRAP C\$SPRI
 TRAP C\$ERDF
 .WORD 33
 .WORD ME:8
 .WORD ERR32
 L10050:
 RTI
 INTD4::
 MOV 6,R0
 TRAP C\$SPRI
 INC IFLAG
 L10051:
 RTI
 EXMDT:
 L10045:
 TRAP C\$ETST

;CORRECT VECTOR (XX4)

CZDMTC.P11 25-MAR-81 08:24

***** TEST 12 *****

4579
4580
4581 021136
4582
4583
4584
4585
4586
4587
4588
4589
4590 021136
4591
4592 021136 004737 004522
4593
4594 021142 005037 002402
4595 021146 104410
4596 021150 000076
4597
4598
4599
4600
4601 021152
4602
4603 021152 052777 000200 161260
4604
4605 021160
4606 021160 004537 002732
4607
4608 021164 005037 002402
4609 021170 104410
4610 021172 000054
4611
4612
4613
4614
4615
4616 021174 005077 161254
4617
4618
4619 021200 112777 000001 161236
4620
4621 021206 004737 004244
4622 021212 004737 004244
4623 021216 004737 004244
4624 021222 004737 004244
4625
4626
4627 021226 032777 000020 161210
4628 021234 001004
4629
4630 021236 104455
4631 021240 000042
4632 021242 011630
4633 021244 007524
4634

```

.SBTTL ***** TEST 12 *****
.SBTTL RDI REMAINS SET TEST
ZZ
:*
:*ROM FUNCTION TEST      IN THIS TEST, WE'RE GOING TO SET RQI, GET A
:*                        RDI, DO A CONTROL IN COMMAND WITH A REQUEST
:*                        KEY OF 00 (NO REQUEST). NEXT WE'LL WAIT
:*                        FOR RDI TO SET AGAIN SINCE RQI WAS
:*                        LEFT SET
.SBTTL ***** TEST 12 *****
T12::
JSR      PC,MINITS      ;INITIALIZE & START MCPU
CLR      ERRWRD
TRAP    C$ESCAPE
.WORD   L10052-.
:.....
: JUMP TO END OF TEST IF ERROR
:.....
7$:
BIS      #RQI,@BSEL0    ;SET RQI
10$:
JSR      R5,WRDI        ;WAIT FOR RDI TO SET
CLR      ERRWRD        ;CLEAR ERROR
TRAP    C$ESCAPE
.WORD   L10052-.
:.....
: TIME OUT - OR READY ERROR REPORTS
:                        THIS ADDRESS
:.....
20$:
CLR      @BSEL6        ;CLEAR RDI, ISSUE REQUEST OF NO REQUEST
:THIS SHOULD CAUSE RDI TO SET AGAIN
:SINCE RQI HAD REMAINED SET
:START
MOVB    #1,@BSEL2
JSR      PC,WAIT50      ;WAIT THIS SHORT TIME SO THAT THE
JSR      PC,WAIT50      ;DMP MICRO-CODE MAY RESET 'RDI' IF
JSR      PC,WAIT50
JSR      PC,WAIT50
:IT IS GOING TO
BIT      #RDI,@BSEL2   ;IS RDI SET?
BNE     30$
TRAP    C$ERDF
.WORD   34
.WORD   MRFT
.WORD   ERR9
;COMPLETING A 'NO REQUEST' CONTROL

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 12 *****

:IN COMMAND

4635
4636
4637 021246
4638 021246
4639 021246 104401

30\$:
L10052:
TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 12 *****

4640
 4641
 4642
 4643 021250
 4644
 4645
 4646
 4647
 4648
 4649
 4650 021250
 4651
 4652 021250 004737 004522
 4653
 4654 021254 005037 002402
 4655 021260 104410
 4656 021262 000174
 4657
 4658
 4659
 4660
 4661 021264 052777 000200 161146
 4662 021272 004537 002732
 4663
 4664 021276 005037 002402
 4665 021302 104410
 4666 021304 000152
 4667
 4668
 4669
 4670
 4671 021306 042777 000200 161124 20\$:
 4672 021314 012777 000020 161132
 4673 021322 112777 000001 161114
 4674
 4675 021330 004737 004244
 4676 021334 004737 004244
 4677
 4678 021340 032777 000200 161076
 4679 021346 001006
 4680
 4681 021350 104455
 4682 021352 000043
 4683 021354 011630
 4684 021356 007552
 4685
 4686 021360 104410
 4687 021362 000074
 4688
 4689 021364 117737 161054 002340 30\$:
 4690 021372 042737 177770 002340
 4691 021400 012737 000002 002336
 4692 021406 023737 002336 002340
 4693 021414 001411
 4694 021416 012737 012007 002430
 4695 021424 104455

```

.SBTTL :***** TEST 13 *****
.SBTTL *ROM FUNC TEST. VERIFY RDO SETS
ZZ
:*ROM FUNC      IN THIS TEST WE'LL DO A CONTROL IN WITH
:*              READ MODEM AS THE REQUEST KEY. WE'LL MAKE
:*              SURE THAT RDO SETS. WE SHOULD GET A
:*              RETURN KEY OF 10 'RETURN MODEM'
.SBTTL :***** TEST 13 *****
T13::
JSR      PC,MINITS      ;INIT & START MCPU

CLR      ERRWRD
TRAP     C$ESCAPE
.WORD    L10053-.
:
: JUMP TO END OF TEST IF ERROR
:
BIS      #RQI,@BSELO    ;SET REQUEST IN
JSR      R5,WRDI       ;WAIT FOR RDI TO SET

CLR      ERRWRD        ;CLEAR ERROR
TRAP     C$ESCAPE
.WORD    L10053-.
:
: TIME OUT OR READY ERROR REPORTS
: THIS ADDRESS AS FAILING PC
:
BIC      #RQI,@BSELO    ;DROP REQUEST
MOV      #20,@BSEL6    ;READ MODEM
MOVB     #1,@BSEL2

JSR      PC,WAIT50     ;STALL
JSR      PC,WAIT50

BIT      #RDO,@BSEL2   ;DID 'RDO' SET?
BNE      30$

TRAP     C$ERDF
.WORD    35
.WORD    MRFT
.WORD    ERR10
:
: REQUEST FOR READ MODEM

TRAP     C$ESCAPE
.WORD    L10053-.

MOVB     @BSEL2,$BDDAT ;NOW GET CSR AND
BIC      #^C<7>,$BDDAT ;STRIP FOR
MOV      #2,$GDDAT    ;TYPE CODE OF INFORMATION OUT
CMP      $GDDAT,$BDDAT
BEQ      40$
MOV      #M28F,CODEW
TRAP     C$ERDF

```


CZDMTC.P11 25-MAR-81 08:24

;***** TEST 13 *****

4696 021426 000044
 4697 021430 011506
 4698 021432 010324
 4699
 4700 021434 104410
 4701 021436 000020
 4702
 4703 021440
 4704 021440 112737 000010 002336
 4705 021446 004537 003372
 4706
 4707 021452 005037 002402
 4708 021456
 4709 021456 104401

.WORD 36
 .WORD EROIC
 .WORD ERR27

;IN RDO

TRAP C\$ESCAPE
 .WORD L10053-

40\$:

MOVB #10, \$GDDAT ; SHOULD=10 'RETURN MODFM'
 JSR R5,GETRKY ; GO GET AND CHECK RETURN KEY

_10053:

CLR ERRWRD
 TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

..... TEST 14

```

4710
4711
4712 021460
4713
4714
4715
4716
4717
4718
4719
4720 021460
4721
4722 021460 112777 000100 160754
4723 021466 022737 000004 002472
4724 021474 001003
4725 021476 112777 000200 160736
4726
4727 021504
4728 021504 004537 005070
4729 021510 005037 002402
4730 021514 104410
4731 021516 000204
4732 021520 005777 160714
4733 021524 100367
4734 021526 052777 000200 160704
4735 021534 004537 002732
4736
4737 021540 005037 002402
4738 021544 104410
4739 021546 000154
4740 021550 105077 160664
4741 021554 105077 160666
4742 021560 112777 000052 160666
4743 021566 112777 000001 160650
4744 021574 004537 002646
4745 021600 005037 002402
4746 021604 104410
4747 021606 000114
4748 021610 117737 160630 002340
4749 021616 042737 177770 002340
4750 021624 022737 000002 002340
4751 021632 001433
4752 021634 022737 000001 002340
4753 021642 001411
4754 021644 012737 012000 002430
4755 021652 104455
4756 021654 000045
4757 021656 011506
4758 021660 010324
4759 021662 104410
4760 021664 000036
4761 021666 117737 160562 002340
4762 021674 022737 000100 002340
4763 021702 001407
4764 021704 012737 011762 002430
4765 021712 104455

```

```

.SBTTL :..... TEST 14 .....
.SBTTL * NON-MODE DEF AFTER MC PROCEDURE ERR CHECK
ZZ
*
* THIS TEST CHECKS FOR PROCEDURE ERROR WHEN
* NON-MODE DEFINITION IS DONE AFTER MC
*
.SBTTL :..... TEST 14 .....
T14::
MOV# #100, @BSEL1 :MASTER CLEAR
CMP #4, OPTYP :8206
BNE TLB10 :IF NOT GO TO TLB10
MOV# #200, @BSEL1 :SET RUN 8206

TLB10:
JSR R5, TOUT
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10054-
ERLB10: TST @BSEL0
BPL TLB10 :LOOP IF NOT RUN
BIS #R01, @BSEL0 :SET REQUEST
JSR R5, WRDI :WAIT FOR RDI TO SET

CLR ERRWRD :CLEAR ERROR
TRAP C$ESCAPE
.WORD L10054-
CLRB @BSEL0 :CLEAR REQUEST
CLRB @BSEL3 :MAKE TRIB ADD 0
MOV# #52, @BSEL6 :READ TSS
MOV# #01, @BSEL2 :EXECUTE CONTROL IN
JSR R5, WRDO :WAIT FOR RDO
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10054-
MOV# @BSEL2, @BDDAT :STRIP TO COMMAND CODE
BIC #<C<7>, @BDDAT :IS IT INFO OUT
CMP #2, @BDDAT :IF YES EXIT TEST
BEQ T14EX :IF YES EXIT TEST
CMP #01, @BDDAT :IF NOT IS IT CONTROL OUT
BEQ T14A
MOV #M1BF, CODEW
TRAP C$ERDF
.WORD 37
.WORD EROIC
.WORD ERR27
TRAP C$ESCAPE
.WORD L10054-
T14A: MOV# @BSEL6, @BDDAT :IS IT '00
CMP #100, @BDDAT :IF SO END TEST ELSE ERROR
BEQ T14EX
MOV #M13F, CODEW
TRAP C$ERDF

```

CZDMTC.P11 25-MAR-81 08:24

:..... TEST 14

4766	021714	000046
4767	021716	011506
4768	021720	010324
4769	021722	
4770	021722	
4771	021722	10440*
4772		

.WORD	38
.WORD	ERCIC
.WORD	ERR27

T14EX:
L10054:

TRAP CSETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 14 *****

4773
4774
4775
4776 021724
4777
4778
4779
4780
4781
4782
4783
4784
4785
4786
4787 021724
4788
4789 021724 004737 004522
4790
4791
4792 021730 005037 002402
4793 021734 104410
4794 021736 000174
4795
4796
4797
4798 021740 004737 004244
4799
4800 021744 004737 004244
4801 021750 142777 000010 160464
4802 021756 022737 000004 002476
4803 021764 001027
4804 021766 032777 000200 160450
4805 021774 001423
4806 021776 012737 000304 002336
4807 022004 117737 160444 002340
4808 022012 023737 002340 002336
4809 022020 001406
4810 022022 104455
4811 022024 000047
4812 022026 012516
4813 022030 010422
4814 022032 104432
4815 022034 000076
4816 022036
4817 022036 042777 000200 160400
4818 022044 052777 000200 160366
4819 022052 004537 002732
4820
4821 022056 005037 002402
4822 022062 104410
4823 022064 000046
4824 022066 043777 000200 160344
4825 022074 012777 000007 160352
4826 022102 012777 000002 160334
4827 022110 012737 000104 002410
4828

```
.SBTTL :***** TEST 15 *****
.SBTTL * MODE DEF,MODE DEF PROCEDURE ERROR
ZZ
:*
:*
:*
:* THIS TEST CHECKS THAT AFTER THE SEQUENCE OF
:* MASTER CLEAR MODE DEF FOLLOWED BY MODE DEF
:* DIFFERENT TYPE PRODUCES A PROCEDURE ERROR OF
:* OCTAL 104
:*
*:-
:***** TEST 15 *****
```

```
.SBTTL :***** TEST 15 *****
T15::
JSR PC,MINITS ;DO MC,MODE DEF(CONT STA/FD)

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10055-.
: JUMP TO END OF TEST IF ERROR
:
JSR PC,WAIT50 ;WAIT A WHILE TO BE SURE MODEM READY
:IS SET

JSR PC,WAIT50
BIC #BIT3,@BSEL1 ;CLEAR LU LOOP
CMP #4,TSTCON ;IS THIS NO LOOPBACK
BNE 20$ ;IF LOOPBACK GO TO 20
BIT #RDO,@BSEL2 ;IS RDO SET?
BEQ 20$ ;IF NOT GO TO 20
MOV #304,$GDDAT
MOV @BSEL6,$BDDAT ;IF YES.IS IT 304
CMP $BDDAT,$GDDAT
BEQ 25$ ;IF EQUAL GO TO 25
TRAP C$ERDF
.WORD 39
.WORD MEF30
.WORD ERR32
TRAP C$EXIT
.WORD L10055-.

25$: BIC #RDO,@BSEL2 ;CLEAR RDO
20$: BIS #RQI,@BSEL0 ;SET REQUEST
JSR R5,WRDI ;WAIT FOR RDI TO SET

CLR ERRWRD ;CLEAR ERROR
TRAP C$ESCAPE
.WORD L10055-.
BIC RQI,@BSEL0 ; CLEAR REQUEST
MOV #7,@BSEL6 ;SET MODE FOR TRIB/FD
MOV #02,@BSEL2 ;DO MODE DEF
MOV #104,PERR ;SET PROCEDURE ERROR OF
; 104 TO BE CHECKED
```

CZDMTC.P11 25-MAR-81 08:24

;***** TEST 15 *****

4829 022116 004537 003016
 4830
 4831 022122 005037 002402
 4832 022126 104410
 4833 022130 000002
 4834
 4835
 4836
 4837
 4838
 4839 022132
 4840 022132 104401

JSR R5,WFPE ;WAIT FOR PROCEDURE ERROR

CLR ERRWRD
 TRAP C\$ESCAPE
 .WORD L10055-

.....
 : ESCAPE TEST IF ERROR

; ERROR,OR TIME OUT.

L10055:
 TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 15 *****

4841
4842
4843
4844
4845
4846
4847
4848
4849
4850
4851
4852
4853
4854
4855
4856
4857
4858
4859
4860
4861
4862
4863
4864
4865
4866
4867
4868
4869
4870
4871
4872
4873
4874
4875
4876
4877
4878
4879
4880
4881
4882
4883
4884
4885
4886
4887
4888
4889
4890
4891
4892
4893
4894
4895
4896

022134

022134

004737 004522
004737 004244
005037 002402
104410
00212

022154 022737 000003 002476
022162 001011
022164 012704 000104
022170 012703 000021
022174 004537 003162

022200 005737 002402
022204 100467

022206 142777 000010 160226

022214 004737 004244
022220 004737 004244
022224 022737 000004 002476
022232 001020
022234 032777 000200 160202
022242 001414
022244 012737 000304 002410
022252 004537 003016

022256 005037 002402
022262 104410
022264 000100

042777 000200 160150
052777 000200 160136
004537 002732

022306 005037 002402
022312 104410
022314 000050

```
.SBTTL ***** TEST 16 *****
.SBTTL * MODE DEF ,MODE DEF CHANGE DUPLEX ONLY.
ZZ
:*
:*
:* THIS CHECKS THAT YOU CAN CHANGE THE DUPLEX PORTION
:* OF A MODE DEF
:*
:*-
.SBTTL ***** TEST 16 *****
T16::
JSR PC,MINITS ;MC,MODE DEF(CONT/FD)
JSR PC,WAIT50 ;DELAY

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10056-.

; JUMP TO END OF TEST IF ERROR

CMP #3,TSTCON ;IS IT REMOTE MODEM
BNE 2$ ;IF NOT THEN GO TO 2A
MOV #104,R4
MOV #21,R3
JSR R5,CONTIN ; WRITE MODEM WITH CORRECT
; TYPE OF LOOP CODE

TST ERRWRD
BMI 10$ ;EXIT IF ERROR

BICB #BIT3,@BSEL1 ;CLEAR LU LOOP

JSR PC,WAIT50 ;WAIT A WHILE
JSR PC,WAIT50
CMP #4,TSTCON
BNE 20$ ;IF LOOPBACK GO TO 20
BIT #RDO,@BSEL2 ;ELSE SEE IF READY OUT
BEQ 20$ ;IF NOT GO TO 20
MOV #304,PERR
JSR R5,WFPD ;WAIT FOR PROCEDURE ERROR

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10056-.

;:;:;:;:
;: ESCAPE TEST IF ERROR
;:;:;:;:
BIC #RDO,@BSEL2 ;CLEAR OUTPUT
BIS #RQI,@BSEL0 ;SET REQUEST
JSR R5,WRDI ;WAIT FOR RDI TO SET

CLR ERRWRD ;CLEAR ERROR
TRAP C$ESCAPE
.WORD L10056-.

1$:
2$:
20$:
```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 16 *****

```

4897
4898
4899
4900 022316 043777 000200 160114
4901 022324 112777 000004 160122
4902 022332 112777 000002 160104
4903 022340 004737 004244
4904 022344 032777 000200 160072
4905 022352 001404
4906 022354 104455
4907 022356 000050
4908 022360 012516
4909 022362 010422
4910 022364
4911 022364
4912 022364 104401
4913

```

```

: TIME OUT OR READY ERRORS REPORT THIS PC
:*****:
BIC RQ1,@BSL0 ;NO MORE REQUESTS
MOVB #04,@BSL6 ;CONT/FD FOR MODE
MOVB #02,@BSL2 ;DO MODE DEF
JSR PC,WAIT50 ;DELAY A WHILE
BIT #RDO,@BSL2 ;IS RDO SET
BEQ 10$ ;BRANCH IF NOT
TRAP C$ERDF
.WORD 40
.WORD MEF30
.WORD ERR32

```

```

10$:
L10056: TRAP C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 17 *****

4914
4915
4916 022366
4917
4918
4919
4920
4921
4922
4923
4924
4925
4926
4927 022366
4928 022366 012737 000040 002362
4929 022374 032737 000003 002472
4930 022402 001403
4931 022404 012737 000014 002362
4932 022412
4933
4934 022412 112737 000161 002360
4935 022420 112737 000161 002364
4936 022426 063737 002362 002364
4937
4938 022434 004737 004522
4939
4940 022440 005037 002402
4941 022444 104410
4942 022446 000150
4943
4944
4945
4946 022450
4947 022450 112703 000001
4948 022454 004537 003162
4949
4950
4951
4952
4953 022460 005037 002402
4954 022464 104410
4955 022466 000130
4956
4957
4958
4959 022470 005237 002360
4960 022474 023737 002364 002360
4961 022502 001362
4962
4963 022504
4964 022504 112703 000001
4965 022510 004537 003162
4966
4967
4968
4969

```

.SBTTL ***** TEST 17 *****
.SBTTL *ROM FUNC. TEST. VERIFY THAT MAX TRIBS CAN BE ESTABLISHED
ZZ
*ROM FUNCTION TEST-ESTABLISHING TRIBS-
* THIS TEST WILL ESTABLISH MAX TRIBS
* THEN TRY TO ESTABLISH MAX+1 TRIBS
* AND CHECK FOR PROCEDURE ERROR.
* THE TEST ALSO CHECKS FOR PROCEDURE
* ERROR WHEN TRYING TO ESTABLISH AN
* ALREADY ESTABLISHED TRIB.
.SBTTL ***** TEST 17 *****

T17::
MOV #32.,TRIBMX ;SET MAX TRIB TO 32
BIT #3,OPTYP ;IS THIS DMV
BEQ XX ;IF NOT BRANCH
MOV #12.,TRIBMX ;ELSE SET THE MAX TO 12

XX:
MOVB #161,TRIBN ;NUMBER OF TRIBUTARY,
MOVB #161,TRIBH ;START TRIB HIGH AT SAME AS TRIBN
ADD TRIBMX,TRIBH ;ADD MAX NUMBER OF TRIBS TO TRIBH

JSR PC,MINITS ;INITIALIZE

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10057-.
: JUMP TO END OF TEST IF ERROR
: JUMP TO END OF TEST IF ERROR
: JUMP TO END OF TEST IF ERROR

30$:
MOVB #01,R3 ;SET ESTABLISH TRIB
JSR R5,CONTIN ;
: READY OR TIME OUT ERRORS REPORT THIS PC
: READY OR TIME OUT ERRORS REPORT THIS PC

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10057-.
: JUMP TO END OF TEST IF ERROR
: JUMP TO END OF TEST IF ERROR

INC TRIBN ;UPDATE TRIB#
CMP TRIBH, TRIBN ;ONLY ALLOW MAX TRIBS TO BE SET
BNE 30$

37$:
MOVB #01, R3 ;ESTABLISH MAX +1 TRIBS
JSR R5,CONTIN ; DO IT
: READY OR TIME OUT ERRORS REPORT THIS PC
: READY OR TIME OUT ERRORS REPORT THIS PC

```


CZDMTC.P11 25-MAR-81 08:24

***** TEST 17 *****

4970 022514 005037 002402
 4971 022520 104410
 4972 022522 000074
 4973
 4974
 4975
 4976
 4977 022524 012737 000114 002410
 4978
 4979
 4980 022532 004537 003016
 4981
 4982 022536 005037 002402
 4983 022542 104410
 4984 022544 000052
 4985
 4986
 4987
 4988 022546
 4989 022546 042777 000200 157670
 4990 022554 005337 002360
 4991 022560 112703 000001
 4992 022564 004537 003162
 4993
 4994
 4995
 4996
 4997 022570 005037 002402
 4998 022574 104410
 4999 022576 000020
 5000
 5001
 5002
 5003
 5004 022600 112737 000116 002410
 5005
 5006
 5007 022606 004537 003016
 5008 022612 005037 002402
 5009 022616
 5010 022616
 5011 022616 104401
 5012

```

CLR   ERRWRD
TRAP  C$ESCAPE
.WORD L10057-.
:
:
: JUMP TO END OF TEST IF ERROR
:
:
MOV   #114, PERR ; SHOULD READ 114, PROCEDURE ERROR
: TRYING TO ESTABLISH MAX+1 TRIBUTARIES
:
JSR   R5,WFPE ; WAIT FOR PROCEDURE ERROR
:
CLR   ERRWRD
TRAP  C$ESCAPE
.WORD L10057-.
:
:
: ESCAPE TEST IF ERROR
:
:
BIC   #RDO,@BSEL2 ; CLEAR RDO
DEC   TRIBN ; DEC TRIB NUMBER
MOVB  #01, R3 ; SET ESTABLISH TRIB
JSR   R5,CONTIN ; DO IT
:
:
: READY OR TIME OUT ERRORS REPORT THIS PC
:
:
CLR   ERRWRD
TRAP  C$ESCAPE
.WORD L10057-.
:
:
: JUMP TO END OF TEST IF ERROR
:
:
MOVB  #116, PERR ; SHOULD BE PROCEDURE ERROR
: OF 116 ESTABLISH ALREADY
: ESTABLISHED TRIB.
: GO CHECK FOR PROCEDURE ERROR
:
JSR   R5,WFPE
CLR   ERRWRD
:
:
70$:
L10057: TRAP  C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

```

***** TEST 18 *****
.SBTTL ***** TEST 18 *****
.SBTTL * READ/WRITE TSS TEST
ZZ
:
:
: * THIS TEST CHECKS THAT A TRIB STATUS SLOT CAN
: * BE WRITTEN AND READ
:
:
: *
: ***** TEST 18 *****
T18: .SBTTL ***** TEST 18 *****
MOV #30,TSSADD ;START ADD AT 30
NEWSLT: CLR R2 ;CLEAR R2
NEWPAT: JSR PC,MINITS ;MASTER CLEAR MODE DEF
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10060-.
:
: JUMP TO END OF TEST IF ERROR
:
:
MOV #55,TRIBN ;PUT 55 IN TRIB NUMBER
MOV #01,R3 ;THIS WILL ESTABLISH
JSR R5,CONTIN ; A TRIB
:
: TIME OUT AND READY ERRORS REPORT THIS PC
:
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10060-.
:
: JUMP TO END OF TEST IF ERROR
:
:
MOV DATLST(R2),R4 ;PATTERN TO BE WRITTEN
MOV TSSADD,R3 ;WRITE TO TSS
BIS #BIT7,R3 ;SET THE WRITE BIT
JSR R5,CONTIN ; GO DO IT!!!
:
: TIME OUT AND READY ERRORS REPORT THIS PC
:
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10060-.
:
: JUMP TO END OF TEST IF ERROR
:
:
MOV TSSADD,R3 ;SET UP TO READ SLOT
BIS #BIT5,R3 ;SET THE READ BIT
JSR R5,CONTIN ;DO CONTROL IN
:
:

```

```

5013
5014
5015 022620
5016
5017
5018
5019
5020
5021
5022
5023 022620
5024 022620 012737 000030 002412
5025 022626 005002
5026
5027 022630 004737 004522
5028
5029 022634 005037 002402
5030 022640 104410
5031 022642 000240
5032
5033
5034
5035
5036 022644 012737 000055 002360
5037 022652 012703 000001
5038 022656 004537 003162
5039
5040
5041
5042
5043 022662 005037 002402
5044 022666 104410
5045 022670 000212
5046
5047
5048
5049
5050 022672 016204 033714
5051 022676 013703 002412
5052 022702 052703 000200
5053 022706 004537 003162
5054
5055
5056
5057
5058 022712 005037 002402
5059 022716 104410
5060 022720 000162
5061
5062
5063
5064
5065 022722 013703 002412
5066 022726 052703 000040
5067 022732 004537 003162
5068

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 18 *****

```

5069                                     ; TIME OUT AND READY ERRORS REPORT THIS PC
5070                                     ; : : : : :
5071                                     ; : : : : :
5072 022736 005037 002402                CLR   ERRWRD
5073 022742 104410                        TRAP  C$ESCAPE
5074 022744 000136                        .WORD L10060-.
5075                                     ; : : : : :
5076                                     ; JUMP TO END OF TEST IF ERROR
5077                                     ; : : : : :
5078 022746 012737 000002 002336         MOV   #02,$GDDAT ; COMPARE FOR A INFO OUT
5079 022754 004537 003242                JSR   R5,GETOUT  ; CHECK FOR INFO OUT AND
                                           ; CORRECT TRIB NO. IF ERROR
                                           ; REPORT THIS PC.
5080                                     ; : : : : :
5081                                     ; : : : : :
5082 022760 005037 002402                CLR   ERRWRD
5083 022764 104410                        TRAP  C$ESCAPE
5084 022766 000114                        .WORD L10060-.
5085                                     ; : : : : :
5086                                     ; JUMP TO END OF TEST IF ERROR
5087                                     ; : : : : :
5088                                     ; : : : : :
5089 022770 013737 002412 002336         MOV   TSSADD,$GDDAT ; MOVE EXPECTED ADDRESS TO GDDAT
5090 022776 052737 000040 002336         BIS   #BIT5,$GDDAT ; SET THE READ TSS BIT IN EXPECTED
5091 023004 004537 003372                JSR   R5,GETRKY  ; GO CHECK FOR GOOD RETURN KEY
5092                                     ; : : : : :
5093 023010 005037 002402                CLR   ERRWRD
5094 023014 104410                        TRAP  C$ESCAPE
5095 023016 000064                        .WORD L10060-.
5096                                     ; : : : : :
5097                                     ; JUMP TO END OF TEST IF ERROR
5098                                     ; : : : : :
5099                                     ; : : : : :
5100 023020 016237 033714 002336 30$:    MOV   DATLST(R2),$GDDAT ; MOVE EXPECTED PATTERN
5101 023026 004537 003446                JSR   R5,GETDAT  ; GET DATA RETURNED.
                                           ; IF ERROR REPORT THIS PC.
5102                                     ; : : : : :
5103 023032 005037 002402                CLR   ERRWRD
5104 023036 104410                        TRAP  C$ESCAPE
5105 023040 000042                        .WORD L10060-.
5106                                     ; : : : : :
5107                                     ; JUMP TO END OF TEST IF ERROR
5108                                     ; : : : : :
5109                                     ; : : : : :
5110 023042 022762 000562 033714         CMP   #562,DATLST(R2) ; ARE WE DONE WITH PATTERN
5111 023050 001404                        BEQ   50$        ; IF SO DO NEXT SLOT
5112 023052 062702 000002                ADD   #2,R2     ; BUMP LIST POINTER
5113 023056 000137 022630                JMP   NEWPAT    ; GO BACK FOR THIS PATTERN.
5114 023062 022737 000037 002412 50$:    CMP   #37,TSSADD ; IS THIS THE LAST SLOT
5115 023070 001404                        BEQ   60$        ; IF SO END TEST
5116 023072 005237 002412                INC   TSSADD    ; ELSE BUMP ADD
5117 023076 000137 022626                JMP   NEWSLT    ; AND DO NEXT SLOT
5118 023102                                     60$:
5119                                     ; : : : : :
5120 023102                                     L10060:
5121 023102 104401                        TRAP  C$ETST
5122

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 18 *****

5123
5124
5125
5126
5127
5128
5129
5130
5131
5132
5133
5134
5135
5136
5137
5138
5139
5140
5141
5142
5143
5144
5145
5146
5147
5148
5149
5150
5151
5152
5153
5154
5155
5156
5157
5158
5159
5160
5161
5162
5163
5164
5165
5166
5167
5168
5169
5170
5171
5172
5173
5174
5175
5176
5177
5178

023104

023104

023104 004737 004522

023110 005037 002402

023114 104410

023116 000072

023120 012737 000022 002360

023126 012703 000001

023132 004537 003162

023136 005037 002402

023142 104410

023144 000044

023146 012703 000204

023152 004537 003162

023156 005037 002402

023162 104410

023164 000024

023166 012737 000132 002410

023174 004537 003016

023200 005037 002402

023204 104410

023206 000002

```

.SBTTL ***** TEST 19 *****
.SBTTL *WRITE RESERVED AREA OF TSS. P.E. 132
ZZ
:
:
: * THIS TEST CHECKS FOR PROCEDURE ERROR
: * ON WRITING TO ILLEGAL SLOT
:
:
:***** TEST 19 *****
T19::
JSR PC,MINITS ;MASTER CLEAR MODE DEF

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10061-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #22,TRIBN ;SET TRIB NUMBER TO 22
MOV #01,R3
JSR R5,CONTIN ;ESTABLISH TRIB
:
: TIME OUT OR READY ERRORS REPORT THIS PC
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10061-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #204,R3 ;SEL6=204
JSR R5,CONTIN ;WRITE TSS (ILLEGAL)
:
: TIME OUT OR READY ERROR REPORT THIS PC
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10061-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #132,PERR ;CHECK FOR PROCEDURE ERROR
JSR R5,WFPD ;WAIT FOR PROCEDURE ERROR

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10061-.
:

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 19 *****

5179
5180
5181
5182 023210
5183 023210 104401
5184

; ESCAPE TEST IF ERROR
; ; ; ; ; ;

L10061: TRAP CSETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 19 *****

5185
5186
5187
5188
5189
5190
5191
5192
5193
5194
5195
5196
5197
5198
5199
5200
5201
5202
5203
5204
5205
5206
5207
5208
5209
5210
5211
5212
5213
5214
5215
5216
5217
5218
5219
5220
5221
5222
5223
5224
5225
5226
5227
5228
5229
5230
5231
5232
5233
5234
5235
5236
5237
5238
5239
5240

023212

023212

023212 004737 004522

023216 005037 002402

023222 104410

023224 000072

023226 012737 000077 002360

023234 012703 000001

023240 004537 003162

023244 005037 002402

023250 104410

023252 000044

023254 012703 000106

023260 004537 003162

023264 005037 002402

023270 104410

023272 000024

023274 012737 000132 002410

023302 004537 003016

023306 005037 002402

023312 104410

023314 000002

.SBTTL ***** TEST 20 *****
.SBTTL *READ CLEAR WRONG ADD P.E.132

ZZ

THIS TEST CHECKS FOR PROCEDURE ERROR
FOR RD/CLR TSS WRONG ADD(132)

.SBTTL ***** TEST 20 *****
120::

JSR PC,MINITS ;MASTER CLEAR MODE DEF

CLR ERRWRD

TRAP C\$ESCAPE

.WORD L10062-

: JUMP TO END OF TEST IF ERROR

MOV #77,TRIBN ;MAKE TRIBN 77

MOV #01,R3 ;ESTABLISH TRIB

JSR R5,CONTIN

: TIME OUT OR READY ERROR REPORTS THIS PC

CLR ERRWRD

TRAP C\$ESCAPE

.WORD L10062-

: JUMP TO END OF TEST IF ERROR

MOV #106,R3

JSR R5,CONTIN ;READ/CLEAR ADD 6

: TIME OUT OR READY ERRORS REPORT THIS PC

CLR ERRWRD

TRAP C\$ESCAPE

.WORD L10062-

: JUMP TO END OF TEST IF ERROR

MOV #132,PEER ;SET PROCEDURE ERROR TO

JSR R5,WFPE ;BE CHECKED TO 132

JSR R5,WFPE ;WAIT FOR PROCEDURE ERROR

CLR ERRWRD

TRAP C\$ESCAPE

.WORD L10062-

CZDMTC.P11 25-MAR-81 08:26

..... TEST 20

5241
5242
5243
5244
5245
5246

023316
023316 106401

.....
: ESCAPE TEST IF ERROR
:

L10062:
TRAP CBETST

CZDM C.P.: 25-MAR-81 08:24

..... TEST 20

5247
5248
5249
5250
5251 023320
5252
5253
5254
5255
5256
5257
5258 023320
5259
5260 023320 004737 004522
5261
5262 023324 005037 002402
5263 023330 104410
5264 023332 000100
5265
5266
5267
5268
5269 023334 012737 000003 002360
5270 023342 012703 000001
5271 023346 004537 003162
5272
5273
5274
5275
5276 023352 005037 002402
5277 023356 104410
5278 023360 000052
5279
5280
5281
5282
5283 023362 012703 000107
5284 023366 004537 003162
5285
5286
5287
5288
5289 023372 005037 002402
5290 023376 104410
5291 023400 000032
5292
5293
5294
5295
5296 023402 004737 004244
5297 023406 004737 004244
5298 023412 032777 000200 157024
5299 023420 001004
5300
5301 023422 104455
5302 023424 000051

```

.SBTTL ..... TEST 21 .....
.SBTTL READ/CLEAR TSS
ZZ
:
:
: THIS TEST THAT READ CLEAR WORKS
:
:
.SBTTL ..... TEST 21 .....
T21::
JSR PC,MINITS ;MASTER CLEAR MODE-DEF
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10063-
:
: JUMP TO END OF TEST IF ERROR
:
MOV #03,TRIBN ;SET TRIB NUMBER
MOV #01,R3
JSR R5,CONTIN ;ESTABLISH TRIB
:
: TIME OUT OR READY ERRORS REPORT THIS PC
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10063-
:
: JUMP TO END OF TEST IF ERROR
:
MOV #107,R3
JSR R5,CONTIN ;READ/CLEAR ADD 7
:
: TIME OUT OR READY ERRORS REPORT THIS PC
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10063-
:
: JUMP TO END OF TEST IF ERROR
:
JSR PC,WAIT50
JSR PC,WAIT50 ;DELAY
BIT #RDO,@BSEL2
BNE 10$ ;IF RDO THEN END
;ELSE ERROR
TRAP C$ERDF
.WORD 41

```


CZDMTC.P11 25-MAR-81 08:24

***** TEST 21 *****

5303 023426 011630
5304 023430 007552
5305
5306 023432
5307 023432
5308 023432 104401

.WORD MRFT
.WORD ERR10

10\$:
L10063:
TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

```

:***** TEST 22 *****
.SBTTL :***** TEST 22 *****
.SBTTL *GLOBAL STATUS SLOT TESTS
      ZZ
      *
      * THIS TEST CHECKS THAT GLOBAL STATUS
      * SLOTS RESPOND TO COMMANDS
      *
.SBTTL :***** TEST 22 *****
T22::
T22.1: TRAP C$BSUB
      :
      : READ ALL SLOTS TEST
      :
      CLR TSSADD ;CLEAR ADD
      CLR TRIBN ;MAKE TRIB #0(GLOBAL COMM)
5$: JSR PC,MINITS ;MASTER CLEAR INIT
      CLR ERRWRD
      TRAP C$ESCAPE
      .WORD L10065-.
      :
      : JUMP TO END OF TEST IF ERROR
      :
      MOV TSSADD,R3
      BIS #BIT5,R3 ;SET UP READ GSS COMMAND
      JSR R5,CONTIN ;GO DO IT
      CLR ERRWRD
      TRAP C$ESCAPE
      .WORD L10065-.
      :
      : JUMP TO END OF TEST IF ERROR
      :
      MOV #2,$GDDAT
      JSR R5,GETOUT ;CHECK CORRECT TYPE AND
      ; TRIB NO.
      CLR ERRWRD
      TRAP C$ESCAPE
      .WORD L10065-.
      :
      : ESCAPE SUB IF ERROR
      :
10$: MOV TSSADD,$GDDAT
      BIS #BIT5,$GDDAT ;SET THE READ TSS BIT
      JSR R5,GETRKY ;CHECK RETURN KEY

```

```

5309
5310
5311 023434
5312
5313
5314
5315
5316
5317
5318 023434
5319
5320 023434
5321 023434 104402
5322
5323
5324
5325
5326 023436 005037 002412
5327 023442 005037 002360
5328 023446
5329 023446 004737 004522
5330
5331 023452 005037 002402
5332 023456 104410
5333 023460 000114
5334
5335
5336
5337
5338 023462 013703 002412
5339 023466 052703 000040
5340 023472 004537 003162
5341
5342 023476 005037 002402
5343 023502 104410
5344 023504 000070
5345
5346
5347
5348
5349 023506 012737 000002 002336
5350 023514 004537 003242
5351
5352
5353 023520 005037 002402
5354 023524 104410
5355 023526 000046
5356
5357
5358
5359
5360 023530
5361 023530 013737 002412 002336
5362 023536 052737 000040 002336
5363 023544 004537 003372
5364

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 22 *****

```

5365 023550 005037 002402          CLR  ERRWRD
5366 023554 104410          TRAP C$ESCAPE
5367 023556 000016          .WORD L10065-.
5368                               :.:.:.:.:
5369                               : GO TO END OF SUB IF ERROR
5370                               :.:.:.:.:
5371
5372
5373 023560 005237 002412          30$: INC  TSSADD          ;BUMP ADDRESS
5374 023564 022737 000040 002412  CMP  #40,TSSADD      ; ARE WE ALL DONE
5375 023572 001325          BNE  5$              ; IF NOT GO BACK
5376                               ;ELSE END SUBTEST
5377 023574          L10065: TRAP  C$ESUB
5378 023574 104403          T22.2: TRAP  C$BSUB
5379 023576 104402          :.:.:.:.:
5380                               : WRITE ALL SLOTS TEST ;
5381                               :.:.:.:.:
5382
5383
5384
5385 023600 012737 000034 002412          MOV  #34,TSSADD      ;START WITH FIRST WRITABLE ADD
5386 023606 005037 002360          CLR  TRIBN          ;AND TRIBN AT ZERO
5387 023612 004737 004522          40$: JSR  PC,MINITS  ;MASTER CLEAR INT
5388
5389 023616 005037 002402          CLR  ERRWRD
5390 023622 104410          TRAP C$ESCAPE
5391 023624 000226          .WORD L10066-.
5392                               :.:.:.:.:
5393                               : JUMP TO END OF SUB IF ERROR
5394                               :.:.:.:.:
5395
5396 023626 013703 002412          45$: MOV  TSSADD,R3
5397 023632 052703 000200          BIS  #BIT7,R3        ;WRITE TSS(GLOBAL BECAUSE TRIBN=0)
5398 023636 013704 002412          MOV  TSSADD,R4      ;PUT IN ADD FOR DATA
5399 023642 004537 003162          JSR  R5,CONTIN      ;DO IT
5400
5401 023646 005037 002402          CLR  ERRWRD
5402 023652 104410          TRAP C$ESCAPE
5403 023654 000176          .WORD L10066-.
5404                               :.:.:.:.:
5405                               : JUMP TO END OF SUB IF ERROR
5406                               :.:.:.:.:
5407
5408 023656 005237 002412          INC  TSSADD          ;BUMP ADD
5409 023662 022737 000040 002412  CMP  #40,TSSADD      ;DONE ALL
5410 023670 001356          BNE  45$            ;NO GO FINISH!!!
5411 023672 012703 000227          MOV  #227,R3
5412 023676 013704 002412          MOV  TSSADD,R4
5413 023702 004537 003162          JSR  R5,CONTIN      ; TRY TO WRITE BAD ADDRESS
5414
5415 023706 005037 002402          CLR  ERRWRD
5416 023712 104410          TRAP C$ESCAPE
5417 023714 000136          .WORD L10066-.
5418                               :.:.:.:.:
5419                               : JUMP TO END OF SUB IF ERROR
5420                               :.:.:.:.:

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 22 *****

```

5421 023716 012737 000132 002410      MOV    #132,PERR
5422 023724 004537 003016              JSR    R5,WFPE          ;WAIT FOR PROCEDURE ERROR
5423
5424 023730 005037 002402      CLR    ERRWRD
5425 023734 104410      TRAP   C$ESCAPE
5426 023736 000222      .WORD  L10064-.
5427
5428              .: ESCAPE TEST IF ERROR
5429              .:
5430
5431 023740 042777 000200 156476 GSSREP: BIC    #RDO,@BSEL2      ;CLEAR OUTPUT
5432 023746 005337 002412      DEC    TSSADD          ;GET TSSADD BACK TO MAX
5433 023752 013703 002412      MOV    TSSADD,R3
5434 023756 052703 000040      BIS    #BIT5,R3        ;SET READ BIT
5435 023762 004537 003162      JSR    R5,CONTIN      ;READ TSS
5436
5437 023766 005037 002402      CLR    ERRWRD
5438 023772 104410      TRAP   C$ESCAPE
5439 023774 000056      .WORD  L10066-.
5440
5441              .: JUMP TO END OF SUB IF ERROR
5442              .:
5443
5444 023776 012737 000002 002336      MOV    #2,$GDDAT
5445 024004 004537 003242      JSR    R5,GETOUT      ;CHECK FOR INFOR. OUT AND
5446
5447 024010 005037 002402      CLR    ERRWRD
5448
5449 024014 104410      TRAP   C$ESCAPE
5450 024016 000034      .WORD  L10066-.
5451
5452              .: JUMP TO END OF SUB IF ERROR
5453              .:
5454 024020 013737 002412 002336 60$: MOV    TSSADD,$GDDAT
5455 024026 004537 003446      JSR    R5,GETDAT      ;CHECK FOR GOOD DATA
5456
5457 024032 005037 002402      CLR    ERRWRD
5458 024036 104410      TRAP   C$ESCAPE
5459 024040 000012      .WORD  L10066-.
5460
5461              .: JUMP TO END OF SUB IF ERROR
5462              .:
5463
5464 024042 022737 000034 002412 70$: CMP    #34,TSSADD      ;ARE WE ALL DONE
5465 024050 001333      BNE    GSSREP          ;GO BACK IF NOT
5466
5467 024052      L10066:
5468 024052 104403      TRAP   C$ESUB
5469 024054      T22.3:
5470 024054 104402      TRAP   C$BSUB
5471
5472              .: READ CLEAR SLOT TEST :
5473              .:
5474
5475 024056 004737 004522      JSR    PC,MINITS      ;MASTER CLEAR MODE DEF
5476

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 22 *****

```

5477 024062 005037 002402      CLR      ERRWRD
5478 024066 104410      TRAP     C$ESCAPE
5479 024070 000066      .WORD   L10067-.
5480                               :
5481                               :JUMP TO END OF SUB IF ERROR
5482                               :
5483
5484 024072 005037 002360      CLR      TRIBN
5485 024076 012703 000117      MOV     #117,R3
5486 024102 004537 003162      JSR     R5,CONTIN      ;READ CLEAR SLOT
5487
5488 024106 005037 002402      CLR      ERRWRD
5489 024112 104410      TRAP     C$ESCAPE
5490 024114 000042      .WORD   L10067-.
5491                               :
5492                               :JUMP TO END OF SUB IF ERROR
5493                               :
5494
5495 024116 012737 000002 002336      MOV     #02,$GDDAT
5496 024124 004537 003242      JSR     R5,GETOUT      ;CHECK FOR INFO OUT
5497                               ;AND CORRECT TRIBN.
5498 024130 005037 002402      CLR      ERRWRD
5499 024134 104410      TRAP     C$ESCAPE
5500 024136 000020      .WORD   L10067-.
5501                               :
5502                               :JUMP TO END OF SUB IF ERROR
5503                               :
5504
5505 024140 012737 000117 002336      MOV     #117,$GDDAT
5506 024146 004537 003372      JSR     R5,GETRKY      ;CHECK FOR CORRECT RETURN KEY
5507
5508 024152 005037 002402      CLR      ERRWRD
5509 024156                               L10067:
5510 024156 104403      TRAP     C$ESUB
5511 024160                               L10064:
5512 024160 104401      TRAP     C$ETST
5513

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 22 *****

5514
5515
5516
5517 024162
5518
5519
5520
5521
5522
5523
5524
5525 024162
5526
5527 024162 004737 004522
5528 024166 005037 002402
5529 024172 104410
5530 024174 000412
5531 024176 012737 000036 002360
5532 024204 012703 000001
5533 024210 004537 003162
5534
5535
5536
5537
5538 024214 005037 002402
5539 024220 104410
5540 024222 000364
5541 024224 012703 000003
5542 024230 004537 003162
5543
5544
5545
5546 024234 005037 002402
5547 024240 104410
5548 024242 000344
5549 024244 012737 000001 002336
5550 024252 004537 003242
5551
5552
5553
5554 024256 005037 002402
5555 024262 104410
5556 024264 000322
5557 024266 012737 000024 002336
5558 024274 004537 003670
5559 024300 005037 002402
5560 024304 104410
5561 024306 000300
5562
5563
5564
5565 024310 042777 000200 156126 20\$:
5566 024316 052777 000200 156114
5567
5568 024324 004537 002732
5569

```

.SBTTL :***** TEST 23 *****
.SBTTL *HALT TRIB COMMAND TEST
ZZ
:*
:*
:* THIS TEST CHECKS THE HALT TRIB COMMAND
:* AND THEN CHECKS THAN A 2ND HALT TRIB
:* DOES NOT CAUSE A CONTROL OUT.
:*
.SBTTL :***** TEST 23 *****
T23::
JSR PC,MINITS ;MASTER CLEAR -MODE DEF
CLR ERRWRD ;IF ERROR GO TO END TEST
TRAP C$ESCAPE
.WORD L10070-.
MOV #36,TRIBN ;SET TRIBN
MOV #01,R3
JSR R5,CONTIN ;ESTABLISH TRIB

;IF TIME OUT OR READY ERRORS THE PROGRAM WILL
;REPORT THIS PC AS FAILING PC

CLR ERRWRD ; JUMP TO END OF TEST IF ERROR
TRAP C$ESCAPE
.WORD L10070-.
MOV #03,R3
JSR R5,CONTIN ;ISTRIB TRIB

;IF TIME OUT OR READY ERRORS THE PROGRAM WILL
;REPORT THIS PC AS FAILING PC

CLR ERRWRD ; JUMP TO END OF TEST IF ERROR
TRAP C$ESCAPE
.WORD L10070-.
MOV #01,$GDDAT ;CHECK
JSR R5,GETOUT ;FOR CONTROL OUT AND
;CORRECT TRIBN
;IF ERROR REPORT
;THIS PC AND ESCAPE TEST
; JUMP TO END OF TEST IF ERROR

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10070-.

; QUE REC BUFFER WITH 100 DECIMAL LOCATIONS

BIC #RDO,@BSEL2 ;CLEAR READY OUT
BIS #RQI,@BSELO ;SET REQUEST IN
;WAIT FOR READY IN
JSR R5,WRDI ;WAIT FOR RDI TO SET

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 23 *****

```

5570 024330 005037 002402      CLR      ERRWRD      ;CLEAR ERROR
5571 024334 104410              TRAP     C$ESCAPE
5572 024336 000250              .WORD   L10070-
5573 024340 042777 000200 156072  BIC     #RQI,@BSEL0  ;CLEAR REQUEST IN
5574 024346 012777 033734 156074  MOV     #RECBU1,@BSEL4 ;
5575 024354 012777 000144 156072  MOV     #100.,@BSEL6
5576 024362 113777 002360 156056  MOVB   TRIBN,@BSEL3
5577 024370 105077 156050      CLRB    @BSEL2      ;QUE UP BUFF
5578 024374 012703 000005      MOV     #05,R3
5579 024400 004537 003162      JSR     R5,CONTIN   ;HALT TRIB
5580                                ;IF TIME OUT OR READY ERRORS THE PROGRAM WILL
5581                                ;REPORT THIS PC AS FAILING PC
5582
5583 024404 005037 002402      CLR      ERRWRD      ; JUMP TO END OF TEST IF ERROR
5584 024410 104410              TRAP     C$ESCAPE
5585 024412 000174              .WORD   L10070-
5586 024414 012737 000003 002336  MOV     #03,$GDDAT
5587 024422 004537 003242      JSR     R5,GETOUT   ;CHECK FOR BUFFER UNUSED
5588                                ;AND CORRECT TRIBN.
5589 024426 005037 002402      CLR      ERRWRD      ; JUMP TO END OF TEST IF ERROR
5590 024432 104410              TRAP     C$ESCAPE
5591 024434 000152              .WORD   L10070-
5592 024436 012737 000144 002336  MOV     #100.,$GDDAT
5593 024444 004537 003514      JSR     R5,GETCC    ;CHECK FOR GOOD CHAR.COUNT
5594 024450 005037 002402      CLR      ERRWRD      ; JUMP TO END OF TEST IF ERROR
5595 024454 104410              TRAP     C$ESCAPE
5596 024456 000130              .WORD   L10070-
5597 024460 012737 033734 002336  MOV     #RECBU1,$GDDAT
5598 024466 004537 003576      JSR     R5,GETBA    ;CHECK FOR GOOD BUFF. ADD.
5599 024472 005037 002402      CLR      ERRWRD      ; JUMP TO END OF TEST IF ERROR
5600 024476 104410              TRAP     C$ESCAPE
5601 024500 000106              .WORD   L10070-
5602 024502 042777 000200 155734 23$:  BIC     #RDO,@BSEL2  ;CLEAR OUTPUT AND LOOK FOR NEXT
5603 024510 012737 000002 002336  MOV     #2,$GDDAT
5604 024516 004537 003242      JSR     R5,GETOUT   ;NEXT GET INFO OUT
5605 024522 005037 002402      CLR      ERRWRD      ; JUMP TO END OF TEST IF ERROR
5606 024526 104410              TRAP     C$ESCAPE
5607 024530 000056              .WORD   L10070-
5608 024532 012737 000020 002336  MOV     #20,$GDDAT
5609 024540 004537 003372      JSR     R5,GETRKY   ;CHECK FOR GOOD RETURN KEY
5610                                ; BUFFER RETURN COMPLETE
5611 024544 005037 002402      CLR      ERRWRD      ; JUMP TO END OF TEST IF ERROR
5612 024550 104410              TRAP     C$ESCAPE
5613 024552 000034              .WORD   L10070-
5614 024554 042777 000200 155662 25$:  BIC     #RDO,@BSEL2  ;CLEAR OUTPUT
5615 024562 004737 004244      JSR     PC,WAIT50   ;WAIT A WHILE
5616 024566 004737 004244      JSR     PC,WAIT50   ;DO A NO REQUEST
5617 024572 012703 000000      MOV     #0,R3
5618 024576 004537 003162      JSR     R5,CONTIN
5619 024602 005037 002402      CLR      ERRWRD
5620                                ;IF ERROR OCCURS THE SECOND
5621                                ;HALT TRIB CAUSED AN OUTPUT
5622                                ;AND SHOULD NOT HAVE.
5622 024606                    L10070:
5623 024606 104401              TRAP     C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 23 *****

5624
5625
5626
5627 024610
5628
5629
5630
5631
5632 024610
5633
5634 024610 004737 004522
5635
5636 024614 005037 002402
5637 024620 104410
5638 024622 000412
5639
5640
5641
5642
5643 024624 012737 000143 002360
5644 024632 012703 000001
5645 024636 004537 003162
5646
5647
5648
5649
5650 024642 005037 002402
5651 024646 104410
5652 024650 000364
5653
5654
5655
5656
5657 024652 012703 000041
5658 024656 004537 003162
5659
5660
5661
5662
5663 024662 005037 002402
5664 024666 104410
5665 024670 000344
5666
5667
5668
5669
5670 024672 012737 000002 002336
5671 024700 004537 003242
5672
5673
5674 024704 005037 002402
5675 024710 104410
5676 024712 000322
5677
5678
5679

```

.SBTTL :***** TEST 24 *****
.SBTTL *KILL TRIB TESTS
ZZ
:*
:* THIS TEST CHECKS THE KILL TRIB FUNCTIONS
:*-
.SBTTL :***** TEST 24 *****
T24::
JSR PC,MINITS

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10071-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #143,TRIBN ;SET TRIB NUMBER
MOV #01,R3
JSR R5,CONTIN ;ESTABLISH TRIB
:
: TIME OUT OR READY ERRORS REPORT THIS P
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10071-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #41,R3
JSR R5,CONTIN ;READ TRIB STATUS SLOT 1
:
: TIME OUT OR READY ERRORS REPORT THIS PC
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10071-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #02,$GDDAT ;SET TYPE FOR INFO OUT
JSR R5,GETOUT ;CHECK FOR INFO OUT AND
;CORRECT TRIBN.

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10071-.
:
: JUMP TO END OF TEST IF ERROR
:

```


CZDMTC.P11 25-MAR-81 08:24

:***** TEST 24 *****

```

5680
5681 024714 000377 155530 20$: SWAB @BSEL4 ;SWAB BYTES
5682 024720 117737 155524 002340 MOVB @BSEL4,$BDDAT ;MOVE TRIB ADD TO BDDAT
5683 024726 113737 002360 002336 MOVB TRIBN,$GDDAT ;MOVE TRIB NUMBER TO GDDAT
5684 024734 123737 002336 002340 CMPB $GDDAT,$BDDAT ;COMPARE
5685 024742 001407 BEQ 30$ ;IF OK GO TO 30
5686 ;ELSE ERROR
5687 024744 012737 012014 002430 MOV #M30F,COEW
5688 024752 104455 TRAP C$ERDF
5689 024754 000052 .WORD 42
5690 024756 011506 .WORD EROIC
5691 024760 010324 .WORD ERR27
5692
5693 024762 042777 000200 155454 30$: BIC #RDO,@BSEL2 ;CLEAR OUTPUT
5694 024770 012703 000004 MOV #04,R3
5695 024774 004537 003162 JSR R5,CONTIN ;MAINT STATE TRIB
5696 ;TIME OUT OR READY ERRORS REPORT THIS PC
5697 ;*****
5698
5699
5700 025000 005037 002402 CLR ERRWRD
5701 025004 104410 TRAP C$ESCAPE
5702 025006 000226 .WORD L10071-.
5703 ;*****
5704 ; JUMP TO END OF TEST IF ERROR
5705 ;*****
5706
5707 025010 012703 000002 MOV #02,R3
5708 025014 004537 003162 JSR R5,CONTIN ;KILL TRIB
5709 ;TIME OUT OR READY ERRORS REPORT THIS PC
5710 ;*****
5711
5712
5713 025020 005037 002402 CLR ERRWRD
5714 025024 104410 TRAP C$ESCAPE
5715 025026 000206 .WORD L10071-.
5716 ;*****
5717 ; JUMP TO END OF TEST IF ERROR
5718 ;*****
5719
5720 025030 012737 000112 002410 MOV #112,PERR ;CHECK FOR KILL TO UNHALTED
5721 025036 004537 003016 JSR R5,WFE ;WAIT FOR PROCEDURE ERROR
5722
5723 025042 005037 002402 CLR ERRWRD
5724 025046 104410 TRAP C$ESCAPE
5725 025050 000164 .WORD L10071-.
5726 ;*****
5727 ; ESCAPE TEST IF ERROR
5728 ;*****
5729 025052 042777 000200 155364 BIC #RDO,@BSEL2 ;CLEAR OUTPUT
5730 025060 012703 000005 MOV #05,R3
5731 025064 004537 003162 JSR R5,CONTIN ;HALT TRIB
5732 ;TIME OUT OR READY ERRORS REPORT THIS PC
5733 ;*****
5734
5735

```

CZDMTC.P11 25-MAR-81 08.24

:***** TEST 24 *****

```

5736 025070 005037 002402      CLR      ERRWRD
5737 025074 104410              TRAP     C$ESCAPE
5738 025076 000136              .WORD   L10071-.
5739                               :
5740                               : JUMP TO END OF TEST IF ERROR
5741                               :
5742                               :
5743 025100 012737 000002 002336  MOV     #2,$GDDAT
5744 025106 004537 003242              JSR     R5,GETOUT      ;CHECK FOR INFO OUT
5745                               :AND CORRECT PC
5746                               :
5747 025112 005037 002402      CLR      ERRWRD
5748 025116 104410              TRAP     C$ESCAPE
5749 025120 000114              .WORD   L10071-.
5750                               :
5751                               : JUMP TO END OF TEST IF ERROR
5752                               :
5753                               :
5754 025122 012737 000020 002336  MOV     #20,$GDDAT
5755 025130 004537 003372              JSR     R5,GETRKY     ;CHECK FOR GOOD RETURN KEY
5756                               :
5757                               :
5758 025134 005037 002402      CLR      ERRWRD
5759 025140 104410              TRAP     C$ESCAPE
5760 025142 000072              .WORD   L10071-.
5761                               :
5762                               : JUMP TO END OF TEST IF ERROR
5763                               :
5764                               :
5765 025144 042777 000200 155272 2$: BIC     #RDO,@BSEL2    ;CLEAR OUTPUT
5766 025152 012703 000002              MOV     #02,R3        ;KILL TRIB
5767 025156 004537 003162              JSR     R5,CONTIN
5768                               :
5769                               : TIME OUT OR READY ERRORS REPORT THIS PC
5770                               :
5771                               :
5772 025162 005037 002402      CLR      ERRWRD
5773 025166 104410              TRAP     C$ESCAPE
5774 025170 000044              .WORD   L10071-.
5775                               :
5776                               : JUMP TO END OF TEST IF ERROR
5777                               :
5778                               :
5779 025172 012703 000041              MOV     #41,R3        ;READ SLOT 1
5780 025176 004537 003162              JSR     R5,CONTIN
5781                               :
5782                               : TIME OUT OR READY ERRORS REPORT THIS PC
5783                               :
5784                               :
5785 025202 005037 002402      CLR      ERRWRD
5786 025206 104410              TRAP     C$ESCAPE
5787 025210 000024              .WORD   L10071-.
5788                               :
5789                               : JUMP TO END OF TEST IF ERROR
5790                               :
5791                               :

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 24 *****

5792 025212 012737 000106 002410
 5793 025220 004537 003016
~~5794~~
 5795 025224 005037 002402
 5796 025230 104410
 5797 025232 000002
 5798
 5799
 5800
 5801
 5802 025234
 5803 025234 104401
 5804
 5805

MOV #106,PERR ;CHECK FOR PROCEDURE 106 ERROR
 JSR R5,WFPE ;WAIT FOR PROCEDURE ERROR

 CLR ERRWRD
 TRAP C\$ESCAPE
 .WORD L10071-
 :::::
 : ESCAPE TEST IF ERROR
 :::::

 L10071:
 TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

..... TEST 24

5806
 5807
 5808
 5809 025236
 5810
 5811
 5812
 5813
 5814
 5815
 5816
 5817
 5818 025236
 5819 025236 012702 002414
 5820
 5821 025242 004737 004522
 5822
 5823 025246 005037 002402
 5824 025252 104410
 5825 025254 000074
 5826
 5827
 5828
 5829
 5830 025256 052777 000200 155154
 5831 025264 004537 002732
 5832
 5833 025270 005037 002402
 5834 025274 104410
 5835 025276 000052
 5836 025300 042777 000200 155132
 5837 025306 112277 155132
 5838 025312 012737 000102 002410
 5839 025320 004537 003016
 5840
 5841 025324 005037 002402
 5842 025330 104410
 5843 025332 000016
 5844
 5845
 5846
 5847 025334 042777 000200 155102
 5848 025342 022702 002420
 5849 025346 001335
 5850
 5851
 5852 025350
 5853 025350 104401
 5854

```

.SBTTL ..... TEST 25 .....
.SBTTL *CHECK FOR PROCEDURE ERROR 102
ZZ
:
:
: THIS TEST CHECKS THAT ILLEGAL TYPE CODES
: ON INPUT COMMANDS WILL PRODUCE PROCEDURE
: ERRORS.
:
:
.SBTTL ..... TEST 25 .....
T25::
MOV #TYLST,R2 ;SET R2 TO START OF LIST
108: JSR PC,MINITS ;MASTER CLEAR-MODE DEF

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10072-
:
: JUMP TO END OF TEST IF ERROR
:
:
BIS #RQI, @BSEL0 ;SET REQUEST
JSR R5,WRDI ;WAIT FOR RDI TO SET

CLR ERRWRD ;CLEAR ERROR
TRAP C$ESCAPE
.WORD L10072-
BIC #RQI, @BSEL0 ;CLEAR REQUEST
MOV #R2+, @BSEL2 ;DO FIRST BAD CODE
MOV #102, PERR
JSR R5,WFPE ;WAIT FOR PROCEDURE ERROR

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10072-
:
: ESCAPE TEST IF ERROR
:
BIC #RDO, @BSEL2 ;CLEAR READY OUT
CMP #TYEND, R2 ;IS IT END
BNE 108 ;IF NOT GO BACK

L10072:
TRAP C$ETST
    
```


CZDMTC.P11 25-MAR-81 08:24

;***** TEST 26 *****

5900
5901
5902
5903
5904 025436
5905
5906
5907
5908
5909
5910
5911
5912
5913 025436
5914
5915 025436 004737 004522
5916
5917 025442 005037 002402
5918 025446 104410
5919 025450 000212
5920
5921
5922
5923
5924 025452 012737 000003 002360
5925 025460 012703 000001
5926 025464 004537 003162
5927
5928
5929
5930
5931 025470 005037 002402
5932 025474 104410
5933 025476 000164
5934
5935
5936
5937
5938 025500 012703 000007
5939 025504 004537 003162
5940
5941 025510 005037 002402
5942 025514 104410
5943 025516 000144
5944
5945
5946
5947
5948 025520 012737 000120 002410
5949 025526 004537 003016
5950
5951 025532 005037 002402
5952 025536 104410
5953 025540 000122
5954
5955

```

.SBTTL ;***** TEST 27 *****
.SBTTL * CHECKS FOR PROCEDURE ERROR 120
ZZ
:*
:*
:* THIS TEST ISSUES A CONTROL IN WITH A REQUEST
:* KEY OF 7 AND LOOKS FOR A PROCEDURE ERROR OF
:* ILLEGAL REQUEST KEY ON CONTROL IN (120)
:*
:*-
.SBTTL ;***** TEST 27 *****
T27::
JSR PC,MINITS ;MASTER CLEAR - MODE DEF
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10074-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #3,TRIBN
MOV #01,R3
JSR R5,CONTIN ;ESTABLISH TRIB
:
: TIME OUT OR READY ERRORS REPORT HERE
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10074-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #07,R3
JSR R5,CONTIN ;DO CONTROL IN WITH KEY OF 07
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10074-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #120,PERR ;LOOK FOR ERROR
JSR R5,WFPE ;WAIT FOR PROCEDURE ERROR
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10074-.
:
: ESCAPE TEST IF ERROR

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 27 *****

```

5956
5957 025542 042777 000200 154674      BIC      #RDO,@BSEL2      ;CLEAR RDO
5958 025550 012703 000017              MOV      #17,R3          ;MOV 17 TO KEY WORD
5959 025554 004537 003162              JSR      R5,CONTIN       ;DO CONTROL IN WITH KEY OF 17
5960
5961 025560 005037 002402              CLR      ERRWRD
5962 025564 104410              TRAP    C$ESCAPE
5963 025566 000074              .WORD   L10074-.
5964
5965              ; JUMP TO END OF TEST IF ERROR
5966
5967
5968 025570 012737 000120 002410      MOV      #120,PERR       ;LOOK FOR ERROR
5969 025576 004537 003016              JSR      R5,WFPE         ;WAIT FOR PROCEDURE ERROR
5970
5971 025602 005037 002402              CLR      ERRWRD
5972 025606 104410              TRAP    C$ESCAPE
5973 025610 000052              .WORD   L10074-.
5974
5975              ; ESCAPE TEST IF ERROR
5976
5977 025612 042777 000200 154624      BIC      #RDO,@BSEL2      ;CLEAR OUTPUT
5978 025620 005003              CLR      R3
5979 025622 004537 003162              JSR      R5,CONTIN       ;DO A NO REQUEST
5980
5981 025626 005037 002402              CLR      ERRWRD
5982 025632 104410              TRAP    C$ESCAPE
5983 025634 000026              .WORD   L10074-.
5984
5985              ; JUMP TO END OF TEST IF ERROR
5986
5987
5988 025636 004737 004244              JSR      PC,WAIT50       ;THEN DELAY
5989 025642 032777 000200 154574      BIT      #RDO,@BSEL2
5990 025650 001404              BEQ     10$              ;IF NOT SET THEN END
5991
5992 025652 104455              TRAP    C$ERDF
5993 025654 000053              .WORD   43
5994 025656 012516              .WORD   MEF30
5995 025660 010422              .WORD   ERR32
5996 025662
5997 025662
5998 025662 104401              TRAP    C$ETST
5999

```

10\$:
L10074:

CZDMTC.P11 25-MAR-81 08:24

***** TEST 28 *****

6000
6001
6002 025664
6003
6004
6005
6006
6007
6008
6009
6010 025664
6011
6012 025664 004737 004522
6013
6014 025670 005037 002402
6015 025674 104410
6016 025676 000212
6017
6018
6019
6020
6021 025700 012737 000005 002360
6022 025706 012703 000001
6023 025712 004537 003162
6024
6025
6026
6027
6028 025716 005037 002402
6029 025722 104410
6030 025724 000164
6031
6032
6033
6034
6035 025726 012703 100000
6036 025732 004537 003162
6037
6038 025736 005037 002402
6039 025742 104410
6040 025744 000144
6041
6042
6043
6044
6045 025746 012737 000134 002410
6046 025754 004537 003016
6047
6048 025760 005037 002402
6049 025764 104410
6050 025766 000122
6051
6052
6053
6054 025770 042777 000200 154446
6055 025776 012703 046000

```

.SBTTL ***** TEST 28 *****
.SBTTL * CHECK FOR PROCEDURE ERROR 134
ZZ
:
:
:
: THIS TEST CHECKS FOR PROCEDURE ERROR OF USING
: RESERVED BITS IN BSEL7 ON CONTROL IN
:
:
.SBTTL ***** TEST 28 *****
T28::
JSR PC,MINITS ;MASTER CLEAR MODE DEF
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10075-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #5,TRIBN
MOV #1,R3
JSR R5,CONTIN ;ESTABLISH TRIB
:
: TIME OUT OR READY ERRORS REPORT THIS PC
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10075-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #100000,R3 ;SET BIT 7
JSR R5,CONTIN
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10075-.
:
: JUMP TO END OF TEST IF ERROR
:
MOV #134,PERR
JSR R5,WFPE ;WAIT FOR PROCEDURE ERROR
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10075-.
:
: ESCAPE TEST IF ERROR
:
BIC #RDO,@BSEL2 ;CLEAR OUTPUT
MOV #46000,R3 ;SET 6 3 AND2

```


CZDMTC.P11 25-MAR-81 08:24

***** TEST 28 *****

```

6056 026002 004537 003162      JSR    R5,CONTIN      ;DO CONTROL IN
6057
6058 026006 005037 002402      CLR    ERRWRD
6059 026012 104410      TRAP   C$ESCAPE
6060 026014 000074      .WORD  L10075-.
6061      ;:
6062      ;: JUMP TO END OF TEST IF ERROR
6063      ;:
6064
6065 026016 012737 000134 002410      MOV    #134,PERR
6066 026024 004537 003016      JSR    R5,WFPE        ;WAIT FOR PROCEDURE ERROR
6067
6068 026030 005037 002402      CLR    ERRWRD
6069 026034 104410      TRAP   C$ESCAPE
6070 026036 000052      .WORD  L10075-.
6071      ;:
6072      ;: ESCAPE TEST IF ERROR
6073      ;:
6074 026040 005003      CLR    R3
6075 026042 042777 000200 154374      BIC    #RDO,@BSEL2    ;CLEAR OUTPUT
6076 026050 004537 003162      JSR    R5,CONTIN      ;DO CONTROL IN
6077
6078 026054 005037 002402      CLR    ERRWRD
6079 026060 104410      TRAP   C$ESCAPE
6080 026062 000026      .WORD  L10075-.
6081      ;:
6082      ;: JUMP TO END OF TEST IF ERROR
6083      ;:
6084
6085 026064 004737 004244      JSR    PC,WAIT50      ;WAIT A WHILE
6086 026070 032777 000200 154346      BIT    #RDO,@BSEL2    ;IS RDO SET NOW
6087 026076 001404      BEQ    10$            ;IF NOT THEN GO ON
6088      ;ELSE ERROR
6089 026100 104455      TRAP   C$ERDF
6090 026102 000054      .WORD  44
6091 026104 012516      .WORD  MEF30
6092 026106 010422      .WORD  ERR32
6093 026110
6094 026110
6095 026110 104401      10$:
L10075: TRAP   C$ETST
6096

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 29 *****

6097
6098
6099 026112
6100
6101
6102
6103
6104
6105
6106
6107
6108
6109 026112
6110
6111 026112 004737 004522
6112
6113 026116 005037 002402
6114 026122 104410
6115 026124 000312
6116
6117
6118
6119
6120 026126 012737 000027 002360
6121 026134 012703 000001
6122 026140 004537 003162
6123
6124
6125
6126
6127 026144 005037 002402
6128 026150 104410
6129 026152 000264
6130
6131
6132
6133
6134 026154 012704 000003
6135 026160 012703 020000
6136 026164 004537 003162
6137
6138
6139
6140
6141 026170 005037 002402
6142 026174 104410
6143 026176 000240
6144
6145
6146
6147
6148 026200 012703 000042
6149 026204 004537 003162
6150
6151
6152

```

.SBTTL ;***** TEST 29 *****
.SBTTL *LATCH - UNLATCH POLL CHECK
ZZ
;*
;*
;* THIS TEST CHECKS THE LATCH - UNLATCH POLL
;* COMMANDS. FIRST LATCH TRIB IN DEAD STATE
;* MAKE SURE ITS DEAD. THEN UNLATCH AND MAKE
;* SURE THAT IT GOES ACTIVE.
;*
.SBTTL ;***** TEST 29 *****
T29::
JSR PC,MINITS ;MASTER CLEAR MODE DEF
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10076-.
; JUMP TO END OF TEST IF ERROR
; JUMP TO END OF TEST IF ERROR
MOV #27,TRIBN
MOV #01,R3 ;ESTABLISH TRIB
JSR R5,CONTIN
; TIME OUT OR READY ERROR REPORT THIS PC
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10076-.
; JUMP TO END OF TEST IF ERROR
MOV #3,R4
MOV #20000,R3 ;LATCH POLL DEAD
JSR R5,CONTIN
; TIME OUT OR READY ERRORS REPORT THIS PC
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10076-.
; JUMP TO END OF TEST IF ERROR
MOV #42,R3
JSR R5,CONTIN ;READ SLOT 2 TSS
; TIME OUT OR READY ERRORS REPORT THIS PC

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 29 *****

6153					
6154	026210	005037	002402		CLR ERRWRD
6155	026214	104410			TRAP C\$ESCAPE
6156	026216	000220			.WORD L10076-
6157				
6158					: JUMP TO END OF TEST IF ERROR
6159				
6160					
6161	026220	012737	000002	002336	MOV #02,\$GDDAT
6162	026226	004537	003242		JSR R5,GETOUT
6163				
6164					: CHECK FOR INFO OUT AND CORRECT TRIBN
6165					: IF ERROR REPORT THIS PC
6166				
6167					
6168	026232	005037	002402		CLR ERRWRD
6169	026236	104410			TRAP C\$ESCAPE
6170	026240	000176			.WORD L10076-
6171				
6172					: JUMP TO END OF TEST IF ERROR
6173				
6174					
6175	026242	012737	000042	002336	MOV #42,\$GDDAT
6176	026250	004537	003372		JSR R5,GETRKY
6177				
6178					: CHECK FOR CORRECT RETURN KEY
6179				
6180					
6181	026254	005037	002402		CLR ERRWRD
6182	026260	104410			TRAP C\$ESCAPE
6183	026262	000154			.WORD L10076-
6184				
6185					: JUMP TO END OF TEST IF ERROR
6186				
6187					
6188	026264	012737	100220	002336	MOV #100220,\$GDDAT ;
6189	026272	004537	003446		JSR R5,GETDAT
6190				
6191					: CHECK FOR DEAD STATE
6192				
6193					
6194	026276	005037	002402		CLR ERRWRD
6195	026302	104410			TRAP C\$ESCAPE
6196	026304	000132			.WORD L10076-
6197				
6198					: JUMP TO END OF TEST IF ERROR
6199				
6200					
6201	026306	042777	000200	154130	BIC #RDO,@BSEL2 ;CLEAR OUTPUT
6202	026314	012703	010000		MOV #10000,R3
6203	026320	004537	003162		JSR R5,CONTIN ;UNLATCH POLL
6204				
6205					: TIME OUT OR READY ERRORS REPOFT THIS PC
6206				
6207					
6208	026324	005037	002402		CLR ERRWRD

CZDMTC.P11 25-MAR-81 08:24

***** TEST 29 *****

6209 026330 104410
6210 026332 000104
6211
6212
6213
6214
6215 026334 012703 000042
6216 026340 004537 003162
6217
6218
6219
6220
6221 026344 005037 002402
6222 026350 104410
6223 026352 000064
6224
6225
6226
6227
6228 026354 012737 000002 002336
6229 026362 004537 003242
6230
6231
6232
6233
6234 026366 005037 002402
6235 026372 104410
6236 026374 000042
6237
6238
6239
6240
6241 026376 012737 000042 002336
6242 026404 004537 003372
6243
6244
6245
6246
6247 026410 005037 002402
6248 026414 104410
6249 026416 000020
6250
6251
6252
6253
6254 026420 012737 000600 002336
6255 026426 004537 003446
6256
6257
6258
6259 026432 005037 002402
6260 026436
6261 026436 104401
6262

```

TRAP  C$ESCAPE
.WORD L10076-.
:.....:
: JUMP TO END OF TEST IF ERROR
:.....:

MOV   #42,R3
JSR   R5,CONTIN      ;READ TSS SLOT 2
:.....:
:TIME OUT OR READY ERRORS REPORT THIS PC
:.....:

CLR   ERRWRD
TRAP  C$ESCAPE
.WORD L10076-.
:.....:
: JUMP TO END OF TEST IF ERROR
:.....:

MOV   #02,$GDDAT      ;
JSR   R5,GETOUT
:.....:
:CHECK FOR INFORMATION OUT AND CORRECT TRIBN
:.....:

CLR   ERRWRD
TRAP  C$ESCAPE
.WORD L10076-.
:.....:
: JUMP TO END OF TEST IF ERROR
:.....:

MOV   #42,$GDDAT
JSR   R5,GETRKY
:.....:
:CHECK FOR CORRECT RETURN KEY
:.....:

CLR   ERRWRD
TRAP  C$ESCAPE
.WORD L10076-.
:.....:
: JUMP TO END OF TEST IF ERROR
:.....:

MOV   #600,$GDDAT
JSR   R5,GETDAT
:.....:
: CHECK FOR ACTIVE STATE
:.....:

CLR   ERRWRD
L10076: TRAP  C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 29 *****

6263
6264
6265
6266
6267 026440
6268
6269
6270
6271
6272
6273
6274
6275
6276
6277 026440
6278 026440 012737 001750 002424
6279 026446 012737 033734 002422
6280 026454 012737 033454 002420
6281 026462 012737 000004 002426
6282 026470 005037 002432
6283 026474 012737 000003 002360
6284 026502 004737 004522
6285
6286
6287 026506 005037 002402
6288 026512 104410
6289 026514 000016
6290
6291
6292
6293
6294 026516 004537 006346
6295 026522 005037 002402
6296 026526 104410
6297 026530 000002
6298 026532
6299 026532 104401
6300

.SBTTL ***** TEST 30 *****
.SBTTL SHORT MESSAGE SENDING TEST, WITH INTERNAL LOOPBACK

ZZ
*
* THIS TEST SENDS A 4 BYTE MESSAGE FROM AN EVEN TRANSMIT
* BUFFER TO AN EVEN REC BUFFER IN DDCMP FORMAT CONFIGURED
* AS A MULTIPOINT CONTROL STATION FULL DUPLEX. THE TEST
* CHECKS THAT REC BUFFERS ARE RETURNED AND THAT THE DATA
* IS CORRECT. IT ALSO CHECKS THAT THE NEXT OUTPUT COMMAND
* IS A TX BUFFER COMPLETE.

.SBTTL ***** TEST 30 *****

T30::
MOV #1000, RXCC ;SET UP RX CC
MOV #RECBU1, RXADD ;SET UP RX ADD
MOV #MR1, TXADD ;SET UP TX ADD
MOV #4, TXCC ;SET UP TX COUNT
CLR GENWRD ;CLEAR GEN WORD
MOV #3, TRIBN ;SET TRIB ADDRESS
JSR PC, MINITS ;INITIALIZE

CLR ERRWRD
TRAP C\$ESCAPE
.WORD L10077-.
: JUMP TO END OF TEST IF ERROR

JSR R5, TXRX3 ;TRANSMIT AND REC.
CLR ERRWRD
TRAP C\$ESCAPE
.WORD L10077-.
L10077: TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 31 *****

6301
6302
6303 026534
6304
6305
6306
6307
6308
6309
6310 026534
6311
6312 026534 004737 004522
6313
6314 026540 005037 002402
6315 026544 104410
6316 026546 000104
6317
6318
6319
6320
6321 026550 052777 000200 153662
6322 026556 004537 002732
6323
6324 026562 005037 002402
6325 026566 104410
6326 026570 000062
6327 026572 042777 000200 153640
6328 026600 112777 000003 153640
6329 026606 012777 000010 153640
6330 026614 012777 033734 153626
6331 026622 112777 000000 153614
6332 026630 012737 000122 002410
6333 026636 004537 003016
6334
6335 026642 005037 002402
6336 026646 104410
6337 026650 000002
6338
6339
6340
6341 026652
6342 026652 104401
6343

```

.SBTTL ***** TEST 31 *****
.SBTTL * PROCEDURE ERROR 122 CHECK
      ZZ
      *
      *
      * THIS TEST CHECKS FOR PROCEDURE ERROR 122
      * ESTABLISH BUFFER FOR UNESTABLISHED TRIB.
      *
.SBTTL ***** TEST 31 *****
T31::
      JSR      PC,MINITS          ;MASTER CLEAR MODE DEF
      CLR      ERRWRD
      TRAP     C$ESCAPE
      .WORD    L10100-
      :::::::
      : JUMP TO END OF TEST IF ERROR
      :::::::
      BIS      #RQI,@BSELO        ;SET REQUEST FOR INPUT
      JSR      R5,WRDI            ;WAIT FOR RDI TO SET
      CLR      ERRWRD            ;CLEAR ERROR
      TRAP     C$ESCAPE
      .WORD    L10100-
      BIC      #RQI,@BSELO        ;CLEAR REQUEST
      MOVB     #03,@BSEL3
      MOV      #10,@BSEL6         ;SET CC
      MOV      #RECBU1,@BSEL4    ;SET BA
      MOVB     #0,@BSEL2         ;ESTABLISH BUFFER
      MOV      #122,PERR         ;WAIT
      JSR      R5,WFPE           ;WAIT FOR PROCEDURE ERROR

      CLR      ERRWRD
      TRAP     C$ESCAPE
      .WORD    L10100-
      :::::::
      : ESCAPE TEST IF ERROR
      :::::::
L10100:
      TRAP     C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 32 *****

6344
6345
6346 026654
6347
6348
6349
6350
6351
6352
6353 026654
6354
6355 026654 004737 004522
6356
6357 026660 005037 002402
6358 026664 104410
6359 026666 000132
6360
6361
6362
6363
6364 026670 012737 000047 002360
6365 026676 012703 000001
6366 026702 004537 003162
6367
6368
6369
6370
6371 026706 005037 002402
6372 026712 104410
6373 026714 000104
6374
6375
6376
6377
6378 026716 052777 000200 153514
6379 026724 004537 002732
6380
6381 026730 005037 002402
6382 026734 104410
6383 026736 000062
6384 026740 042777 000200 153472
6385 026746 012777 000010 153500
6386 026754 012777 033734 153466
6387 026762 113777 002360 153456
6388 026770 112777 000000 153446
6389 026776 012737 000124 002410
6390 027004 004537 003016
6391
6392 027010 005037 002402
6393 027014 104410
6394 027016 000002
6395
6396
6397
6398 027020
6399 027020 104401

```

.SBTTL :***** TEST 32 *****
.SBTTL *PROCEDURE ERROR 124 CHECK
      ZZ
      *
      *
      * THIS TEST CHECKS FOR PROCEDURE ERROR 124
      * ESTABLISHING BUFFER FOR HALTED TRIB
      *
.SBTTL :***** TEST 32 *****
T32::
      JSR    PC,MINITS      ;MASTER CLEAR MODE DEF
      CLR    ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10101-.
      :
      : JUMP TO END OF TEST IF ERROR
      :
      MOV    #47,TRIBN
      MOV    #01,R3
      JSR    R5,CONTIN      ;ESTABLISH TRIB
      :
      : TIME OUT AND READY ERRORS REPORT THIS PC
      :
      CLR    ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10101-.
      :
      : JUMP TO END OF TEST IF ERROR
      :
      BIS    #RQI,@BSEL0    ;SET REQUEST
      JSR    R5,WRDI        ;WAIT FOR RDI TO SET
      CLR    ERRWRD        ;CLEAR ERROR
      TRAP   C$ESCAPE
      .WORD  L10101-.
      BIC    #RQI,@BSEL0    ;CLEAR REQUEST
      MOV    #10,@BSEL6     ;SET CC
      MOV    #RECBU1,@BSEL4 ;SET BA
      MOVB   TRIBN,@BSEL3   ;SET TRIB NO.
      MOVB   #0,@BSEL2      ;ESTABLISH BUFFER
      MOV    #124,PERR      ;WAIT FOR ERROR
      JSR    R5,WFPE        ;WAIT FOR PROCEDURE ERROR
      CLR    ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10101-.
      :
      : ESCAPE TEST IF ERROR
      :
L10101:
      TRAP   C$ETST

```

CZDMTC.P11

25-MAR-81 08:24

;***** TEST 32 *****

- 6400
- 6401
- 6402
- 6403

CZDMTC.P11 25-MAR-81 08:24

***** TEST 33 *****

6404
6405
6406 027022
6407
6408
6409
6410
6411
6412
6413
6414 027022
6415
6416 027022 004737 004522
6417
6418 027026 005037 002402
6419 027032 104410
6420 027034 000152
6421
6422
6423
6424
6425 027036 012737 000074 002360
6426 027044 012703 000001
6427 027050 004537 003162
6428
6429 027054 005037 002402
6430 027060 104410
6431 027062 000124
6432
6433
6434
6435
6436 027064 012703 000004
6437 027070 004537 003162
6438
6439 027074 005037 002402
6440 027100 104410
6441 027102 000104
6442
6443
6444
6445
6446 027104 052777 000200 153326
6447 027112 004537 002732
6448
6449 027116 005037 002402
6450 027122 104410
6451 027124 000062
6452 027126 042777 000200 153304
6453 027134 012777 000000 153312
6454 027142 012777 033734 153300
6455 027150 113777 002360 153270
6456 027156 112777 000000 153260
6457 027164 012737 000126 002410
6458 027172 004537 003016
6459

```

.SBTTL ***** TEST 33 *****
.SBTTL *PROCEDURE ERROR #126 CHECK
      ZZ
      *
      * THIS TEST CHECKS FOR A PROCEDURE ERROR OF #126
      * ASSIGNING A BUFFER WITH A ZERO BYTE COUNT
      *
.SBTTL ***** TEST 33 *****
133::
      JSR    PC,MINITS      ;MASTER CLEAR MODE DEF

      CLR    ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10102-.
      :
      : JUMP TO END OF TEST IF ERROR
      :
      MOV    #74,TRIBN
      MOV    #01,R3
      JSR    R5,CONTIN      ; ESTABLISH TRIB

      CLR    ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10102-.
      :
      : JUMP TO END OF TEST IF ERROR
      :
      MOV    #04,R3
      JSR    R5,CONTIN      ;MAINT STATE

      CLR    ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10102-.
      :
      : JUMP TO END OF TEST IF ERROR
      :
      BIS    #R0I,@BSELO    ;SET REQUEST
      JSR    R5,WRDI        ;WAIT FOR RDI TO SET

      CLR    ERRWRD        ;CLEAR ERROR
      TRAP   C$ESCAPE
      .WORD  L10102-.
      BIC    #R0I,@BSELO    ;CLEAR REQUEST
      MOV    #0,@BSEL6      ;0 BYTES
      MOV    #RECBU1,@BSEL4 ;BA
      MOVB   TRIBN,@BSEL3   ;SET TRIBN
      MOVB   #0,@BSEL2      ;SET BUFFER
      MOV    #126,PERR      ;WAIT FOR ERROR
      JSR    R5,WfPE        ;WAIT FOR PROCEDURE ERROR

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 33 *****

6460	027176	005037	002402
6461	027202	104410	
6462	027204	000002	
6463			
6464			
6465			
6466			
6467	027206		
6468	027206	104401	
6469			

```

CLR      ERRWRD
TRAP     C$ESCAPE
.WORD    L10102-
:
:
: ESCAPE TEST IF ERROR
:
:

```

```

L10102:
TRAP     C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 34 *****

6470
6471
6472 027210
6473
6474
6475
6476
6477
6478
6479
6480 027210
6481
6482 027210 004737 004522
6483
6484 027214 005037 002402
6485 027220 104410
6486 027222 000102
6487
6488
6489
6490
6491 027224 052777 000200 153206
6492
6493 027232 004537 002732
6494
6495 027236 005037 002402
6496 027242 104410
6497 027244 000060
6498 027246 042777 000200 153164
6499 027254 105077 153166
6500 027260 012777 000010 153166
6501 027266 012777 033454 153154
6502 027274 012777 000004 153142
6503
6504 027302 012737 000130 002410
6505 027310 004537 003016
6506
6507 027314 005037 002402
6508 027320 104410
6509 027322 000002
6510
6511
6512
6513 027324
6514 027324 104401
6515
6516
6517

```

.SBTTL ***** TEST 34 *****
.SBTTL *CHECK FOR PROCEDURE ERROR 130
ZZ
:*
:*
:* THIS TEST CHECKS FOR A PROCEDURE ERROR OF 130
:* ASSIGNING TRANSMIT BUFFER FOR TRIB 0
:*
:*-
.SBTTL ***** TEST 34 *****
T34::

JSR    PC,MINITS      ;MASTER CLEAR MODE DEF

CLR    ERRWRD
TRAP   C$ESCAPE
.WORD  L10103-.
:
: JUMP TO END OF TEST IF ERROR
:
BIS    #RQ1,@BSEL0    ;SET REQUEST
:WAIT FOR READY
JSR    R5,WRDI        ;WAIT FOR RDI TO SET

CLR    ERRWRD        ;CLEAR ERROR
TRAP   C$ESCAPE
.WORD  L10103-.
BIC    #RQ1,@BSEL0    ;CLEAR REQUEST.
CLRB   @BSEL3         ;MAKE TRIB NO. 0
MOV    #10,@BSEL6     ;MAKE CC 10
MOV    #MR1,@BSEL4    ;MAKE ADD MR1
MOV    #4,@BSEL2      ;CLEAR RDI AND ESTAB
:TRANSMIT BUFFER
MOV    #130,PERR      ;WAIT FOR PROCEDURE ERROR
JSR    R5,WFPE        ;WAIT FOR PROCEDURE ERROR

CLR    ERRWRD
TRAP   C$ESCAPE
.WORD  L10103-.
:
: ESCAPE TEST IF ERROR
:
L10103: TRAP   C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 34 *****

6518
6519
6520
6521 027326
6522
6523
6524
6525
6526
6527
6528
6529
6530
6531
6532 027326
6533 027326 012737 000035 002370
6534
6535 027334 012737 001750 002424
6536 027342 012737 033734 002422
6537 027350 012737 000400 002426
6538 027356 012737 033454 002420
6539 027364 005037 002432
6540 027370 012737 000001 002360
6541 027376 022737 000004 002476
6542 027404 001402
6543 027406 005237 002400
6544 027412 004737 004512
6545 027416 005037 002400
6546 027422 005037 002402
6547 027426 104410
6548 027430 000016
6549
6550
6551
6552
6553 027432 004537 005720
6554
6555 027436 005037 002402
6556 027442 104410
6557 027444 000002
6558
6559
6560
6561
6562 027446
6563 027446 104401
6564
6565

```

.SBTTL ***** TEST 35 *****
.SBTTL * TRANSMIT REC 256,PTP,DDCMP
      ZZ
      *
      *
      * THIS TEST WILL TRANSMIT 256 BYTE MESSAGE
      * DDCMP PROTOCOL
      * THIS WILL BE DONE EXTERNAL LOOPBACK IF
      * IT EXISTS ELSE INTERNAL LOOPBACK WILL
      * BE USED
      *
      *
.SBTTL ***** TEST 35 *****
T35::
      MOV     #35,ROMN1      ;SET UP TEST NUMBER
      MOV     #1000.,RXCC    ;SET REC BUFFER FOR 1000BYTES
      MOV     #RECBU1,RXADD  ; SET UP BUFF ADD
      MOV     #256.,TXCC    ;SET UP RX CHAR COUNT
      MOV     #MR1,TXADD    ;AND ADDRESS
      CLR     GENWRD        ;CLEAR GEN WORD
      MOV     #01,TRIBN     ;SET THE TRIB TO 01
      CMP     #4,TSTCON     ;CHECK FOR LOOPBACK CONNECTOR.
      BEQ     1$
      INC     EXLOOP        ;EXT CONNECTOR PRESENT: SET EXT LOOP FLAG
1$:    JSR     PC,MINIT1    ;MASTER CLEAR-MODE DEF
      CLR     EXLOOP        ;CLEAR EXTERNAL LOOP FLAG
      CLR     ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10104-.
      .:
      .: JUMP TO END OF TEST IF ERROR
      .:
      JSR     R5,TXRCSR     ;GO TRANSMIT RX AND CHECK
      CLR     ERRWRD
      TRAP   C$ESCAPE
      .WORD  L10104-.
      .:
      .: JUMP TO END OF TEST IF ERROR
      .:
L10104:
      TRAP   C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 35 *****

6566
6567
6568
6569 027450
6570
6571
6572
6573
6574
6575
6576
6577
6578
6579 027450
6580 027450 032737 000003 002472
6581 027456 001454
6582
6583 027460 012737 000036 002370
6584 027466 012737 001750 002424
6585 027474 012737 033734 002422
6586 027502 012737 000400 002426
6587 027510 012737 033454 002420
6588 027516 005037 002432
6589 027522 012737 000001 002360
6590 027530 022737 000004 002476
6591 027536 001402
6592 027540 005237 002400
6593 027544 004737 004512
6594 027550 005037 002400
6595 027554 005037 002402
6596 027560 104410
6597 027562 000026
6598
6599
6600
6601 027564 005237 002376
6602 027570 004537 005720
6603 027574 005037 002402
6604 027600 005037 002376
6605 027604 104410
6606 027606 000002
6607
6608
6609
6610 027610
6611 027610
6612 027610 104401
6613

```

.SBTTL ***** TEST 36 *****
.SBTTL * DMV Q22 MODE TX AND RX,256 BYTES,DDCMP
ZZ
:*
:*          **** DMV ONLY ****
:* THIS TEST WILL TRANSMIT A 256 BYTE MESSAGE DDCMP PROTOCOL
:* USING THE 'Q22' CSR MODE OF THE DMV-11
:* THIS WILL BE DONE EXTERNAL LOOPBACK IF IT EXISTS ( ELSE
:* INTERNAL LOOPBACK WILL BE USED).
:*
:*-
.SBTTL ***** TEST 36 *****
T36::
BIT      #3,OPTYP      ;IS THIS A DMV ?
BEQ      T36END        ;IF NOT: EXIT TEST

MOV      #36,ROMN1     ;SET UP TEST NUMBER
MOV      #1000.,RXCC   ;SET REC BUFFER FOR 1000BYTES
MOV      #RECBU1,RXADD ; SET UP BUFF ADD
MOV      #256.,TXCC   ;SET UP RX CHAR COUNT
MOV      #MR1,TXADD    ;AND ADDRESS
CLR      GENWRD        ;CLEAR GEN WORD
MOV      #01,TRIBN    ;SET THE TRIB TO 01
CMP      #4,TSTCON     ;CHECK FOR LOOPBACK CONNECTOR.
BEQ      1$            ;
INC      EXLOOP        ;EXT CONNECTOR PRESENT: SET EXT LOOP FLAG
1$:      JSR      PC,MINIT1 ;MASTER CLEAR-MODE DEF
CLR      EXLOOP        ;CLEAR NO TTLOOP FLAG
CLR      ERRWRD
TRAP    C$ESCAPE
.WORD    L10105-.

:.....:
: JUMP TO END OF TEST IF ERROR
:.....:
INC      MODQ22        ;SET Q22 MODE FLAG (FOR TXRXSR)
JSR      R5,TXRXSR    ;GO TRANSMIT RX AND CHECK
CLR      ERRWRD
CLR      MODQ22
TRAP    C$ESCAPE
.WORD    L10105-.

:.....:
: JUMP TO END OF TEST IF ERROR
:.....:
T36END:
L10105: TRAP    C$SETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 37 *****

6614
6615
6616 027612
6617
6618
6619
6620
6621
6622
6623
6624
6625
6626 027612
6627 027612 012737 000377 002426
6628 027620 012737 033454 002420
6629 027626 012737 000764 002424
6630 027634 012737 033735 002422
6631 027642 012737 000075 002360
6632 027650 005037 002432
6633 027654 004737 004522
6634
6635
6636 027660 005037 002402
6637 027664 104410
6638 027666 000016
6639
6640
6641
6642
6643 027670 004537 006346
6644
6645
6646 027674 005037 002402
6647 027700 104410
6648 027702 000002
6649
6650
6651
6652
6653 027704
6654 027704 104401
6655

```

.SBTTL ;***** TEST 37 *****
.SBTTL * TX AND RX,255 BYTES,EVEN TX,ODD RX,DDCMP,MULTIPOINT
ZZ
;*
;*
;* THIS TEST WILL TRANSMIT A MESSAGE OF 255
;* BYTES FROM AN EVEN TX START ADD TO AN ODD
;* REC START ADD. IN DDCMP MODE MULTI POINT
;* CONTROL STATION.
;*
;*-
.SBTTL ;***** TEST 37 *****
i37::
MOV #255.,TXCC ;SET UP TRANSMIT CHAR COUNT
MOV #MR1,TXADD ;SET UP TRANSMIT ADD
MOV #500.,RXCC ;SET UP REC CHAR COUNT
MOV #RECBU1+1,RXADD ;SET UP REC ADD
MOV #75,TRIBN ;SET UP TRIB NO.
CLR GENWRD ;CLEAR THE GENWRD
JSR PC,MINITS ;MASTER CLEAR MODE DEF
;MULTI POINT CONTROL

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10106-.
;*****
; JUMP TO END OF TEST IF ERROR
;*****

20$: JSR R5,TXRX3 ;GO TRANSMIT AND REC
; AND CHECK DATA.

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10106-.
;*****
; JUMP TO END OF TEST IF ERROR
;*****

L10106. TRAP C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 38 *****

6656
6657
6658 027706
6659
6660
6661
6662
6663
6664
6665
6666
6667 027706
6668
6669 027706 032737 000003 002472
6670 027714 001402
6671 027716 104432
6672 027720 000412
6673
6674 027722 005002
6675 027724 022737 000000 002476
6676 027732 001135
6677 027734 016237 030306 002506
6678 027742 004737 004530
6679 027746 004737 004244
6680
6681 027752 005037 002402
6682 027756 104410
6683 027760 000352
6684
6685
6686
6687 027762 022702 000010
6688 027766 101403
6689
6690 027770 142777 000010 152444
6691
6692 027776 005037 002360
6693 030002 016204 030232
6694 030006 012703 000023
6695 030012 004537 003162
6696
6697
6698
6699
6700 030016 005037 002402
6701 030022 104410
6702 030024 000306
6703
6704
6705
6706 030026 012704 000100
6707 030032 012703 000021
6708 030036 004537 003162
6709
6710
6711

```

.SBTTL ***** TEST 38 *****
.SBTTL *READ/WRITE MODEM TESTS
ZZ
:*
:*
:* THIS TEST WILL SELECT EACH OF THE 4 MODEM
:* INTERFACES AND WRITE AND READ THEM. THIS IS
:* ONLY DONE IF CONNECTORS ARE PRESENT
:* ( DMP ONLY ).
:*
.SBTTL ***** TEST 38 *****
T38::
BIT #3,OPTYP ;IS THIS A DMV11 ?
BEQ 1$ ; NO: GOOD, CONTINUE TEST
TRAP C$EXIT
.WORD L10107-.

1$: CLR R2 ;CLEAR R2
CMP #0,TSTCON ;IS THIS WITH TEST CONN LOOPBACK
BNE MODEX ;IF NOT GO TO END
MODEB: MOV DUPTYP(R2),AXNUM
JSR PC,MINTR ;MASTER CLEAR MODE DEF
JSR PC,WAIT50 ;DELAY

CLR ERRWRD
TRAP C$ESCAPE
.WORD L10107-.
: JUMP TO END OF TEST IF ERROR
: JUMP TO END OF TEST IF ERROR
CMP #10,R2
BLOS MODEA ;DON'T TURN OFF LINE UINT LOOP
;IF PAST THIS POINT IN TABLES
MODEA: BICB #BIT3,@BSEL1
CLR TRIBN ;MAKE TRIB NO. = 0
MOV MODTYP(R2),R4 ;SELECT TYPE OF INTERFACE
MOV #23,R3 ;SELECT INTERFACE
JSR R5,CONTIN
: TIME OUT OR READY ERRORS REPORT THIS PC
: JUMP TO END OF TEST IF ERROR
: JUMP TO END OF TEST IF ERROR
MODED: MOV #100,R4
MOV #21,R3 ;WRITE MODEM
JSR R5,CONTIN
: TIME OUT OR READY ERRORS REPORT THIS PC
: JUMP TO END OF TEST IF ERROR

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 38 *****

6712					
6713	030042	005037	002402		CLR ERRWRD
6714	030046	104410			TRAP C\$ESCAPE
6715	030050	000262			.WORD L10107-.
6716				
6717					: JUMP TO END OF TEST IF ERROR
6718				
6719					
6720	030052	012703	000020		MOV #20,R3
6721	030056	004537	003162		JSR R5,CONTIN ;READ MODEM
6722				
6723					: TIME OUT OR READY ERRORS REPORT THIS PC
6724				
6725					
6726	030062	005037	002402		CLR ERRWRD
6727	030066	104410			TRAP C\$ESCAPE
6728	030070	000242			.WORD L10107-.
6729				
6730					: JUMP TO END OF TEST IF ERROR
6731				
6732					
6733	030072	012737	000002	002336	MOV #02,\$GDDAT
6734	030100	004537	003242		JSR R5,GETOUT
6735				
6736					: CHECK FOR INFORMATION CUT AND CORRECT TRIBN
6737				
6738					
6739	030104	005037	002402		CLR ERRWRD
6740	030110	104410			TRAP C\$ESCAPE
6741	030112	000220			.WORD L10107-.
6742				
6743					: JUMP TO END OF TEST IF ERROR
6744				
6745					
6746	030114	012737	000010	002336	MOV #10,\$GDDAT
6747	030122	004537	003372		JSR R5,GETRKY
6748				
6749					: CHECK FOR CORRECT RETURN KEY MODEM STATUS
6750				
6751					
6752	030126	005037	002402		CLR ERRWRD
6753	030132	104410			TRAP C\$ESCAPE
6754	030134	000176			.WORD L10107-.
6755				
6756					: JUMP TO END OF TEST IF ERROR
6757				
6758	030136	005037	002336		CLR \$GDDAT
6759	030142	005037	002340		CLR \$BDDAT
6760	030146	117737	152276	002340	MOVB @BSEL4,\$BDDAT
6761	030154	116237	030262	002336	MOVB MODOUT(R2),\$GDDAT
6762	030162	123737	002340	002336	CMPB \$BDDAT,\$GDDAT
6763	030170	001411			BEQ 10\$
6764	030172	012737	012014	002430	MOV #M30F,CODEW
6765	030200	104455			TRAP C\$ERDF
6766	030202	000055			.WORD 45
6767	030204	011506			.WORD EROIC

CZDMTC.P11 25-MAR-81 08:24

***** TEST 38 *****

6768 030206 010324
6769 030210 104410
6770 030212 000120
6771 030214 062702 000002
6772
6773 030220 022702 000024
6774 030224 001243
6775 030226
6776 030226 104432
6777 030230 000102
6778
6779 030232 000323
6780 030234 000313
6781 030236 000233
6782 030240 000133
6783 030242 000133
6784 030244 000233
6785 030246 000323
6786 030250 000323
6787 030252 000323
6788 030254 000313
6789 030256 000377
6790 030260 000100
6791
6792 030262 000310
6793 030264 000310
6794 030266 000330
6795 030270 000330
6796 030272 000330
6797 030274 000330
6798 030276 000330
6799 030300 000330
6800 030302 000310
6801 030304 000310
6802
6803
6804
6805 030306 000005
6806 030310 000005
6807 030312 000004
6808 030314 000004
6809 030316 000004
6810 030320 000004
6811 030322 000004
6812 030324 000004
6813 030326 000005
6814 030330 000005
6815
6816 030332
6817 030332 104401
6818
6819
6820
6821
6822

```

          .WORD  ERR27
          TRAP   C$ESCAPE
10$:      .WORD  L10107-.
          ADD    #2,R2

          CMP    #24,R2
          BNE   MODEB          ;IF NOT DONE GO TO B
MODEX:    TRAP   C$EXIT
          .WORD  L10107-.

MODTYP:   .WORD  323
          .WORD  313
          .WORD  233
          .WORD  133
          .WORD  133
          .WORD  233
          .WORD  323
          .WORD  323
          .WORD  323
          .WORD  313
MODIN:    .WORD  377
          .WORD  100

MODOUT:   .WORD  310
          .WORD  310
          .WORD  330
          .WORD  330
          .WORD  330
          .WORD  330
          .WORD  330
          .WORD  330
          .WORD  310
          .WORD  310

DUPTYP:   .WORD  5
          .WORD  5
          .WORD  4
          .WORD  4
          .WORD  4
          .WORD  4
          .WORD  4
          .WORD  4
          .WORD  4
          .WORD  5
          .WORD  5

L10107:   TRAP   C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 39 *****

6823
6824
6825 030334
6826
6827
6828
6829
6830
6831
6832
6833
6834
6835
6836
6837
6838 030334
6839
6840 030334 012737 004222 000004
6841 030342 005037 000006
6842 030346 012737 000002 002542
6843 030354 012746 000002
6844 030360 012705 030570
6845
6846 030364 005737 177572
6847
6848
6849 030370 062716 000002
6850 030374 012700 000000
6851 030400 104441
6852 030402 012701 172300
6853 030406 012702 000010
6854 030412
6855 030412 012721 077406
6856 030416 005302
6857 030420 001374
6858 030422 012701 172340
6859 030426 005011
6860 030430 012761 000200 000002
6861 030436 012761 000400 000004
6862 030444 012761 000600 000006
6863 030452 012761 001000 000010
6864 030460 012761 007600 000016
6865 030466 012761 002000 000012
6866 030474 005037 002432
6867 030500 012737 000033 002360
6868 030506 052737 040000 002432
6869 030514 012737 177776 002422
6870 030522 012737 001750 002424
6871 030530 012737 033454 002420
6872 030536 012737 001000 002426
6873 030544 004737 004522
6874 030550 004537 006346
6875 030554 005037 002402
6876 030560 104410
6877 030562 000012
6878 030564 004537 003744

.SBTTL :***** TEST 39 *****
.SBTTL TEST OF MEM EXTENSION BIT 16, ADDRESS 200000
ZZ
:
:
:
* THIS TEST WE'LL TRY TRANSMITTING A MESSAGE
* TO VIRTUAL ADDRESS 200000 (BIT 16 SET).
* IF MEMORY MANAGEMENT AND/OR SUFFICIENT MEMORY IS NOT
* AVAILABLE FOR MESSAGE STORAGE, WE WILL EXPECT THE DMP-11
* TO RETURN A NON-EXISTENT MEMORY ERROR FOR THE BUFFER.
* IF ENOUGH MEMORY EXISTS, WE'LL MAKE SURE THE TRANSFER TAKES
* PLACE PROPERLY.
:
:
*--

.SBTTL :***** TEST 39 *****
T39::

```

MOV #META,4 ;SET UP TRAP FOR NO MEM
CLR 6
MOV #2,$TMP0
MOV #2,-(SP) ;DUMMY MOVE TO STACK
MOV #RET16,R5 ;SET UP R5 FOR RETURN IF TRAP
; TAKES U TO SR.
TST @#177572 ;DOES MEM MANAGEMENT EXIST
; IF NOT TRAP TO META
; ELSE CONTINUE
;FIX THE STACK
ADD #2,(SP)
MOV #0,R0
TRAP C$SPRI
MOV #172300,R1 ;GET ADDRESS OF KERNEL PDR REQD/
MOV #8.,R2 ; DO 8 TIMES

10$:
MOV #77406,(R1)+
DEC R2
BNE 10$
MOV #172340,R1
CLR (R1)
MOV #200,2(R1)
MOV #400,4(R1)
MOV #600,6(R1)
MOV #1000,10(R1)
MOV #7600,16(R1)
MOV #2000,12(R1)
CLR GENWRD
MOV #33,TRIBN ;MAP MEM CLEAR GEN AND SET UP TRIB
;SET MM BIT
BIS #BIT14,GENWRD ;SET UP RX ADD
MOV #177776,RXADD ;SET UP RX COUNT(1000 DEC)
MOV #1750,RXCC ;SET UP TRANSMIT COUNT
MOV #MR1,TXADD ;SET UP TRANSMIT COUNT
MOV #512.,TXCC ; MASTER CLEAR MODE DEF
JSR PC,MINITS ;GO TRANSMIT AND RX
JSR R5,TXRX3 ; IF ERROR GO TO END
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10110-.
JSR R5,MEMEX ;CHECK MEM EXTENSION

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 39 *****

6879 030570 005037 002402
6880 030574
6881 030574 104401

RET16: CLR ERRWRD ;CLEAR ERROR WORD
L10110: TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 40 *****

6882
6883
6884 030576
6885
6886
6887
6888
6889
6890
6891
6892
6893
6894
6895
6896
6897 030576
6898
6899 030576 012737 004222 000004
6900 030604 005037 000006
6901 030610 012737 000004 002542
6902 030616 012746 000004
6903 030622 012705 031032
6904
6905 030626 005737 177572
6906
6907
6908 030632 062716 000002
6909 030636 012700 000000
6910 030642 104441
6911 030644 012701 172300
6912 030650 012702 000010
6913 030654
6914 030654 012721 077406
6915 030660 005302
6916 030662 001374
6917 030664 012701 172340
6918 030670 005011
6919 030672 012761 000200 000002
6920 030700 012761 000400 000004
6921 030706 012761 000600 000006
6922 030714 012761 001000 000010
6923 030722 012761 007600 000016
6924 030730 012761 004000 000012
6925 030736 005037 002432
6926 030742 012737 000033 002360
6927 030750 052737 040000 002432
6928 030756 012737 177776 002422
6929 030764 012737 041750 002424
6930 030772 012737 033454 002420
6931 031000 012737 001000 002426
6932 031006 004737 004522
6933 031012 004537 006346
6934 031016 005037 002402
6935 031022 104410
6936 031024 000012
6937 031026 004537 003744

```

.SBTTL ***** TEST 40 *****
.SBTTL TEST OF MEM EXTENSION BIT 17, ADDRESS 400000
ZZ
:
:
: * THIS TEST WE'LL TRY TRANSMITTING A MESSAGE
: * TO VIRTUAL ADDRESS 400000 (BIT 17 SET).
: * IF MEMORY MANAGEMENT AND/OR SUFFICIENT MEMORY IS NOT
: * AVAILABLE FOR MESSAGE STORAGE, WE WILL EXPECT THE DMP-11
: * TO RETURN A NON-EXISTENT MEMORY ERROR FOR THE BUFFER.
: * IF ENOUGH MEMORY EXISTS, WE'LL MAKE SURE THE TRANSFER TAKES
: * PLACE PROPERLY.
:
:
:
.SBTTL ***** TEST 40 *****
T40::
MOV #META,4 ;SET UP TRAP FOR NO MEM
(LR 5
MOV #4,$TMP0
MOV #4,-(SP) ;DUMMY MOVE ON STACK IF TRAP
MOV #RET17,R5 ;SET UP R5 FOR RETURN IF TRAP IS
; TO SR.EXMEM.
TST @#177572 ;DOES MEM MANAGEMENT EXIST
; IF NOT TRAP TO META
; ELSE CONTINUE
; FIX THE STACK
ADD #2,(SP)
MOV #0,R0
TRAP C$SPR1
MOV #172300,R1 ;GET ADDRESS OF KERNEL PDR REQD/
MOV #8.,R2 ; DO 8 TIMES
10$:
MOV #77406,(R1)+
DEC R2
BNE 10$
MOV #172340,R1
CLR (R1)
MOV #200,2(R1)
MOV #400,4(R1)
MOV #600,6(R1)
MOV #1000,10(R1)
MOV #7600,16(R1)
MOV #4000,12(R1)
CLR GENWRD
MOV #33,TRIBN ;MAP MEM CLEAR GEN AND SET UP TRIB
BIS #BIT14,GENWRD ;SET MM BIT
MOV #177776,RXADD ;SET UP RX ADD
MOV #41750,RXCC ;SET UP RX COUNT (1000 DEC)+BA16
MOV #MR1,TXADD ;SET UP TRANSMIT COUNT
MOV #512.,TXCC ;SET UP TRANSMIT COUNT
JSR PC,MINITS ; MASTER CLEAR MODE DEF
JSR R5,TXRX3 ;GO TRANSMIT AND RX
CLR ERRWRD ; IF ERROR GO TO END
TRAP C$ESCAPE
.WORD L10111-
JSR R5,MEMEX ;CHECK MEM EXTENSION

```

CZDMTC.P11 25-MAR-81 08:24

;***** TEST 40 *****

6938 031032 005037 002402
6939 031036
6940 031036 104401
6941

RET17: CLR ERRWRD ;CLEAR ERROR WORD
L10111: TRAP C\$ETST

CZDMTC.P11 25-MAR-81 08:24

***** TEST 41 *****

6942
6943
6944 031040
6945
6946
6947
6948
6949
6950
6951
6952
6953
6954
6955
6956
6957 031040
6958
6959 031040 012737 004222 000004
6960 031046 005037 000006
6961 031052 012737 000006 002542
6962 031060 012746 000006
6963
6964 031064 012705 031274
6965
6966 031070 005737 177572
6967
6968
6969 031074 062716 000002
6970 031100 012700 000000
6971 031104 104441
6972 031106 012701 172300
6973 031112 012702 000010
6974 031116
6975 031116 012721 077406
6976 031122 005302
6977 031124 001374
6978 031126 012701 172340
6979 031132 005011
6980 031134 012761 000200 000002
6981 031142 012761 000400 000004
6982 031150 012761 000600 000006
6983 031156 012761 001000 000010
6984 031164 012761 007600 000016
6985 031172 012761 006000 000012
6986 031200 005037 002432
6987 031204 012737 000033 002360
6988 031212 052737 040000 002432
6989 031220 012737 177776 002422
6990 031226 012737 101750 002424
6991 031234 012737 033454 002420
6992 031242 012737 001000 002426
6993 031250 004737 004522
6994 031254 004537 006346
6995 031260 005037 002402
6996 031264 104410
6997 031266 000012

```

.SBTTL ***** TEST 41 *****
.SBTTL TEST OF MEM EXTENSION BIT 16 AND 17, ADDRESS 600000
ZZ
:
:
: * THIS TEST WE'LL TRY TRANSMITTING A MESSAGE
: * TO VIRTUAL ADDRESS 600000 (BIT 16 AND 17 SET).
: * IF MEMORY MANAGEMENT AND/OR SUFFICIENT MEMORY IS NOT
: * AVAILABLE FOR MESSAGE STORAGE, WE WILL EXPECT THE DMP-11
: * TO RETURN A NON-EXISTENT MEMORY ERROR FOR THE BUFFER.
: * IF ENOUGH MEMORY EXISTS, WE'LL MAKE SURE THE TRANSFER TAKES
: * PLACE PROPERLY.
:
:
:
.SBTTL ***** TEST 41 *****
141::
MOV #META 4 ;SET UP TRAP FOR NO MEM
CLR 6
MOV #6,$TMP0
MOV #6,-(SP) ;DUMMY MOVE TO STACK IF
;TRAP TAKES U TO SR.
MOV #RET18,R5 ;SET UP R5 FOR RETURN
TST @#177572 ;DOES MEM MANAGEMENT EXIST
;IF NOT TRAP TO META
;ELSE CONTINUE
;FIX STACK
ADD #2,(SP)
MOV #0,R0
TRAP C$SPRI
MOV #172300,R1 ;GET ADDRESS OF KERNEL PDR REQD/
MOV #8.,R2 ;DO 8 TIMES
10$:
MOV #77406,(R1)+
DEC R2
BNE 10$
MOV #172340,R1
CLR (R1)
MOV #200,2(R1)
MOV #400,4(R1)
MOV #600,6(R1)
MOV #1000,10(R1)
MOV #7600,16(R1)
MOV #6000,12(R1)
CLR GENWRD
MOV #33,TRIBN ;MAP MEM CLEAR GEN AND SET UP TRIB
BIS #BIT14,GENWRD ;SET MM BIT
MOV #177776,RXADD ;SET UP RX ADD
MOV #101750,RXCC ;SET UP RX COUNT (1000 DEC)+BA17
MOV #MR1,TXADD ;SET UP TRANSMIT COUNT
MOV #512.,TXCC ;SET UP TRANSMIT COUNT
JSR PC,MINITS ; MASTER CLEAR MODE DEF
JSR R5,TXRX3 ;GO TRANSMIT AND RX
CLR ERRWRD ; IF ERROR GO TO END
TRAP C$ESCAPE
.WORD L10112-

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 41 *****

6998	031270	004537	003744
6999	031274	005037	002402
7000	031300		
7001	031300	104401	
7002			

RET18:	JSR	R5, MEMEX
L10112:	CLR	ERRWRD
	TRAP	CSETST

:	CHECK MEM EXTENSION
:	CLEAR ERROR WORD

CZDMTC.P11 25-MAR-81 08:24

***** TEST 42 *****

7003
7004
7005 031302
7006
7007
7008
7009
7010
7011
7012
7013
7014
7015
7016
7017 031302
7018
7019 031302 012737 000042 002370
7020
7021 031310 005037 002432
7022 031314 012737 000401 002426
7023 031322 012737 033455 002420
7024 031330 012737 033735 002422
7025 031336 012737 001750 002424
7026 031344 012737 000001 002360
7027 031352 022737 000004 002476
7028 031360 001402
7029 031362 005237 002400
7030 031366 004737 004512
7031 031372 005037 002400
7032 031376 005037 002402
7033 031402 104410
7034 031404 000012
7035
7036
7037
7038 031406 004537 005720
7039 031412 005037 002402
7040 031416
7041 031416 104401
7042

```

.SBTTL ***** TEST 42 *****
.SBTTL *TX AND RX 257 BYTES,ODD TX,ODD RX,DDCMP,POINT TO POINT
ZZ
:
:
: * THIS TEST WILL TRANSMIT A MESSAGE OF 257 BYTES
: * FROM A TX BUFFER STARTING WITH AN ODD BYTE TO
: * A RECEIVE BUFFER STARTING WITH AN ODD BYTE IN
: * DDCMP MODE,POINT TO POINT IF THERE IS EXTERNAL LOOP
: * BACK THEN THE TEST WILL BE DONE OVER THAT LOOPBACK
: * ELSE THE LOOPBACK WILL BE SET TO INTERNAL(TTL).
:
:
:
.SBTTL ***** TEST 42 *****
T42::
MOV #42,ROMN1 ;SET UP TEST NUMBER
CLR GENWRD ;CLEAR THE GEN WORD
MOV #257,TXCC ;SET UP TRANSMIT CHAR COUNT
MOV #MR1+1,TXADD ;SET UP ADD FOR TX
MOV #RECBU1+1,RXADD ;SET UP X
MOV #1000,RXCC ;SET UP R. COUNT 1000 DECIMAL
MOV #01,TRIBN ;SET TRIB # TO 1
CMP #4,TSTCON ;CHECK FOR LOOPBACK CONNECTOR
BEQ 1$
INC EXLOOP ;EXT CONNECTOR PRESENT: SET EXT LOOP FLAG
1$: JSR PC,MINIT1 ;MASTER CLEAR-MODE DEF
CLR EXLOOP ;CLEAR EXTERNAL LOOP FLAG
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10113-.
:
:
:ESCAPE TEST IF ERROR
:
:
20$: JSR R5,TXRXSR ;GO DO IT
CLR ERRWRD
L10113: TRAP C$ETST

```


CZDMTC.P11 25-MAR-81 08:24

***** TEST 43 *****

7043
7044
7045 031420
7046
7047
7048
7049
7050
7051
7052
7053
7054 031420
7055
7056 031420 012737 000001 002360
7057 031426 012737 000001 002426
7058 031434 012737 033455 002420
7059 031442 012737 000764 002424
7060 031450 012737 033734 002422
7061 031456 005037 002432
7062 031462 052737 100000 002432
7063 031470 004737 004522
7064 031474 005037 002402
7065 031500 104410
7066 031502 000012
7067
7068
7069
7070 031504 004537 006346
7071 031510 005037 002402
7072 031514
7073 031514 104401

```

.SBTTL :***** TEST 43 *****
.SBTTL *TX AND RX 1 BYTE,ODD TX,EVEN RX,MAINT,MULTIPOINT
ZZ
:*
:*
:* THIS TEST TRANSMITS AND RECEIVES 1 BYTE MESSAGE
:* FROM AN ODD TRANSMIT BUFFER TO AN EVEN RX BUFFER
:* IN MAINTAINCE MODE,MULTIPOINT
:*
:*-
.SBTTL :***** TEST 43 *****
T43::
MOV #01,TRIBN
MOV #01, TXCC
MOV #MR1+1, TXADD
MOV #500.,RXCC
MOV #RECBU1, RXADD
CLR GENWRD
BIS #BIT15, GENWRD ;SET UP TX RX AND MAINT STATE
JSR PC,MINITS ;MASTER CLEAR MULTIPOINT
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10114-.
: : : : :
: ESCAPE TEST IF ERROR
: : : : :
20$: JSR R5, TXRX3 ;GO DO IT
CLR ERRWRD ;CLEAR ERROR WORD
L10114: TRAP C$ETST

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 44 *****

7074
7075
7076 031516
7077
7078
7079
7080
7081
7082
7083
7084 031516
7085
7086 031516 005037 002432
7087 031522 004737 004522
7088 031526 005037 002402
7089 031532 104410
7090 031534 000774
7091
7092
7093
7094 031536 005037 002360
7095 031542 012703 000237
7096 031546 012704 000100
7097 031552 004537 003162
7098 031556 005037 002402
7099 031562 104410
7100 031564 000744
7101
7102
7103
7104
7105 031566 012737 000034 002360
7106 031574 012703 000001
7107 031600 004537 003162
7108 031604 005037 002402
7109 031610 104410
7110 031612 000716
7111
7112
7113
7114 031614 012703 000236
7115 031620 012704 002000
7116 031624 032737 000003 002472
7117 031632 001402
7118 031634 012704 000144
7119 031640 004537 003162
7120 031644 005037 002402
7121 031650 104410
7122 031652 000656
7123
7124
7125
7126 031654 012704 002010
7127 031660 012703 000234
7128 031664 004537 003162
7129 031670 005037 002402

```

.SBTTL :***** TEST 44 *****
.SBTTL POLLING STATE TESTS
ZZ
:*
:*
:* THIS TEST CHECKS THE DEGRADING OF THE POLLING
:* STATES FROM ACTIVE TO INACTIVE TO POTEN. DEAD
:* TO DEAD.
.SBTTL :***** TEST 44 *****
T44::
BEGPOL: CLR GENWRD ;CLEAR FLAG WORD
JSR PC,MINITS ;MASTER CLEAR MODE DEF(FD/CS/MP)
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
:
:
:SET POLL DELAY
:
:
CLR TRIBN
MOV #237,R3
MOV #100,R4
JSR R5,CONTIN
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
:
:
: ESTABLISH TRIB
:
MOV #34,TRIBN ;SET TRIB NO.
MOV #01,R3
JSR R5,CONTIN
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
:
:
: SET SELECTION TIMER TO 1 SEC
:
MOV #236,R3
MOV #2000,R4
BIT #3,OPTYP ;* IS THIS A DMV ?
BEQ 1$ ;*
MOV #144,R4 ;* IF YES: ADJUST VALUE.
JSR R5,CONTIN
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
:
:
: WRITE NDM > INACTIVE AND #TO > PDEAD
:
MOV #2010,R4
MOV #234,R3
JSR R5,CONTIN ;WRITE TSS SLOT
CLR ERRWRD

```

CZDMTC.P11 25-MAR-81 08:24

:***** TEST 44 *****

7130	031674	104410				TRAP	C\$ESCAPE	
7131	031676	000632				.WORD	L10115-	
7132						:	:	:
7133						:	ISTRIB	TRIB
7134						:	:	:
7135	031700	012703	000003			MOV	#03,R3	
7136	031704	004537	003162			JSR	R5,CONTIN	
7137	031710	005037	002402			CLR	ERRWRD	
7138	031714	104410				TRAP	C\$ESCAPE	
7139	031716	000612				.WORD	L10115-	
7140						:	:	:
7141						:	WAIT FOR RUN STATE	
7142						:	:	:
7143	031720	012737	000001	002336		MOV	#01,\$GDDAT	
7144	031726	004537	003242			JSR	R5,GETOUT	
7145	031732	005037	002402			CLR	ERRWRD	
7146	031736	104410				TRAP	C\$ESCAPE	
7147	031740	000570				.WORD	L10115-	
7148	031742	012737	000024	002336		MOV	#24,\$GDDAT	
7149	031750	004537	003670			JSR	R5,GETOC	
7150	031754	005037	002402			CLR	ERRWRD	
7151	031760	104410				TRAP	C\$ESCAPE	
7152	031762	000546				.WORD	L10115-	
7153						:	:	:
7154						:	READ TSS SLOT WITH POLL STATUS	
7155						:	:	:
7156	031764	042777	000200	150452	20\$:	BIC	#RDO,@BSEL2	;CLEAR RDO
7157	031772	012703	000042			MOV	#42,R3	
7158	031776	004537	003162			JSR	R5,CONTIN	;READ TSS SLOT 2
7159	032002	005037	002402			CLR	ERRWRD	
7160	032006	104410				TRAP	C\$ESCAPE	
7161	032010	000520				.WORD	L10115-	
7162	032012	012737	000002	002336		MOV	#02,\$GDDAT	
7163	032020	004537	003242			JSR	R5,GETOUT	;GET INFO OUT
7164	032024	005037	002402			CLR	ERRWRD	
7165	032030	104410				TRAP	C\$ESCAPE	
7166	032032	000476				.WORD	L10115-	
7167	032034	032777	001000	150406		BIT	#BIT9,@BSEL4	;IS IT INACTIVE
7168	032042	001750				BEQ	20\$;IF NOT GO BACK
7169						:	:	:
7170						:	GET HERE WHEN STATE GOES TO INACTIVE	
7171						:	:	:
7172	032044	042777	000200	150372		BIC	#RDO,@BSEL2	;CLEAR OUTPUT
7173						:	:	:
7174						:	READ # OF SELECTION INTERVALS	
7175						:	:	:
7176	032052	012703	000051			MOV	#51,R3	
7177	032056	004537	003162			JSR	R5,CONTIN	
7178	032062	005037	002402			CLR	ERRWRD	
7179	032066	104410				TRAP	C\$ESCAPE	
7180	032070	000440				.WORD	L10115-	
7181	032072	012737	000002	002336		MOV	#02,\$GDDAT	
7182	032100	004537	003242			JSR	R5,GETOUT	
7183						:	:	:
7184						:	MAKE SURE #OF SELC. INTV IS CORRECT	
7185						:	:	:

CZDMTC.P11 25-MAR-81 08:24

***** TEST 44 *****

7186	032104	012737	000010	002336		MOV	#10,\$GDDAT	
7187	032112	004537	003446			JSR	R5,GETDAT	
7188							
7189						:	GET HERE IN INACTIVE STATE	
7190							
7191	032116				40\$:			
7192	032116	042777	000200	150320		BIC	#RDO,@BSEL2	;CLEAR OUTPUT
7193							
7194						:	CHANGE MODE TO HALF DUPLEX	
7195							
7196	032124	052777	000200	150306		BIS	#RQI,@BSELO	
7197	032132	004537	002732			JSR	R5,WRDI	;WAIT FOR RDI TO SET
7198								
7199	032136	005037	002402			CLR	ERRWRD	;CLEAR ERROR
7200	032142	104410				TRAP	C\$ESCAPE	
7201	032144	000364				.WORD	L10115-	
7202	032146	042777	000200	150264		BIC	#RQI,@BSELO	
7203	032154	012777	000004	150272		MOV	#04,@BSEL6	
7204	032162	112777	000002	150254		MOVB	#02,@BSEL2	
7205	032170	042777	000200	150246	50\$:	BIC	#RDO,@BSEL2	
7206								
7207								
7208							
7209						:	READ POLL STATUS SLOT	
7210							
7211	032176	012703	000042			MOV	#42,R3	
7212	032202	004537	003162			JSR	R5,CONTIN	
7213	032206	005037	002402			CLR	ERRWRD	
7214	032212	104410				TRAP	C\$ESCAPE	
7215	032214	000314				.WORD	L10115-	
7216	032216	012737	000002	002336		MOV	#02,\$GDDAT	
7217								
7218	032224	004537	003242			JSR	R5,GETOUT	
7219							
7220						:	IS THE STATE POTN. DEAD??	
7221							
7222	032230	032777	010000	150212		BIT	#BIT12,@BSEL4	
7223	032236	001754				BEQ	50\$	
7224							
7225						:	IF NOT GO BACK TO 50	
7226							
7227								
7228							
7229						:	IF SO READ THE SELECTION TIMER	
7230							
7231								
7232	032240	012703	000056			MOV	#56,R3	
7233	032244	042777	000200	150172		BIC	#RDO,@BSEL2	
7234	032252	004537	003162			JSR	R5,CONTIN	
7235	032256	005037	002402			CLR	ERRWRD	
7236	032262	104410				TRAP	C\$ESCAPE	
7237	032264	000244				.WORD	L10115-	
7238	032266	012737	000002	002336		MOV	#02,\$GDDAT	
7239	032274	004537	003242			JSR	R5,GETOUT	
7240	032300	005037	002402			CLR	ERRWRD	
7241	032304	104410				TRAP	C\$ESCAPE	

CZDMTC.P11 25-MAR-81 08:24

***** TEST 44 *****

7242 032306 000222
7243
7244
7245
7246
7247 032310 012737 001004 002336
7248 032316 004537 003446
7249
7250
7251
7252 032322 042777 000200 150114
7253
7254 032330 012737 000001 002336
7255 032336 004537 003242
7256 032342 005037 002402
7257 032346 104410
7258 032350 000160
7259 032352 012737 000006 002336
7260 032360 004537 003670
7261 032364 005037 002402
7262 032370 104410
7263 032372 000136
7264 032374 042777 000200 150042
7265
7266
7267
7268 032402 012737 000001 002336
7269 032410 004537 003242
7270 032414 005037 002402
7271 032420 104410
7272 032422 000106
7273 032424 012737 000022 002336
7274 032432 004537 003670
7275
7276 032436 005037 002402
7277 032442 104410
7278 032444 000064
7279
7280
7281
7282
7283
7284
7285 032446 042777 000200 147770
7286 032454 012703 000056
7287 032460 004537 003162
7288 032464 005037 002402
7289 032470 104410
7290 032472 000036
7291 032474 012737 000002 002336
7292 032502 004537 003242
7293 032506 005037 002402
7294 032512 104410
7295 032514 000014
7296 032516 012737 001010 002336
7297 032524 004537 003446

```

.WORD L10115-.
:
: COMPARE SELECTION TIME OUTS WITH # WRITTEN
:
:
:
MOV #1004,$GDDAT
JSR R5,GETDAT
:
: WAIT FOR TRIB TO POST SELECT. THRESH. ERROR
:
:
BIC #RDO,@BSEL2 ;CLEAR OUTPUT CODE
:
MOV #01,$GDDAT
JSR R5,GETOUT
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
MOV #6,$GDDAT
JSR R5,GETOC
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
BIC #RDO,@BSEL2
:
: NOW WAIT FOR TRIB TO POST DEAD STATUS
:
:
MOV #01,$GDDAT
JSR R5,GETOUT
CLR ERRWRD
TRAP C$ESCA
.WORD L10115-.
MOV #22,$GDDAT
JSR R5,GETOC
:
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
:
:
: NOW READ SELECTION TIMER AND
: SEE IF IT IS EQUAL TO 10
:
:
BIC #RDO,@BSEL2
MOV #56,R3
JSR R5,CONTIN
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
MOV #02,$GDDAT
JSR R5,GETOUT
CLR ERRWRD
TRAP C$ESCAPE
.WORD L10115-.
MOV #1010,$GDDAT
JSR R5,GETDAT

```

CZDMTC.P11 25-MAR-81 08:24

***** TEST 44 *****

7298 032530
7299 032530 104401
7300
7301
7302
7303
7304
7305

L10115:

TRAP C\$ETST

.SBTTL ----- END OF HARDWARE TESTS -----

CZDMTC.P11 25-MAR-81 08:24

----- END OF HARDWARE TESTS -----

7306
7307
7308
7309
7310
7311
7312
7313
7314
7315
7316
7317
7318
7319
7320 032532 000041
7321 032534
7322 032534 000032
7323 032536 032636
7324 032540 000007
7325 032542 000000
7326 032544 000007
7327 032546 001031
7328 032550 032730
7329 032552 160000
7330 032554 177776
7331 032556 002031
7332 032560 032756
7333 032562 000000
7334 032564 000674
7335 032566 003032
7336 032570 033007
7337 032572 007000
7338 032574 000004
7339 032576 000007
7340 032600 010032
7341 032602 033040
7342 032604 000007
7343 032606 000000
7344 032610 000004
7345 032612 011032
7346 032614 033144
7347 032616 000007
7348 032620 000000
7349 032622 000007
7350 032624 013032
7351 032626 033355
7352 032630 000007
7353 032632 000000
7354 032634 000007
7355
7356
7357
7358 032636
7359
7360 032636 042523 042514 052103
7361 032644 047440 052120 047511

.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.
://////

```

```

L$HARD: .WORD L10116-L$HARD/2

```

```

.WORD T$CODE
.WORD OPTYPM
.WORD 7
.WORD T$LLOLIM
.WORD T$HILIM
.WORD T$CODE
.WORD ADDRES
.WORD T$LLOLIM
.WORD T$HILIM
.WORD T$CODE
.WORD VECTOR
.WORD T$LLOLIM
.WORD T$HILIM
.WORD T$CODE
.WORD PRIRTY
.WORD 7000
.WORD T$LLOLIM
.WORD T$HILIM
.WORD T$CODE
.WORD LOOPBK
.WORD 7
.WORD T$LLOLIM
.WORD T$HILIM
.WORD T$CODE
.WORD IFTYPM
.WORD 7
.WORD T$LLOLIM
.WORD T$HILIM

```

```

.EVEN
L10116:

```

```

OPTYPM: .ASCIZ /SELECT OPTION TYPE ( 0=8207'DMP',1=8053'DMV',2 8064'DMV):/

```

CZDMTC.P11 25-MAR-81 08:24

HARDWARE PARAMETER CODING SECTION

7362	032652	020116	054524	042520	
7363	032660	024040	030040	034075	
7364	032666	030062	023467	046504	
7365	032674	023520	030454	034075	
7366	032702	032460	023463	046504	
7367	032710	023526	031054	034075	
7368	032716	033060	023464	046504	
7369	032724	024526	000072		
7370	032730	042504	044526	042503	ADDRESS: .ASCIZ /DEVICE CSR ADDRESS : /
7371	032736	041440	051123	040440	
7372	032744	042104	042522	051523	
7373	032752	035040	000040		
7374	032756	042504	044526	042503	VECTOR: .ASCIZ /DEVICE VECTOR ADDRESS : /
7375	032764	053040	041505	047524	
7376	032772	020122	042101	051104	
7377	033000	051505	020123	020072	
7378	033006	000			
7379	033007	104	053105	041511	PRIPTY: .ASCIZ /DEVICE PRIORITY LEVEL : /
7380	033014	020105	051120	047511	
7381	033022	044522	054524	046040	
7382	033030	053105	046105	035040	
7383	033036	000040			
7384	033040	052524	047122	051101	LOOPBK: .ASCII /TURNAROUND TYPE -/
7385	033046	052517	042116	052040	
7386	033054	050131	020105	055	
7387	033061	050	036460	031510	.ASCIZ /(0=H3254&H3255,1=CABLE,2-MOD LOC,3-MOD REM,4=NONE)/
7388	033066	032462	023064	031510	
7389	033074	032462	026065	036461	
7390	033102	040503	046102	026105	
7391	033110	036462	047515	020104	
7392	033116	047514	026103	036463	
7393	033124	047515	020104	042522	
7394	033132	026115	036464	047516	
7395	033140	042516	000051		
7396	033144	046120	040505	042523	SPEM: .ASCII 'PLEASE SELECT BAUD RATE;TYPE '0' FOR 2.4K; '1' FOR 4.8K;''
7397	033152	051440	046105	041505	
7398	033160	020124	040502	042125	
7399	033166	051040	052101	035505	
7400	033174	054524	042520	023440	
7401	033202	023460	043040	051117	
7402	033210	031040	032056	035513	
7403	033216	023440	023461	043040	
7404	033224	051117	032040	034056	
7405	033232	035513			
7406	033234	005015	031047	020047	.ASCII<15><12>'2' FOR 9.6K; '3' FOR 19.2K; '4' FOR 56K; '5' FOR 250K;''
7407	033242	047506	020122	027071	
7408	033250	045466	020073	031447	
7409	033256	020047	047506	020122	
7410	033264	034461	031056	035513	
7411	033272	023440	023464	043040	
7412	033300	051117	032440	045466	
7413	033306	020073	032447	020047	
7414	033314	047506	020122	032462	
7415	033322	045460	073		
7416	033325	015	047412	020122	.ASCIZ<15><12>'OR '6' FOR 500K BAUDS''
7417	033332	033047	020047	047506	

CZDMTC.P11

25-MAR-81 08:24

HARDWARE PARAMETER CODING SECTION

7418	033340	020122	030065	045460
7419	033346	041040	052501	051504
7420	033354	000		
7421	033355	123	046105	041505
7422	033362	020124	047111	042524
7423	033370	043122	041501	020105
7424	033376	054524	042520	024040
7425	033404	036461	047111	042524
7426	033412	051107	046101	031054
7427	033420	042475	040511	031454
7428	033426	053075	031456	026065
7429	033434	036464	031064	024462
7430	033442	000072		
7431				
7432				
7433				
7434				
7435				

IFTYPM: .ASCII /SELECT INTERFACE TYPE (1=INTEGRAL,2=EIA,3=V.35,4=422):/

.EVEN

CZDMTC.P11 25-MAR-81 08:24

SOFTWARE PARAMETER CODING SECTION

.SBTTL SOFTWARE PARAMETER CODING SECTION

7436
7437
7438
7439
7440
7441
7442
7443
7444
7445
7446
7447
7448 033444 000000
7449 03344E
7450
7451
7452 033446
7453
7454
7455
7456
7457
7458
7459
7460
7461
7462
7463 033446
7464 033446 000240
7465 033450 000240
7466 033452 000240
7467
7468
7469
7470 033454 041101 000103
7471 033460
7472
7473 033460 047516 020127 051511
7474 033466 052040 042510 052040
7475 033474 046511 020105 047506
7476 033502 020122 046101 020114
7477 033510 047507 042117 050040
7478 033516 047505 046120 020105
7479 033524 047524 041440 046517
7480 033532 020105 047524 052040
7481 033540 042510
7482 033542 044124 020105 052521
7483 033550 041511 020113 051102
7484 033556 053517 020116 047506
7485 033564 020130 052512 050115
7486 033572 042105 047440 042526
7487 033600 020122 044124 020105
7488 033606 040514 054532 042040
7489 033614 043517 051447
7490 033620 040502 045503 020056
7491 033626 047514 051520 040440

```

://////
:/ 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.
://////

```

.WORD L10117-L\$SOFT/2

L\$SOFT::

.EVEN

L10117:

:***** PATCH AREA FOR DEBUG *****

PATCH:

NOP
NOP
NOP

:*****

MR1: .ASCIZ 'ABC'

MR1E:

MR12: .ASCII 'NOW IS THE TIME FOR ALL GOOD PEOPLE TO COME TO THE''

.ASCII 'THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG'S''

.ASCII! 'BACK. LOPS ARE TOPS!''

CZDMTC.P11

25-MAR-81 08:24

SOFTWARE PARAMETER CODING SECTION

7492	033634	042522	052040	050117	
7493	033642	020523			
7494	033644	041101	042103	043105	.ASCII 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
7495	033652	044107	045111	046113	
7496	033660	047115	050117	051121	
7497	033666	052123	053125	054127	
7498	033674	055131			
7499	033676	041501	043505	045511	.ASCII 'ACEGIKMOQSUWY'
7500	033704	047515	051521	053525	
7501	033712	131			
7502					
7503		033714			.EVEN
7504	033714				MR12E:
7505					
7506	033714	000000			DATLST: 0
7507	033716	125252			125252
7508	033720	052525			052525
7509	033722	000000			0
7510	033724	177777			-1
7511	033726	000377			377
7512	033730	177400			177400
7513	033732	000562			562
7514					
7515	033734	001750			RECBU1: .BLKW 1000.
7516					
7517					.EVEN
7518	037654	000000			.WORD 0
7519	037656	000000			.WORD 0
7520	037660				L\$LAST::
7521					
7522		000001			.END

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- USER SYMBOLS

		2680*	2801	4231	4233	4318*	4495*	4616*	4672*	4742*	4761	4807	4825*	4901*
		5575*	6329*	6385*	6453*	6500*	7203*							
BSEL7	002456	1501#	3461*	3462*	3711	3714	3734	3747	3758	3861	3864	3884	3897	3908
		4011	4014	4034	4047	4058	4305*	4319*	4332	4372				
CADDR	002404	1472#	2113	2117	3631*	3637*	3640*	3641	3656*	3670*	3681*	3706*	3712*	3715*
		3716	3731*	3745*	3756*	3781*	3787*	3790*	3791	3806*	3820*	3831*	3856*	3862*
		3865*	3866	3881*	3895*	3906*	3931*	3937*	3940*	3941	3956*	3970*	3981*	4006*
		4012*	4015*	4016	4031*	4045*	4056*							
CFM1	005306	2327	2356#											
CFM2	005373	2336	2365#											
CFM3	005451	2344	2373#											
CFM4	005466	2350	2376#											
CODEW	002430	1483#	1724*	1738*	1817*	1858*	1888*	1988*	2032*	3017	4694*	4754*	4764*	5687*
		6764*												
CONTIN	003162	1772#	2494	2592	2601	2612	2633	4868	4948	4965	4992	5038	5053	5067
		5147	5160	5209	5222	5271	5284	5340	5399	5413	5435	5486	5533	5542
		5579	5618	5645	5658	5695	5708	5731	5767	5780	5879	5926	5939	5959
		5979	6023	6036	6056	6076	6122	6136	6149	6203	6216	6366	6427	6437
		6695	6708	6721	7097	7107	7119	7128	7136	7158	7177	7212	7234	7287
COUNT	002342	1454#	3484*	3489*	4214*	4220*								
COUNTT	005134	2293*	2300*	2306#										
CRCCAL	002434	1485#	2411*	2432										
CRCR	004376	2158#	2419	3644	3719	3794	3869	3944	4019					
CWORD	002374	1468#	2162*	2166	2167*	2168*	2393*	2433	2963	3633*	3647*	3648	3708*	3722*
		3723	3783*	3797*	3798	3858*	3872*	3873	3933*	3947*	3948	4008*	4022*	4023
C\$AU =	000052	1094#	3543											
C\$AUTO=	000061	1094#	3497											
C\$BRK =	000022	1094#	2303											
C\$BSEG=	000004	1094#												
C\$BSUB=	000002	1094#	5321	5380	5470									
C\$CEFG=	000045	1094#												
C\$CLCK=	000062	1094#												
C\$CLEA=	000012	1094#	3510											
C\$CLOS=	000035	1094#												
C\$CLP1=	000006	1094#												
C\$CVEC=	000036	1094#	3591	4482	4484									
C\$DCLN=	000044	1094#	3588											
C\$DODU=	000051	1094#	3492											
C\$DRPT=	000024	1094#												
C\$DU =	000053	1094#	3527											
C\$EDIT=	000003	1094#	1171											
C\$ERDF=	000055	1094#	1638	1681	1728	1742	1818	1830	1859	1889	1922	1958	1989	2033
		2046	2062	2069	2295	2435	2745	3601	3652	3662	3686	3727	3737	3761
		3802	3812	3836	3877	3887	3911	3952	3962	3986	4027	4037	4061	4093
		4139	4225	4236	4246	4346	4384	4510	4527	4549	4559	4630	4681	4695
		4755	4765	4810	4906	5301	5688	5992	6089	6765				
C\$ERHR=	000056	1094#												
C\$ERRO=	000060	1094#												
C\$ERSF=	000054	1094#												
C\$ERSO=	000057	1094#	2250											
C\$ESCA=	000010	1094#	4086	4098	4132	4143	4229	4242	4250	4259	4268	4303	4311	4326
		4350	4366	4388	4432	4468	4502	4515	4531	4595	4609	4655	4665	4686
		4700	4730	4738	4746	4759	4793	4822	4832	4858	4885	4895	4941	4954
		4971	4983	4998	5030	5044	5059	5073	5083	5094	5104	5139	5153	5166
		5176	5201	5215	5228	5239	5263	5277	5290	5332	5343	5354	5366	5390
		5402	5416	5425	5438	5449	5458	5478	5489	5499	5529	5539	5547	5555

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- USER SYMBOLS

L\$DUT	002072	G	1187#		
L\$DVTY	002564	G	1178	1568#	
L\$EF	002052	G	1172#		
L\$ENVI	002044	G	1165#		
L\$ETP	002102	G	1195#		
L\$EXP1	002046	G	1167#		
L\$EXP4	002064	G	1181#		
L\$EXP5	002066	G	1183#		
L\$HARD	032534	G	1144	7320	7321#
L\$HIME	002120	G	1209#		
L\$HPCP	002016	G	1143#		
L\$HPTP	002022	G	1147#		
L\$HW	002264	G	1148	1284	1285#
L\$ICP	002104	G	1197#		
L\$INIT	013774	G	1198	3393#	
L\$LADP	002026	G	1151#		
L\$LAST	037660	G	1152	7520#	
L\$LOAD	002100	G	1193#		
L\$LUN	002074	G	1189#		
L\$MREV	002050	G	1169#		
L\$NAME	002000	G	1126#		
L\$PRIO	002042	G	1163#		
L\$PROT	002122	G	1204	1212#	
L\$PRT	002112	G	1203#		
L\$REPP	002062	G	1179#		
L\$REV	002010	G	1135#		
L\$RPT	013766	G	3372#		
L\$SOFT	033446	G	7448	7449#	
L\$SPC	002056	G	1175#		
L\$SPCP	002020	G	1145#		
L\$SPTP	002024	G	1149#		
L\$STA	002030	G	1153#		
L\$SW	002316	G	1318	1319#	
L\$TEST	002114	G	1205#		
L\$TIML	002014	G	1141#		
L\$UNIT	002012	G	1139#	3438	
L10001	002314		1284	1305#	
L10002	002316		1318	1323#	
L10003	007374		2794#		
L10004	007426		2808#		
L10005	007464		2820#		
L10006	007522		2832#		
L10007	007550		2844#		
L10010	007576		2856#		
L10011	007644		2874#		
L10012	007704		2889#		
L10013	007736		2902#		
L10014	010020		2920#		
L10015	010112		2941#		
L10016	010160		2959#		
L10017	010222		2972#		
L10020	010264		2985#		
L10021	010322		2999#		
L10022	010420		3024#		
L10023	010446		3036#		
L10024	013772		3376	3379#	

CZDMTC.P11

25-MAR-81 08:24

CROSS REFERENCE TABLE -- USER SYMBOLS

3607	3608	3609	3610	3611	3612	3613	3614	3615	3652	3653	3654	3655
3656	3662	3663	3664	3665	3666	3673	3674	3675	3676	3677	3678	3679
3680	3681	3686	3687	3688	3689	3690	3692	3693	3727	3728	3729	3730
3731	3737	3738	3739	3740	3741	3748	3749	3750	3751	3752	3753	3754
3755	3756	3761	3762	3763	3764	3765	3767	3768	3802	3803	3804	3805
3806	3812	3813	3814	3815	3816	3823	3824	3825	3826	3827	3828	3829
3830	3831	3836	3837	3838	3839	3840	3842	3843	3877	3878	3879	3880
3881	3887	3888	3889	3890	3891	3898	3899	3900	3901	3902	3903	3904
3905	3906	3911	3912	3913	3914	3915	3917	3918	3952	3953	3954	3955
3956	3962	3963	3964	3965	3966	3973	3974	3975	3976	3977	3978	3979
3980	3981	3986	3987	3988	3989	3990	3992	3993	4027	4028	4029	4030
4031	4037	4038	4039	4040	4041	4048	4049	4050	4051	4052	4053	4054
4055	4056	4061	4062	4063	4064	4065	4067	4068	4086	4087	4088	4093
4094	4095	4096	4097	4098	4099	4100	4101	4102	4103	4104	4105	4106
4107	4108	4109	4114	4115	4132	4133	4134	4139	4140	4141	4142	4143
4144	4145	4146	4147	4148	4149	4150	4151	4152	4153	4154	4158	4159
4225	4226	4227	4228	4229	4230	4231	4236	4237	4238	4239	4240	4242
4243	4244	4246	4247	4248	4249	4250	4251	4252	4259	4260	4261	4268
4269	4270	4271	4272	4303	4304	4305	4311	4312	4313	4326	4327	4328
4346	4347	4348	4349	4350	4351	4352	4366	4367	4368	4384	4385	4386
4387	4388	4389	4390	4407	4408	4409	4410	4411	4412	4413	4415	4416
4417	4418	4419	4420	4421	4425	4426	4427	4432	4433	4434	4446	4447
4448	4449	4450	4451	4452	4453	4454	4455	4456	4457	4458	4460	4461
4462	4468	4469	4470	4481	4482	4483	4484	4485	4502	4503	4504	4510
4511	4512	4513	4514	4515	4516	4517	4527	4528	4529	4530	4531	4532
4533	4536	4537	4538	4540	4541	4542	4544	4545	4547	4548	4549	4550
4551	4552	4553	4554	4555	4557	4558	4559	4560	4561	4562	4563	4564
4565	4567	4568	4569	4571	4572	4576	4577	4595	4596	4597	4609	4610
4611	4630	4631	4632	4633	4634	4639	4640	4655	4656	4657	4665	4666
4667	4681	4682	4683	4684	4685	4686	4687	4688	4695	4696	4697	4698
4699	4700	4701	4702	4709	4710	4730	4731	4732	4738	4739	4740	4746
4747	4748	4755	4756	4757	4758	4759	4760	4761	4765	4766	4767	4768
4769	4771	4772	4793	4794	4795	4810	4811	4812	4813	4814	4815	4816
4822	4823	4824	4832	4833	4834	4840	4841	4858	4859	4860	4885	4886
4887	4895	4896	4897	4906	4907	4908	4909	4910	4912	4913	4941	4942
4943	4954	4955	4956	4971	4972	4973	4983	4984	4985	4998	4999	5000
5011	5012	5030	5031	5032	5044	5045	5046	5059	5060	5061	5073	5074
5075	5083	5084	5085	5094	5095	5096	5104	5105	5106	5121	5122	5139
5140	5141	5153	5154	5155	5166	5167	5168	5176	5177	5178	5183	5184
5201	5202	5203	5215	5216	5217	5228	5229	5230	5239	5240	5241	5246
5247	5263	5264	5265	5277	5278	5279	5290	5291	5292	5301	5302	5303
5304	5305	5308	5309	5321	5322	5332	5333	5334	5343	5344	5345	5354
5355	5356	5366	5367	5368	5378	5379	5380	5381	5390	5391	5392	5402
5403	5404	5416	5417	5418	5425	5426	5427	5438	5439	5440	5449	5450
5451	5458	5459	5460	5468	5469	5470	5471	5478	5479	5480	5489	5490
5491	5499	5500	5501	5510	5511	5512	5513	5529	5530	5531	5539	5540
5541	5547	5548	5549	5555	5556	5557	5560	5561	5562	5571	5572	5573
5584	5585	5586	5590	5591	5592	5595	5596	5597	5600	5601	5602	5606
5607	5608	5612	5613	5614	5623	5624	5637	5638	5639	5651	5652	5653
5664	5665	5666	5675	5676	5677	5688	5689	5690	5691	5692	5701	5702
5703	5714	5715	5716	5724	5725	5726	5737	5738	5739	5748	5749	5750
5759	5760	5761	5773	5774	5775	5786	5787	5788	5796	5797	5798	5803
5804	5824	5825	5826	5834	5835	5836	5842	5843	5844	5853	5854	5871
5872	5873	5882	5883	5884	5892	5893	5894	5899	5900	5918	5919	5920
5932	5933	5934	5942	5943	5944	5952	5953	5954	5962	5963	5964	5972
5973	5974	5982	5983	5984	5992	5993	5994	5995	5996	5998	5999	6015

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- USER SYMBOLS

TYLST 002414
T\$ARGC= 000004

1476#	5819												
1127#	1128#	1129#	1130#	1131#	1132#	2325#	2331	2332#	2340	2343#	2348	2350#	
2354	2562#	2566	2573#	2577	2579#	2584	2788#	2793	2801#	2806	2812#	2818	
2824#	2830	2838#	2842	2850#	2854	2864#	2872	2880#	2887	2895#	2900	2906#	
2910	2911#	2918	2924#	2928	2929#	2934	2935#	2939	2949#	2957	2963#	2970	
2976#	2983	2991#	2997	3007#	3013	3014#	3022	3030#	3034	3607#	3612	3673#	
3680	3748#	3755	3823#	3830	3898#	3905	3973#	3980	4048#	4055	4101#	4108	
4146#	4153												

T\$CODE= 013032
T\$ERRN= 000055

7322#	7327#	7331#	7335#	7340#	7345#	7350#							
1094#	1639#	1682#	1729#	1743#	1819#	1831#	1860#	1890#	1923#	1959#	1990#	2034#	
2047#	2063#	2070#	2251#	2296#	2436#	2746#	3602#	3653#	3663#	3687#	3728#	3738#	
3762#	3803#	3813#	3837#	3878#	3888#	3912#	3953#	3963#	3937#	4028#	4038#	4062#	
4094#	4140#	4226#	4237#	4247#	4347#	4385#	4511#	4528#	4550#	4560#	4631#	4682#	
4696#	4756#	4766#	4811#	4907#	5302#	5689#	5993#	6090#	6766#				

T\$EXCP= 000000
T\$FLAG= 000040

7322#	7327#	7331#	7335#	7340#	7345#	7350#	7355						
3375#	3377	4086#	4098#	4132#	4143#	4229#	4242#	4250#	4259#	4268#	4303#	4311#	
4326#	4350#	4366#	4388#	4432#	4468#	4502#	4515#	4531#	4536#	4595#	4609#	4655#	
4665#	4686#	4700#	4730#	4738#	4746#	4759#	4793#	4814#	4822#	4832#	4858#	4885#	
4895#	4941#	4954#	4971#	4983#	4998#	5030#	5044#	5059#	5073#	5083#	5094#	5104#	
5139#	5153#	5166#	5176#	5201#	5215#	5228#	5239#	5263#	5277#	5290#	5332#	5343#	
5354#	5366#	5390#	5402#	5416#	5425#	5438#	5449#	5458#	5478#	5489#	5499#	5529#	
5539#	5547#	5555#	5560#	5571#	5584#	5590#	5595#	5600#	5606#	5612#	5637#	5651#	
5664#	5675#	5701#	5714#	5724#	5737#	5748#	5759#	5773#	5786#	5796#	5824#	5834#	
5842#	5871#	5882#	5892#	5918#	5932#	5942#	5952#	5962#	5972#	5982#	6015#	6029#	
6039#	6049#	6059#	6069#	6079#	6114#	6128#	6142#	6155#	6169#	6182#	6195#	6209#	
6222#	6235#	6248#	6288#	6296#	6315#	6325#	6336#	6358#	6372#	6382#	6393#	6419#	
6430#	6440#	6450#	6461#	6485#	6496#	6508#	6547#	6556#	6596#	6605#	6637#	6647#	
6671#	6682#	6701#	6714#	6727#	6740#	6753#	6769#	6776#	6876#	6935#	6996#	7033#	
7065#	7089#	7099#	7109#	7121#	7130#	7138#	7146#	7151#	7160#	7165#	7179#	7200#	
7214#	7236#	7241#	7257#	7262#	7271#	7277#	7289#	7294#					

T\$GMAN= 000000
T\$HILI= 000007

1094#													
7322#	7326	7327#	7330	7331#	7334	7335#	7339	7340#	7344	7345#	7349	7350#	
7354													

T\$LAST= 000001
T\$LOLI= 000000

1094#	7518#												
7322#	7325	7327#	7329	7331#	7333	7335#	7338	7340#	7343	7345#	7348	7350#	
7353													

T\$LSYM= 010000

1094#	1306	1324	2795	2809	2821	2833	2845	2857	2875	2890	2903	2921	
2942	2960	2973	2986	3000	3025	3037	3380	3479	3497	3510	3527	3543	
3594	3614	3692	3767	3842	3917	3992	4067	4114	4158	4271	4544	4554	
4564	4571	4576	4639	4709	4771	4840	4912	5011	5121	5183	5246	5308	
5378	5468	5510	5512	5623	5803	5853	5899	5998	6095	6261	6299	6342	
6399	6468	6514	6563	6612	6654	6817	6881	6940	7001	7041	7073	7299	
7359	7453												

T\$LTNO= 000054
T\$NEST 000000

7521#													
1094#	1102#	1212#	1216#	1284#	1305#	1318#	1323#	2787#	2794#	2800#	2808#	2811#	
2820#	2823#	2832#	2837#	2844#	2849#	2856#	2863#	2874#	2879#	2889#	2894#	2902#	
2905#	2920#	2923#	2941#	2948#	2959#	2962#	2972#	2975#	2985#	2990#	2999#	3006#	
3024#	3029#	3036#	3372#	3379#	3393#	3478#	3482#	3496#	3506#	3509#	3523#	3526#	
3541#	3542#	3571#	3593#	3598#	3613#	3628#	3691#	3703#	3766#	3778#	3841#	3853#	
3916#	3928#	3991#	4003#	4066#	4079#	4113#	4125#	4157#	4213#	4270#	4292#	4539#	
4543#	4546#	4553#	4556#	4563#	4566#	4570#	4575#	4591#	4638#	4651#	4708#	4721#	
4770#	4788#	4839#	4853#	4911#	4928#	5010#	5024#	5120#	5135#	5182#	5197#	5245#	
5259#	5307#	5319#	5321#	5377#	5380#	5467#	5470#	5509#	5511#	5526#	5622#	5633#	
5802#	5819#	5852#	5867#	5898#	5914#	5997#	6011#	6094#	6110#	6260#	6278#	6298#	
6311#	6341#	6354#	6398#	6415#	6467#	6481#	6513#	6533#	6562#	6580#	6611#	6627#	
6653#	6668#	6816#	6839#	6880#	6898#	6939#	6958#	7000#	7018#	7040#	7055#	7072#	

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- USER SYMBOLS

T\$NSO = 000000
T\$NS1 = 000005

7085#	7298#	7320#	7357#	7448#	7451#								
1102#						2787#	2794	2800#	2808	2811#	2820	2823#	
1212#	1216	1284#	1305	1318#	1323	2874	2879#	2889	2894#	2902	2905#	2920	
2832	2837#	2844	2849#	2856	2863#	2975#	2985	2990#	2999	3006#	3024	3029#	
2923#	2941	2948#	2959	2962#	2972	3496	3506#	3509	3523#	3526	3541#	3542	
3036	3372#	3379	3393#	3478	3482#	3703#	3766	3778#	3841	3853#	3916	3928#	
3571#	3593	3598#	3613	3628#	3691	4113	4125#	4157	4213#	4270	4292#	4575	4591#
3991	4003#	4066	4079#	4113	4125#	4157	4213#	4270	4292#	4575	4591#	4638	
4651#	4708	4721#	4770	4788#	4839	4853#	4911	4928#	5010	5024#	5120	5135#	
5182	5197#	5245	5259#	5307	5319#	5511	5526#	5622	5633#	5802	5819#	5852	
5867#	5898	5914#	5997	6011#	6094	6110#	6260	6278#	6298	6311#	6341	6354#	
6398	6415#	6467	6481#	6513	6533#	6562	6580#	6611	6627#	6653	6668#	6816	
6839#	6880	6898#	6939	6958#	7000	7018#	7040	7055#	7072	7085#	7298	7320#	

T\$NS2 = 000002

4539#	4543	4546#	4553	4556#	4563	4566#	4570	5321#	5377	5380#	5467	5470#	
5509													

T\$PTNU= 000000
T\$SAVL= 177777
T\$SEGL= 177777
T\$SUBN= C00000

1094#													
1094#													
1094#													
1094#	3570#	3627#	3702#	3777#	3852#	3927#	4002#	4078#	4124#	4212#	4291#	4590#	
4650#	4720#	4787#	4852#	4927#	5023#	5134#	5196#	5258#	5318#	5320#	5379#	5469#	
5525#	5632#	5818#	5866#	5913#	6010#	6109#	6277#	6310#	6353#	6414#	6480#	6532#	
6579#	6626#	6667#	6838#	6897#	6957#	7017#	7054#	7084#					

T\$TAGL= 177777
T\$TAGN= 010120

1094#													
1094#	1212#	1284#	1318#	2787#	2800#	2811#	2823#	2837#	2849#	2863#	2879#	2894#	
2905#	2923#	2948#	2962#	2975#	2990#	3006#	3029#	3372#	3393#	3482#	3506#	3523#	
3541#	3571#	3598#	3628#	3703#	3778#	3853#	3928#	4003#	4079#	4125#	4213#	4292#	
4539#	4546#	4556#	4566#	4591#	4651#	4721#	4788#	4853#	4928#	5024#	5135#	5197#	
5259#	5319#	5321#	5380#	5470#	5526#	5633#	5819#	5867#	5914#	6011#	6110#	6278#	
6311#	6354#	6415#	6481#	6533#	6580#	6627#	6668#	6839#	6898#	6958#	7018#	7055#	

T\$TEMP- 000005

7085#	7320#	7448#											
1216#	1226#	1227#	1228#	1229#	1230#	1231#	1232#	1233#	1234#	1235#	1236#	1237#	
1238#	1239#	1240#	1241#	1242#	1243#	1244#	1245#	1246#	1247#	1248#	1249#	1250#	
1251#	1252#	1253#	1254#	1255#	1256#	1257#	1258#	1259#	1260#	1261#	1262#	1263#	
1264#	1265#	1266#	1267#	1268#	1269#	1270#	1305#	1323#	2794#	2808#	2820#	2832#	
2844#	2856#	2874#	2889#	2902#	2920#	2941#	2959#	2972#	2985#	2999#	3024#	3036#	
3375#	3376	3379#	3478#	3496#	3509#	3526#	3542#	3593#	3613#	3691#	3766#	3841#	
3916#	3991#	4066#	4086#	4087	4098#	4099	4113#	4132#	4133	4143#	4144	4157#	
4229#	4230	4242#	4243	4250#	4251	4259#	4260	4268#	4269	4270#	4303#	4304	
4311#	4312	4326#	4327	4350#	4351	4366#	4367	4388#	4389	4432#	4433	4468#	
4469	4502#	4503	4515#	4516	4531#	4532	4536#	4537	4543#	4553#	4563#	4570#	
4575#	4595#	4596	4609#	4610	4638#	4655#	4656	4665#	4666	4686#	4687	4700#	
4701	4708#	4730#	4731	4738#	4739	4746#	4747	4759#	4760	4770#	4793#	4794	
4814#	4815	4822#	4823	4832#	4833	4839#	4858#	4859	4885#	4886	4895#	4896	
4911#	4941#	4942	4954#	4955	4971#	4972	4983#	4984	4998#	4999	5010#	5030#	
5031	5044#	5045	5059#	5060	5073#	5074	5083#	5084	5094#	5095	5104#	5105	
5120#	5139#	5140	5153#	5154	5166#	5167	5176#	5177	5182#	5201#	5202	5215#	
5216	5228#	5229	5239#	5240	5245#	5263#	5264	5277#	5278	5290#	5291	5307#	
5332#	5333	5343#	5344	5354#	5355	5366#	5367	5377#	5390#	5391	5402#	5403	
5416#	5417	5425#	5426	5438#	5439	5449#	5450	5458#	5459	5467#	5478#	5479	
5489#	5490	5499#	5500	5509#	5511#	5529#	5530	5539#	5540	5547#	5548	5555#	
5556	5560#	5561	5571#	5572	5584#	5585	5590#	5591	5595#	5596	5600#	5601	
5606#	5607	5612#	5613	5622#	5637#	5638	5651#	5652	5664#	5665	5675#	5676	
5701#	5702	5714#	5715	5724#	5725	5737#	5738	5748#	5749	5759#	5760	5773#	
5774	5786#	5787	5796#	5797	5802#	5824#	5825	5834#	5835	5842#	5843	5852#	
5871#	5872	5882#	5883	5892#	5893	5898#	5918#	5919	5932#	5933	5942#	5943	

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- USER SYMBOLS

5952#	5953	5962#	5963	5972#	5973	5982#	5983	5997#	6015#	6016	6029#	6030
6039#	6040	6049#	6050	6059#	6060	6069#	6070	6079#	6080	6094#	6114#	6115
6128#	6129	6142#	6143	6155#	6156	6169#	6170	6182#	6183	6195#	6196	6209#
6210	6222#	6223	6235#	6236	6248#	6249	6260#	6288#	6289	6296#	6297	6298#
6315#	6316	6325#	6326	6336#	6337	6341#	6358#	6359	6372#	6373	6382#	6383
6393#	6394	6398#	6419#	6420	6430#	6431	6440#	6441	6450#	6451	6461#	6462
6467#	6485#	6486	6496#	6497	6508#	6509	6513#	6547#	6548	6556#	6557	6562#
6596#	6597	6605#	6606	6611#	6637#	6638	6647#	6648	6653#	6671#	6672	6682#
6683	6701#	6702	6714#	6715	6727#	6728	6740#	6741	6753#	6754	6769#	6770
6776#	6777	6816#	6876#	6877	6880#	6935#	6936	6939#	6996#	6997	7000#	7033#
7034	7040#	7065#	7066	7072#	7089#	7090	7099#	7100	7109#	7110	7121#	7122
7130#	7131	7138#	7139	7146#	7147	7151#	7152	7160#	7161	7165#	7166	7179#
7180	7200#	7201	7214#	7215	7236#	7237	7241#	7242	7257#	7258	7262#	7263
7271#	7272	7277#	7278	7289#	7290	7294#	7295	7298#	7322#	7327#	7331#	7335#
7340#	7345#	7350#	7357#	7451#								
1094#	3556	3569	3570#	3618	3625	3627#	3693	3700	3702#	3768	3775	3777#
3843	3850	3852#	3918	3925	3927#	3993	4000	4002#	4068	4077	4078#	4115
4123	4124#	4160	4211	4212#	4274	4288	4291#	4579	4589	4590#	4641	4649
4650#	4710	4719	4720#	4774	4786	4787#	4842	4851	4852#	4914	4925	4927#
5013	5022	5023#	5124	5133	5134#	5186	5195	5196#	5249	5257	5258#	5309
5317	5318#	5320	5379	5469	5515	5524	5525#	5625	5631	5632#	5807	5817
5818#	5856	5865	5866#	5902	5912	5913#	6000	6009	6010#	6097	6108	6109#
6265	6275	6277#	6301	6309	6310#	6344	6352	6353#	6404	6413	6414#	6470
6479	6480#	6519	6531	6532#	6567	6578	6579#	6614	6625	6626#	6656	6666
6667#	6823	6837	6838#	6882	6896	6897#	6942	6956	6957#	7003	7016	7017#
7043	7053	7054#	7074	7083	7084#	7521						
1094#	1638	1681	1728	1742	1818	1830	1859	1889	1922	1958	1989	2033
2046	2062	2069	2250	2295	2303	2330	2339	2347	2353	2435	2565	2576
2583	2745	2792	2795	2805	2809	2817	2821	2829	2833	2841	2845	2853
2857	2871	2875	2886	2890	2899	2903	2909	2917	2921	2927	2933	2938
2942	2956	2960	2969	2973	2982	2986	2996	3000	3012	3021	3025	3033
3037	3380	3410	3414	3418	3422	3441	3479	3492	3495	3497	3507	3510
3525	3527	3543	3575	3588	3591	3594	3601	3611	3652	3662	3679	3686
3692	3727	3737	3754	3761	3767	3802	3812	3829	3836	3842	3877	3887
3904	3911	3917	3952	3962	3979	3986	3992	4027	4037	4054	4061	4067
4086	4093	4098	4107	4114	4132	4139	4143	4152	4158	4225	4229	4236
4242	4246	4250	4259	4268	4271	4303	4311	4326	4346	4350	4366	4384
4388	4411	4419	4426	4432	4450	4456	4461	4468	4482	4484	4502	4510
4515	4527	4531	4536	4541	4548	4549	4558	4559	4568	4576	4595	4609
4630	4639	4655	4665	4681	4686	4695	4700	4709	4730	4738	4746	4755
4759	4765	4771	4793	4810	4814	4822	4832	4840	4858	4885	4895	4906
4912	4941	4954	4971	4983	4998	5011	5030	5044	5059	5073	5083	5094
5104	5121	5139	5153	5166	5176	5183	5201	5215	5228	5239	5246	5263
5277	5290	5301	5308	5321	5332	5343	5354	5366	5378	5380	5390	5402
5416	5425	5438	5449	5458	5468	5470	5478	5489	5499	5510	5512	5529
5539	5547	5555	5560	5571	5584	5590	5595	5600	5606	5612	5623	5637
5651	5664	5675	5688	5701	5714	5724	5737	5748	5759	5773	5786	5796
5803	5824	5834	5842	5853	5871	5882	5892	5899	5918	5932	5942	5952
5962	5972	5982	5992	5998	6015	6029	6039	6049	6059	6069	6079	6089
6095	6114	6128	6142	6155	6169	6182	6195	6209	6222	6235	6248	6261
6288	6296	6299	6315	6325	6336	6342	6358	6372	6382	6393	6399	6419
6430	6440	6450	6461	6468	6485	6496	6508	6514	6547	6556	6563	6596
6605	6612	6637	6647	6654	6671	6682	6701	6714	6727	6740	6753	6765
6769	6776	6817	6851	6876	6881	6910	6935	6940	6971	6996	7001	7033
7041	7065	7073	7089	7099	7109	7121	7130	7138	7146	7151	7160	7165
7179	7200	7214	7236	7241	7257	7262	7271	7277	7289	7294	7299	

T\$TEST= 000054

T\$TSTM= 177777

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

GPRMA	1#	1094#	7327	7331															
GPRMD	1#	1094#	7322	7335	7340	7345	7350												
GPRML	1#	1094#																	
HEADER	1#	1094#	1126																
INLOOP	1#	1094#																	
IOSETU	1#	1094#																	
IOSTAR	1#	1094#																	
KT11	1#	1094#																	
LASTAD	1#	1094#	7517																
MANUAL	1#	1094#																	
MEMORY	1#	1094#																	
MSBYTE	1#	1094#	1126#	1132	1133	1134													
MSCHEC	1#	1094#	3375#	4536#	4814#	6671#	6776#												
MSCNTO	1#	1094#	7322#	7327#	7331#	7335#	7340#	7345#	7350#										
MSCOUN	1#	1094#	2325#	2332#	2343#	2350#	2562#	2573#	2579#	2788#	2801#	2812#	2824#	2838#	2850#				
	2864#	2880#	2895#	2906#	2911#	2924#	2929#	2935#	2949#	2963#	2976#	2991#	3007#	3014#	3030#				
	3607#	3673#	3748#	3823#	3898#	3973#	4048#	4101#	4146#										
MSDATA	1#	1094#	1126#	1135	1137	1139	1141	1143	1145	1147	1149	1151	1153	1155	1157				
	1159	1161	1163	1165#	1167	1169	1172	1175	1177	1179	1181	1183	1185	1187	1189				
	1191	1193	1195	1197	1199	1201	1203	1205	1207	1209	1568#	1574#							
MSDECR	1#	1094#	1216#	1305#	1323#	2794#	2808#	2820#	2832#	2844#	2856#	2874#	2889#	2902#	2920#	2941#			
	2941#	2959#	2972#	2985#	2999#	3024#	3036#	3379#	3478#	3496#	3509#	3526#	3542#	3593#	3613#	3691#			
	3691#	3766#	3841#	3916#	3991#	4066#	4113#	4157#	4270#	4543#	4553#	4563#	4570#	4575#	4638#	4708#			
	4708#	4770#	4839#	4911#	5010#	5120#	5182#	5245#	5307#	5377#	5467#	5509#	5511#	5622#	5802#	5852#			
	5852#	5898#	5997#	6094#	6260#	6298#	6341#	6398#	6467#	6513#	6562#	6611#	6653#	6816#	6880#	6939#			
	6939#	7000#	7040#	7072#	7298#	7357#	7451#												
MSDEFA	1#	1094#	7322#	7327#	7331#	7335#	7340#	7345#	7350#										
MSENDE	1#	1094#	1305#	1323#	2794#	2808#	2820#	2832#	2844#	2856#	2874#	2889#	2902#	2920#	2941#				
	2959#	2972#	2985#	2999#	3024#	3036#	3379#	3478#	3496#	3509#	3526#	3542#	3593#	3613#	3691#				
	3766#	3841#	3916#	3991#	4066#	4113#	4157#	4270#	4543#	4553#	4563#	4570#	4575#	4638#	4708#				
	4770#	4839#	4911#	5010#	5120#	5182#	5245#	5307#	5377#	5467#	5509#	5511#	5622#	5802#	5852#				
	5898#	5997#	6094#	6260#	6298#	6341#	6398#	6467#	6513#	6562#	6611#	6653#	6816#	6880#	6939#				
	7000#	7040#	7072#	7298#	7357#	7451#													
MSERRI	1#	1094#	1638#	1681#	1728#	1742#	1818#	1830#	1859#	1889#	1922#	1958#	1989#	2033#	2046#				
	2062#	2069#	2250#	2295#	2435#	2745#	3601#	3652#	3662#	3686#	3727#	3737#	3761#	3802#	3812#				
	3836#	3877#	3887#	3911#	3952#	3962#	3986#	4027#	4037#	4061#	4093#	4139#	4225#	4236#	4246#				
	4346#	4384#	4510#	4527#	4549#	4559#	4630#	4681#	4695#	4755#	4765#	4810#	4906#	5301#	5688#				
	5992#	6089#	6765#																
MSESCA	1#	1094#	4086#	4087	4098#	4099	4132#	4133	4143#	4144	4229#	4230	4242#	4243	4250#				
	4251	4259#	4260	4268#	4269	4303#	4304	4311#	4312	4326#	4327	4350#	4351	4366#	4367				
	4388#	4389	4432#	4433	4468#	4469	4502#	4503	4515#	4516	4531#	4532	4595#	4596	4609#				
	4610	4655#	4656	4665#	4666	4686#	4687	4700#	4701	4730#	4731	4738#	4739	4746#	4747				
	4759#	4760	4793#	4794	4822#	4823	4832#	4833	4858#	4859	4885#	4886	4895#	4896	4941#				
	4942	4954#	4955	4971#	4972	4983#	4984	4998#	4999	5030#	5031	5044#	5045	5059#	5060				
	5073#	5074	5083#	5084	5094#	5095	5104#	5105	5139#	5140	5153#	5154	5166#	5167	5176#				
	5177	5201#	5202	5215#	5216	5228#	5229	5239#	5240	5263#	5264	5277#	5278	5290#	5291				
	5332#	5333	5343#	5344	5354#	5355	5366#	5367	5390#	5391	5402#	5403	5416#	5417	5425#				
	5426	5438#	5439	5449#	5450	5458#	5459	5478#	5479	5489#	5490	5499#	5500	5529#	5530				
	5539#	5540	5547#	5548	5555#	5556	5560#	5561	5571#	5572	5584#	5585	5590#	5591	5595#				
	5596	5600#	5601	5606#	5607	5612#	5613	5637#	5638	5651#	5652	5664#	5665	5675#	5676				
	5701#	5702	5714#	5715	5724#	5725	5737#	5738	5748#	5749	5759#	5760	5773#	5774	5786#				
	5787	5796#	5797	5824#	5825	5834#	5835	5842#	5843	5871#	5872	5882#	5883	5892#	5893				
	5918#	5919	5932#	5933	5942#	5943	5952#	5953	5962#	5963	5972#	5973	5982#	5983	6015#				
	6016	6029#	6030	6039#	6040	6049#	6050	6059#	6060	6069#	6070	6079#	6080	6114#	6115				
	6128#	6129	6142#	6143	6155#	6156	6169#	6170	6182#	6183	6195#	6196	6209#	6210	6222#				
	6223	6235#	6236	6248#	6249	6288#	6289	6296#	6297	6315#	6316	6325#	6326	6336#	6337				

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

	6358#	6359	6372#	6373	6382#	6383	6393#	6394	6419#	6420	6430#	6431	6440#	6441	6450#
	6451	6461#	6462	6485#	6486	6496#	6497	6508#	6509	6547#	6548	6556#	6557	6596#	6597
	6605#	6606	6637#	6638	6647#	6648	6682#	6683	6701#	6702	6714#	6715	6727#	6728	6740#
	6741	6753#	6754	6769#	6770	6876#	6877	6935#	6936	6996#	6997	7033#	7034	7065#	7066
	7089#	7090	7099#	7100	7109#	7110	7121#	7122	7130#	7131	7138#	7139	7146#	7147	7151#
	7152	7160#	7161	7165#	7166	7179#	7180	7200#	7201	7214#	7215	7236#	7237	7241#	7242
	7257#	7258	7262#	7263	7271#	7272	7277#	7278	7289#	7290	7294#	7295			
MSESCS	1#	1094#	4086#	4098#	4132#	4143#	4229#	4242#	4250#	4259#	4268#	4303#	4311#	4326#	4350#
	4366#	4388#	4432#	4468#	4502#	4515#	4531#	4595#	4609#	4655#	4665#	4686#	4700#	4730#	4738#
	4746#	4759#	4793#	4822#	4832#	4858#	4885#	4895#	4941#	4954#	4971#	4983#	4998#	5030#	5044#
	5059#	5073#	5083#	5094#	5104#	5139#	5153#	5166#	5176#	5201#	5215#	5228#	5239#	5263#	5277#
	5290#	5332#	5343#	5354#	5366#	5390#	5402#	5416#	5425#	5438#	5449#	5458#	5478#	5489#	5499#
	5529#	5539#	5547#	5555#	5560#	5571#	5584#	5590#	5595#	5600#	5606#	5612#	5637#	5651#	5664#
	5675#	5701#	5714#	5724#	5737#	5748#	5759#	5773#	5786#	5796#	5824#	5834#	5842#	5871#	5882#
	5892#	5918#	5932#	5942#	5952#	5962#	5972#	5982#	6015#	6029#	6039#	6049#	6059#	6069#	6079#
	6114#	6128#	6142#	6155#	6169#	6182#	6195#	6209#	6222#	6235#	6248#	6288#	6296#	6315#	6325#
	6336#	6358#	6372#	6382#	6393#	6419#	6430#	6440#	6450#	6461#	6485#	6496#	6508#	6547#	6556#
	6596#	6605#	6637#	6647#	6682#	6701#	6714#	6727#	6740#	6753#	6769#	6876#	6935#	6996#	7033#
	7065#	7089#	7099#	7109#	7121#	7130#	7138#	7146#	7151#	7160#	7165#	7179#	7200#	7214#	7236#
	7241#	7257#	7262#	7271#	7277#	7289#	7294#								
MSEXCP	1#	1094#	7322#	7327#	7331#	7335#	7340#	7345#	7350#						
MSEXIT	1#	1094#	3375#	4536#	4537	4814#	4815	6671#	6672	6776#	6777				
MSEXSE	1#	1094#	3375#	4536#	4814#	6671#	6776#								
MSEXTJ	1#	1094#	3375#	3376	4536#	4814#	6671#	6776#							
MSGEN	1#	1094#	1102#	1126#	1135#	1137#	1139#	1141#	1143#	1145#	1147#	1149#	1151#	1153#	1155#
	1157#	1159#	1161#	1163#	1165#	1167#	1169#	1172#	1175#	1177#	1179#	1181#	1183#	1185#	1187#
	1189#	1191#	1193#	1195#	1197#	1199#	1201#	1203#	1205#	1207#	1209#	1212#	1225#	1285#	1286#
	1305#	1319#	1320#	1323#	1568#	1574#	2787#	2794#	2800#	2808#	2811#	2820#	2823#	2832#	2837#
	2844#	2849#	2856#	2863#	2874#	2879#	2889#	2894#	2902#	2905#	2920#	2923#	2941#	2948#	2959#
	2962#	2972#	2975#	2985#	2990#	2999#	3006#	3024#	3029#	3036#	3372#	3379#	3393#	3478#	3482#
	3496#	3506#	3509#	3523#	3526#	3541#	3542#	3570#	3593#	3598#	3613#	3627#	3691#	3702#	3766#
	3777#	3841#	3852#	3916#	3927#	3991#	4002#	4066#	4078#	4113#	4124#	4157#	4212#	4270#	4291#
	4539#	4543#	4546#	4553#	4556#	4563#	4566#	4570#	4575#	4590#	4638#	4650#	4708#	4720#	4770#
	4787#	4839#	4852#	4911#	4927#	5010#	5023#	5120#	5134#	5182#	5196#	5245#	5258#	5307#	5318#
	5320#	5377#	5379#	5467#	5469#	5509#	5511#	5525#	5622#	5632#	5802#	5818#	5852#	5866#	5898#
	5913#	5997#	6010#	6094#	6109#	6260#	6277#	6298#	6310#	6341#	6353#	6398#	6414#	6467#	6480#
	6513#	6532#	6562#	6579#	6611#	6626#	6653#	6667#	6816#	6838#	6880#	6897#	6939#	6957#	7000#
	7017#	7040#	7054#	7072#	7084#	7298#	7321#	7358#	7449#	7452#	7520#				
MSGENB	1#	1094#													
MSGETS	1#	1094#	1216#	1305#	1323#	2794#	2808#	2820#	2832#	2844#	2856#	2874#	2889#	2902#	2920#
	2941#	2959#	2972#	2985#	2999#	3024#	3036#	3379#	3478#	3496#	3509#	3526#	3542#	3593#	3613#
	3691#	3766#	3841#	3916#	3991#	4066#	4113#	4157#	4270#	4543#	4553#	4563#	4570#	4575#	4638#
	4708#	4770#	4839#	4911#	5010#	5120#	5182#	5245#	5307#	5377#	5467#	5509#	5511#	5622#	5802#
	5852#	5898#	5997#	6094#	6260#	6298#	6341#	6398#	6467#	6513#	6562#	6611#	6653#	6816#	6880#
	6939#	7000#	7040#	7072#	7298#	7357#	7451#								
MSGETT	1#	1094#	3375#	4086#	4098#	4132#	4143#	4229#	4242#	4250#	4259#	4268#	4303#	4311#	4326#
	4350#	4366#	4388#	4432#	4468#	4502#	4515#	4531#	4536#	4595#	4609#	4655#	4665#	4686#	4700#
	4730#	4738#	4746#	4759#	4793#	4814#	4822#	4832#	4858#	4885#	4895#	4941#	4954#	4971#	4983#
	4998#	5030#	5044#	5059#	5073#	5083#	5094#	5104#	5139#	5153#	5166#	5176#	5201#	5215#	5228#
	5239#	5263#	5277#	5290#	5332#	5343#	5354#	5366#	5390#	5402#	5416#	5425#	5438#	5449#	5458#
	5478#	5489#	5499#	5529#	5539#	5547#	5555#	5560#	5571#	5584#	5590#	5595#	5600#	5606#	5612#
	5637#	5651#	5664#	5675#	5701#	5714#	5724#	5737#	5748#	5759#	5773#	5786#	5796#	5824#	5834#
	5842#	5871#	5882#	5892#	5913	5932#	5942#	5952#	5962#	5972#	5982#	6015#	6029#	6039#	6049#
	6059#	6069#	6079#	6114#	6128	6142#	6155#	6169#	6182#	6195#	6209#	6222#	6235#	6248#	6288#
	6296#	6315#	6325#	6336#	6358#	6372#	6382#	6393#	6419#	6430#	6440#	6450#	6461#	6485#	6496#
	6508#	6547#	6556#	6596#	6605#	6637#	6647#	6671#	6682#	6701#	6714#	6727#	6740#	6753#	6769#

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

	6776#	6876#	6935#	6996#	7033#	7065#	7089#	7099#	7109#	7121#	7130#	7138#	7146#	7151#	7160#
	7165#	7179#	7200#	7114#	7236#	7241#	7257#	7262#	7271#	7277#	7289#	7294#			
MSGNGB	1#	1094#	1102#	1126#	1135#	1137#	1139#	1141#	1143#	1145#	1147#	1149#	1151#	1153#	1155#
	1157#	1159#	1161#	1163#	1165#	1167#	1169#	1172#	1175#	1177#	1179#	1181#	1183#	1185#	1187#
	1189#	1191#	1193#	1195#	1197#	1199#	1201#	1203#	1205#	1207#	1209#	1212#	1224#	1225#	1284#
	1285#	1286#	1318#	1319#	1320#	1568#	1574#	2787#	2800#	2811#	2823#	2837#	2849#	2863#	2879#
	2894#	2905#	2923#	2948#	2962#	2975#	2990#	3006#	3029#	3372#	3393#	3482#	3506#	3523#	3541#
	3598#	4539#	4546#	4556#	4566#	7320#	7321#	7448#	7449#	7517#	7520#				
MSGNIN	1#	1094#	1126#	1127#	1128#	1129#	1130#	1131#	1132#	1133#	1134#	1135#	1136#	1137#	1138#
	1139#	1140#	1141#	1142#	1143#	1144#	1145#	1146#	1147#	1148#	1149#	1150#	1151#	1152#	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#	1209#	1210#	1224#	1226#	1227#
	1228#	1229#	1230#	1231#	1232#	1233#	1234#	1235#	1236#	1237#	1238#	1239#	1240#	1241#	1242#
	1243#	1244#	1245#	1246#	1247#	1248#	1249#	1250#	1251#	1252#	1253#	1254#	1255#	1256#	1257#
	1258#	1259#	1260#	1261#	1262#	1263#	1264#	1265#	1266#	1267#	1268#	1269#	1284#	1318#	1568#
	1569#	1572#	1574#	1575#	1581#	1638#	1639#	1640#	1641#	1681#	1682#	1683#	1684#	1728#	1729#
	1730#	1731#	1742#	1743#	1744#	1745#	1818#	1819#	1820#	1821#	1830#	1831#	1832#	1833#	1859#
	1860#	1861#	1862#	1889#	1890#	1891#	1892#	1922#	1923#	1924#	1925#	1958#	1959#	1960#	1961#
	1989#	1990#	1991#	1992#	2033#	2034#	2035#	2036#	2046#	2047#	2048#	2049#	2062#	2063#	2064#
	2065#	2069#	2070#	2071#	2072#	2250#	2251#	2252#	2253#	2295#	2296#	2297#	2298#	2303#	2325#
	2326#	2327#	2328#	2329#	2330#	2331#	2332#	2333#	2334#	2335#	2336#	2337#	2338#	2339#	2340#
	2343#	2344#	2345#	2346#	2347#	2348#	2350#	2351#	2352#	2353#	2354#	2435#	2436#	2437#	2438#
	2562#	2563#	2564#	2565#	2566#	2573#	2574#	2575#	2576#	2577#	2579#	2580#	2581#	2582#	2583#
	2584#	2745#	2746#	2747#	2748#	2788#	2789#	2790#	2791#	2792#	2793#	2795#	2801#	2802#	2803#
	2804#	2805#	2806#	2809#	2812#	2813#	2814#	2815#	2816#	2817#	2818#	2821#	2824#	2825#	2826#
	2827#	2828#	2829#	2830#	2833#	2838#	2839#	2840#	2841#	2842#	2845#	2850#	2851#	2852#	2853#
	2854#	2857#	2864#	2865#	2866#	2867#	2868#	2869#	2870#	2871#	2872#	2875#	2880#	2881#	2882#
	2883#	2884#	2885#	2886#	2887#	2890#	2895#	2896#	2897#	2898#	2899#	2900#	2903#	2906#	2907#
	2908#	2909#	2910#	2911#	2912#	2913#	2914#	2915#	2916#	2917#	2918#	2921#	2924#	2925#	2926#
	2927#	2928#	2929#	2930#	2931#	2932#	2933#	2934#	2935#	2936#	2937#	2938#	2939#	2942#	2949#
	2950#	2951#	2952#	2953#	2954#	2955#	2956#	2957#	2960#	2963#	2964#	2965#	2966#	2967#	2968#
	2969#	2970#	2973#	2976#	2977#	2978#	2979#	2980#	2981#	2982#	2983#	2986#	2991#	2992#	2993#
	2994#	2995#	2996#	2997#	3000#	3007#	3008#	3009#	3010#	3011#	3012#	3013#	3014#	3015#	3016#
	3017#	3018#	3019#	3020#	3021#	3022#	3025#	3030#	3031#	3032#	3033#	3034#	3037#	3375#	3376#
	3380#	3409#	3410#	3411#	3413#	3414#	3415#	3417#	3418#	3419#	3421#	3422#	3423#	3440#	3441#
	3442#	3443#	3479#	3491#	3492#	3495#	3497#	3507#	3510#	3525#	3527#	3543#	3571#	3572#	3573#
	3574#	3575#	3576#	3588#	3590#	3591#	3594#	3601#	3602#	3603#	3604#	3607#	3608#	3609#	3610#
	3611#	3612#	3613#	3614#	3652#	3653#	3654#	3655#	3662#	3663#	3664#	3665#	3673#	3674#	3675#
	3676#	3677#	3678#	3679#	3680#	3686#	3687#	3688#	3689#	3692#	3727#	3728#	3729#	3730#	3737#
	3738#	3739#	3740#	3748#	3749#	3750#	3751#	3752#	3753#	3754#	3755#	3761#	3762#	3763#	3764#
	3767#	3802#	3803#	3804#	3805#	3812#	3813#	3814#	3815#	3823#	3824#	3825#	3826#	3827#	3828#
	3829#	3830#	3836#	3837#	3838#	3839#	3842#	3877#	3878#	3879#	3880#	3887#	3888#	3889#	3890#
	3898#	3899#	3900#	3901#	3902#	3903#	3904#	3905#	3911#	3912#	3913#	3914#	3917#	3952#	3953#
	3954#	3955#	3962#	3963#	3964#	3965#	3973#	3974#	3975#	3976#	3977#	3978#	3979#	3980#	3986#
	3987#	3988#	3989#	3992#	4027#	4028#	4029#	4030#	4037#	4038#	4039#	4040#	4048#	4049#	4050#
	4051#	4052#	4053#	4054#	4055#	4061#	4062#	4063#	4064#	4067#	4086#	4087#	4093#	4094#	4095#
	4096#	4098#	4099#	4101#	4102#	4103#	4104#	4105#	4106#	4107#	4108#	4114#	4132#	4133#	4139#
	4140#	4141#	4142#	4143#	4144#	4146#	4147#	4148#	4149#	4150#	4151#	4152#	4153#	4158#	4225#
	4226#	4227#	4228#	4229#	4230#	4236#	4237#	4238#	4239#	4242#	4243#	4246#	4247#	4248#	4249#
	4250#	4251#	4259#	4260#	4268#	4269#	4271#	4303#	4304#	4311#	4312#	4326#	4327#	4346#	4347#
	4348#	4349#	4350#	4351#	4366#	4367#	4384#	4385#	4386#	4387#	4388#	4389#	4407#	4408#	4409#
	4410#	4411#	4412#	4415#	4416#	4417#	4418#	4419#	4420#	4425#	4426#	4432#	4433#	4446#	4447#
	4448#	4449#	4450#	4451#	4452#	4453#	4454#	4455#	4456#	4457#	4460#	4461#	4468#	4469#	4481#
	4482#	4483#	4484#	4502#	4503#	4510#	4511#	4512#	4513#	4515#	4516#	4527#	4528#	4529#	4530#

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

4531#	4532#	4536#	4537#	4540#	4541#	4543#	4544	4547#	4548#	4549#	4550#	4551#	4552#	4553#		
4554	4557#	4558#	4559#	4560#	4561#	4562#	4563#	4564	4567#	4568#	4570#	4571	4576#	4595#		
4596#	4609#	4610#	4630#	4631#	4632#	4633#	4639#	4655#	4656#	4665#	4666#	4681#	4682#	4683#		
4684#	4686#	4687#	4695#	4696#	4697#	4698#	4700#	4701#	4709#	4730#	4731#	4738#	4739#	4746#		
4747#	4755#	4756#	4757#	4758#	4759#	4760#	4765#	4766#	4767#	4768#	4771#	4793#	4794#	4810#		
4811#	4812#	4813#	4814#	4815#	4822#	4823#	4832#	4833#	4840#	4858#	4859#	4885#	4886#	4895#		
4896#	4906#	4907#	4908#	4909#	4912#	4941#	4942#	4954#	4955#	4971#	4972#	4983#	4984#	4998#		
4999#	5011#	5030#	5031#	5044#	5045#	5059#	5060#	5073#	5074#	5083#	5084#	5094#	5095#	5104#		
5105#	5121#	5139#	5140#	5153#	5154#	5166#	5167#	5176#	5177#	5183#	5201#	5202#	5215#	5216#		
5228#	5229#	5239#	5240#	5246#	5263#	5264#	5277#	5278#	5290#	5291#	5301#	5302#	5303#	5304#		
5308#	5321#	5332#	5333#	5343#	5344#	5354#	5355#	5366#	5367#	5378#	5380#	5390#	5391#	5402#		
5403#	5416#	5417#	5425#	5426#	5438#	5439#	5449#	5450#	5458#	5459#	5468#	5470#	5478#	5479#		
5489#	5490#	5499#	5500#	5510#	5512#	5529#	5530#	5539#	5540#	5547#	5548#	5555#	5556#	5560#		
5561#	5571#	5572#	5584#	5585#	5590#	5591#	5595#	5596#	5600#	5601#	5606#	5607#	5612#	5613#		
5623#	5637#	5638#	5651#	5652#	5664#	5665#	5675#	5676#	5688#	5689#	5690#	5691#	5701#	5702#		
5714#	5715#	5724#	5725#	5737#	5738#	5748#	5749#	5759#	5760#	5773#	5774#	5786#	5787#	5796#		
5797#	5803#	5824#	5825#	5834#	5835#	5842#	5843#	5853#	5871#	5872#	5882#	5883#	5892#	5893#		
5899#	5918#	5919#	5932#	5933#	5942#	5943#	5952#	5953#	5962#	5963#	5972#	5973#	5982#	5983#		
5992#	5993#	5994#	5995#	5998#	6015#	6016#	6029#	6030#	6039#	6040#	6049#	6050#	6059#	6060#		
6069#	6070#	6079#	6080#	6089#	6090#	6091#	6092#	6095#	6114#	6115#	6128#	6129#	6142#	6143#		
6155#	6156#	6169#	6170#	6182#	6183#	6195#	6196#	6209#	6210#	6222#	6223#	6235#	6236#	6248#		
6249#	6261#	6288#	6289#	6296#	6297#	6299#	6315#	6316#	6325#	6326#	6336#	6337#	6342#	6358#		
6359#	6372#	6373#	6382#	6383#	6393#	6394#	6399#	6419#	6420#	6430#	6431#	6440#	6441#	6450#		
6451#	6461#	6462#	6468#	6485#	6486#	6496#	6497#	6508#	6509#	6514#	6547#	6548#	6556#	6557#		
6563#	6596#	6597#	6605#	6606#	6612#	6637#	6638#	6647#	6648#	6654#	6671#	6672#	6682#	6683#		
6701#	6702#	6714#	6715#	6727#	6728#	6740#	6741#	6753#	6754#	6765#	6766#	6767#	6768#	6769#		
6770#	6776#	6777#	6817#	6850#	6851#	6876#	6877#	6881#	6909#	6910#	6935#	6936#	6940#	6970#		
6971#	6996#	6997#	7001#	7033#	7034#	7041#	7065#	7066#	7073#	7089#	7090#	7099#	7100#	7109#		
7110#	7121#	7122#	7130#	7131#	7138#	7139#	7146#	7147#	7151#	7152#	7160#	7161#	7165#	7166#		
7179#	7180#	7200#	7201#	7214#	7215#	7236#	7237#	7241#	7242#	7257#	7258#	7262#	7263#	7271#		
7272#	7277#	7278#	7289#	7290#	7294#	7295#	7299#	7320#	7322#	7323	7324	7325	7326	7327#		
7328	7329	7330	7331#	7332	7333	7334	7335#	7336	7337	7338	7339	7340#	7341	7342		
7343	7344	7345#	7346	7347	7348	7349	7350#	7351	7352	7353	7354	7357#	7448#	7451#		
7517#	7518#	7519#														
MSGNLS	1#	1094#														
MSGNSU	1#	1094#														
MSGNTA	1#	1094#	5320#	5379#	5469#	2808#	2820#	2832#	2844#	2856#	2874#	2889#	2902#	2920#	2941#	
	2959#	2972#	1305#	1323#	2794#	3024#	3036#	3379#	3478#	3496#	3509#	3526#	3542#	3593#	3613#	3691#
	3766#	3841#	3910#	3991#	4066#	4113#	4157#	4270#	4543#	4553#	4563#	4570#	4575#	4638#	4708#	
	4770#	4839#	4911#	5010#	5120#	5182#	5245#	5307#	5377#	5467#	5509#	5511#	5622#	5802#	5852#	
	5898#	5997#	6094#	6260#	6298#	6341#	6398#	6467#	6513#	6562#	6611#	6653#	6816#	6880#	6939#	
	7000#	7040#	7072#	7298#	7357#	7358	7451#	7452								
MSGNTE	1#	1094#	3570#	3627#	3702#	3777#	3852#	3927#	4002#	4078#	4124#	4212#	4291#	4590#	4650#	
	4720#	4787#	4852#	4927#	5023#	5134#	5196#	5258#	5318#	5525#	5632#	5818#	5866#	5913#	6010#	
	6109#	6277#	6310#	6353#	6414#	6480#	6532#	6579#	6626#	6667#	6838#	6897#	6957#	7017#	7054#	
	7084#															
M\$HAPT	1#	1094#	1126#													
M\$HNAP	1#	1094#	1126#	1165												
M\$INCR	1#	1094#	1102#	1212#	1284#	1318#	1638#	1681#	1728#	1742#	1818#	1830#	1859#	1889#	1922#	
	1958#	1989#	2033#	2046#	2062#	2069#	2250#	2295#	2303#	2330#	2339#	2347#	2353#	2435#	2565#	
	2576#	2583#	2745#	2787#	2792#	2795#	2800#	2805#	2809#	2811#	2817#	2821#	2823#	2829#	2833#	
	2837#	2841#	2845#	2849#	2853#	2857#	2863#	2871#	2875#	2879#	2886#	2890#	2894#	2899#	2903#	
	2905#	2909#	2917#	2921#	2923#	2927#	2933#	2938#	2942#	2948#	2956#	2960#	2962#	2969#	2973#	
	2975#	2982#	2986#	2990#	2996#	3000#	3006#	3012#	3021#	3025#	3029#	3033#	3037#	3372#	3380#	
	3393#	3410#	3414#	3418#	3422#	3441#	3479#	3482#	3492#	3495#	3497#	3506#	3507#	3510#	3523#	
	3525#	3527#	3541#	3543#	3570#	3571#	3575#	3588#	3591#	3594#	3598#	3601#	3611#	3627#	3628#	

CZMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

3652#	3662#	3679#	3686#	3692#	3702#	3703#	3727#	3757#	3754#	3761#	3767#	3777#	3778#	3802#	
3812#	3829#	3836#	3842#	3852#	3853#	3877#	3887#	3904#	3911#	3917#	3927#	3928#	3952#	3962#	
3979#	3986#	3992#	4002#	4003#	4027#	4037#	4054#	4061#	4067#	4078#	4079#	4086#	4093#	4098#	
4107#	4114#	4124#	4125#	4132#	4139#	4143#	4152#	4158#	4212#	4213#	4225#	4229#	4236#	4242#	
4246#	4250#	4259#	4268#	4271#	4291#	4292#	4303#	4311#	4326#	4346#	4350#	4366#	4384#	4388#	
4411#	4419#	4426#	4432#	4450#	4456#	4461#	4468#	4482#	4484#	4502#	4510#	4515#	4527#	4531#	
4536#	4539#	4541#	4546#	4548#	4549#	4556#	4558#	4559#	4566#	4568#	4576#	4590#	4591#	4595#	
4609#	4630#	4639#	4650#	4651#	4655#	4665#	4681#	4686#	4695#	4700#	4709#	4720#	4721#	4730#	
4738#	4746#	4755#	4759#	4765#	4771#	4787#	4788#	4793#	4810#	4814#	4822#	4832#	4840#	4852#	
4853#	4858#	4885#	4895#	4906#	4912#	4927#	4928#	4941#	4954#	4971#	4983#	4998#	5011#	5023#	
5024#	5030#	5044#	5059#	5075#	5083#	5094#	5104#	5121#	5134#	5135#	5139#	5153#	5166#	5176#	
5183#	5196#	5197#	5201#	5215#	5228#	5239#	5246#	5258#	5259#	5263#	5277#	5290#	5301#	5308#	
5318#	5319#	5320#	5321#	5332#	5343#	5354#	5366#	5378#	5379#	5380#	5390#	5402#	5416#	5425#	
5438#	5449#	5458#	5468#	5469#	5470#	5478#	5489#	5499#	5510#	5512#	5525#	5526#	5529#	5539#	
5547#	5555#	5560#	5571#	5584#	5590#	5595#	5600#	5606#	5612#	5623#	5632#	5633#	5637#	5651#	
5664#	5675#	5688#	5701#	5714#	5724#	5737#	5748#	5759#	5773#	5786#	5796#	5803#	5818#	5819#	
5824#	5834#	5842#	5853#	5866#	5867#	5871#	5882#	5892#	5899#	5913#	5914#	5918#	5932#	5942#	
5952#	5962#	5972#	5982#	5992#	5998#	6010#	6011#	6015#	6029#	6039#	6049#	6059#	6069#	6079#	
6089#	6095#	6109#	6110#	6114#	6128#	6142#	6155#	6169#	6182#	6195#	6209#	6222#	6235#	6248#	
6261#	6277#	6278#	6288#	6296#	6299#	6310#	6311#	6315#	6325#	6336#	6342#	6353#	6354#	6358#	
6372#	6382#	6393#	6399#	6414#	6415#	6419#	6430#	6440#	6450#	6461#	6468#	6480#	6481#	6485#	
6496#	6508#	6514#	6532#	6533#	6547#	6556#	6563#	6579#	6580#	6596#	6605#	6612#	6626#	6627#	
6637#	6647#	6654#	6667#	6668#	6671#	6682#	6701#	6714#	6727#	6740#	6753#	6765#	6769#	6776#	
6817#	6838#	6839#	6851#	6876#	6881#	6897#	6898#	6910#	6935#	6940#	6957#	6958#	6971#	6996#	
7001#	7017#	7018#	7033#	7041#	7054#	7055#	7065#	7073#	7084#	7085#	7089#	7099#	7109#	7121#	
7130#	7138#	7146#	7151#	7160#	7165#	7179#	7200#	7214#	7236#	7241#	7257#	7262#	7271#	7277#	
7289#	7294#	7299#	7320#	7448#											
MSIOSE	1#	1094#													
MSLDRO	1#	1094#	3409#	3413#	3417#	3421#	3440#	3491#	3590#	4425#	4460#	4481#	4483#	4540#	4547#
	4557#	4567#	6850#	6909#	6970#										
MSMASK	1#	1094#													
MSMCHI	1#	1094#													
MSMCLO	1#	1094#													
MSMSK1	1#	1094#													
MSPOP	1#	1094#	1216#	1305#	1323#	2794#	2808#	2820#	2832#	2844#	2856#	2874#	2889#	2902#	2920#
	2941#	2959#	2972#	2985#	2999#	3024#	3036#	3379#	3478#	3496#	3509#	3526#	3542#	3593#	3613#
	3691#	3766#	3841#	3916#	3991#	4066#	4113#	4157#	4270#	4543#	4553#	4563#	4570#	4575#	4638#
	4708#	4770#	4839#	4911#	5010#	5120#	5182#	5245#	5307#	5377#	5467#	5509#	5511#	5622#	5802#
	5852#	5898#	5997#	6094#	6260#	6298#	6341#	6398#	6467#	6513#	6562#	6611#	6653#	6816#	6880#
	6939#	7000#	7040#	7072#	7298#	7357#	7451#								
MSPRIN	1#	1094#	2325#	2332#	2343#	2350#	2562#	2573#	2579#	2788#	2801#	2812#	2824#	2838#	2850#
	2864#	2880#	2895#	2906#	2911#	2924#	2929#	2935#	2949#	2963#	2976#	2991#	3007#	3014#	3030#
	3607#	3673#	3748#	3823#	3898#	3973#	4048#	4101#	4146#						
MSPUSH	1#	1094#	1102#	1212#	1284#	1318#	2787#	2800#	2811#	2823#	2837#	2849#	2863#	2879#	2894#
	2905#	2923#	2948#	2962#	2975#	2990#	3006#	3029#	3372#	3393#	3482#	3506#	3523#	3541#	3570#
	3571#	3598#	3627#	3628#	3702#	3703#	3777#	3778#	3852#	3853#	3927#	3928#	4002#	4003#	4078#
	4079#	4124#	4125#	4212#	4213#	4291#	4292#	4539#	4546#	4556#	4566#	4590#	4591#	4650#	4651#
	4720#	4721#	4787#	4788#	4852#	4853#	4927#	4928#	5023#	5024#	5134#	5135#	5196#	5197#	5258#
	5259#	5318#	5319#	5320#	5321#	5379#	5380#	5469#	5470#	5525#	5526#	5632#	5633#	5818#	5819#
	5866#	5867#	5913#	5914#	6010#	6011#	6109#	6110#	6277#	6278#	6310#	6311#	6353#	6354#	6414#
	6415#	6480#	6481#	6532#	6533#	6579#	6580#	6626#	6627#	6667#	6668#	6838#	6839#	6897#	6898#
	6957#	6958#	7017#	7018#	7054#	7055#	7084#	7085#	7320#	7448#					
MSPUT	1#	1094#	2325#	2332#	2343#	2350#	2562#	2573#	2579#	2788#	2801#	2812#	2824#	2838#	2850#
	2864#	2880#	2895#	2906#	2911#	2924#	2929#	2935#	2949#	2963#	2976#	2991#	3007#	3014#	3030#
	3571#	3607#	3673#	3748#	3823#	3898#	3973#	4048#	4101#	4146#	4407#	4415#	4446#	4452#	
MSPUT1	1#	1094#	2325#	2326#	2327#	2328#	2332#	2333#	2334#	2335#	2336#	2337#	2343#	2344#	2345#

CDMTC.P11

25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

	2350#	2351	2562#	2563	2573#	2574	2579#	2580	2581	2788#	2789	2790	2801#	2802	2803
	2812#	2813	2814	2815	2824#	2825	2826	2827	2838#	2839	2850#	2851	2864#	2865	2866
	2867	2868	2869	2880#	2881	2882	2883	2884	2895#	2896	2897	2906#	2907	2911#	2912
	2913	2914	2915	2924#	2925	2929#	2930	2931	2935#	2936	2949#	2950	2951	2952	2953
	2954	2963#	2964	2965	2966	2967	2976#	2977	2978	2979	2980	2991#	2992	2993	2994
	3007#	3008	3009	3010	3014#	3015	3016	3017	3018	3019	3030#	3031	3571#	3572	3573
	3574	3607#	3608	3609	3673#	3674	3675	3676	3677	3748#	3749	3750	3751	3752	3823#
	3824	3825	3826	3827	3898#	3899	3900	3901	3902	3973#	3974	3975	3976	3977	4048#
	4049	4050	4051	4052	4101#	4102	4103	4104	4105	4146#	4147	4148	4149	4150	4407#
	4408	4409	4410	4415#	4416	4417	4418	4446#	4447	4448	4449	4452#	4453	4454	4455
M\$RADI	1#	1094#	7322#	7327#	7331#	7335#	7340#	7345#	7350#						
M\$RBRO	1#	1094#													
M\$RNRO	1#	1094#	3440#	3442											
M\$SETS	1#	1094#	1102#	1212#	1284#	1318#	2787#	2800#	2811#	2823#	2837#	2849#	2863#	2879#	2894#
	2905#	2923#	2948#	2962#	2975#	2990#	3006#	3029#	3372#	3393#	3482#	3506#	3523#	3541#	3571#
	3598#	3628#	3703#	3778#	3853#	3928#	4003#	4079#	4125#	4213#	4292#	4539#	4546#	4556#	4566#
	4591#	4651#	4721#	4788#	4853#	4928#	5024#	5135#	5197#	5259#	5319#	5321#	5380#	5470#	5526#
	5633#	5819#	5867#	5914#	6011#	6110#	6278#	6311#	6354#	6415#	6481#	6533#	6580#	6627#	6668#
	6839#	6898#	6958#	7018#	7055#	7085#	7320#	7448#							
M\$STAR	1#	1094#													
M\$SVC	1#	1094#	1638	1681	1728	1742	1818	1830	1859	1889	1922	1958	1989	2033	2046
	2062	2069	2250	2295	2303#	2325#	2330	2332#	2339	2343#	2347	2350#	2353	2435	2562#
	2565	2573#	2576	2579#	2583	2745	2788#	2792	2794#	2795	2801#	2805	2808#	2809	2812#
	2817	2820#	2821	2824#	2829	2832#	2833	2838#	2841	2844#	2845	2850#	2853	2856#	2857
	2864#	2871	2874#	2875	2880#	2886	2889#	2890	2895#	2899	2902#	2903	2906#	2909	2911#
	2917	2920#	2921	2924#	2927	2929#	2933	2935#	2938	2941#	2942	2949#	2956	2959#	2960
	2963#	2969	2972#	2973	2976#	2982	2985#	2986	2991#	2996	2999#	3000	3007#	3012	3014#
	3021	3024#	3025	3030#	3033	3036#	3037	3375#	3379#	3380	3409#	3410	3413#	3414	3417#
	3418	3421#	3422	3440#	3441	3478#	3479	3491#	3492	3495#	3496#	3497	3507#	3509#	3510
	3525#	3526#	3527	3542#	3543	3571#	3575	3588#	3590#	3591	3593#	3594	3601	3607#	3611
	3652	3662	3673#	3679	3686	3691#	3692	3727	3737	3748#	3754	3761	3766#	3767	3802
	3812	3823#	3829	3836	3841#	3842	3877	3887	3898#	3904	3911	3916#	3917	3952	3962
	3973#	3979	3986	3991#	3992	4027	4037	4048#	4054	4061	4066#	4067	4086#	4093	4098#
	4101#	4107	4113#	4114	4132#	4139	4143#	4146#	4152	4157#	4158	4225	4229#	4236	4242#
	4246	4250#	4259#	4268#	4270#	4271	4303#	4311#	4326#	4346	4350#	4366#	4384	4388#	4407#
	4411	4415#	4419	4425#	4426	4432#	4446#	4450	4452#	4456	4460#	4461	4468#	4481#	4482
	4483#	4484	4502#	4510	4515#	4527	4531#	4536#	4540#	4541	4547#	4548	4549	4557#	4558
	4559	4567#	4568	4575#	4576	4595#	4609#	4630	4638#	4639	4655#	4665#	4681	4686#	4695
	4700#	4708#	4709	4730#	4738#	4746#	4755	4759#	4765	4770#	4771	4793#	4810	4814#	4822#
	4832#	4839#	4840	4858#	4885#	4895#	4906	4911#	4912	4941#	4954#	4971#	4983#	4998#	5010#
	5011	5030#	5044#	5059#	5073#	5083#	5094#	5104#	5120#	5121	5139#	5153#	5166#	5176#	5182#
	5183	5201#	5215#	5228#	5239#	5245#	5246	5263#	5277#	5290#	5301	5307#	5308	5320#	5321
	5332#	5343#	5354#	5366#	5377#	5378	5379#	5380	5390#	5402#	5416#	5425#	5438#	5449#	5458#
	5467#	5468	5469#	5470	5478#	5489#	5499#	5509#	5510	5511#	5512	5529#	5539#	5547#	5555#
	5560#	5571#	5584#	5590#	5595#	5600#	5606#	5612#	5622#	5623	5637#	5651#	5664#	5675#	5688
	5701#	5714#	5724#	5737#	5748#	5759#	5773#	5786#	5796#	5802#	5803	5824#	5834#	5842#	5852#
	5853	5871#	5882#	5892#	5898#	5899	5918#	5932#	5942#	5952#	5962#	5972#	5982#	5992	5997#
	5998	6015#	6029#	6039#	6049#	6059#	6069#	6079#	6089	6094#	6095	6114#	6128#	6142#	6155#
	6169#	6182#	6195#	6209#	6222#	6235#	6248#	6260#	6261	6288#	6296#	6298#	6299	6315#	6325#
	6336#	6341#	6342	6358#	6372#	6382#	6393#	6398#	6399	6419#	6430#	6440#	6450#	6461#	6467#
	6468	6485#	6496#	6508#	6513#	6514	6547#	6556#	6562#	6563	6596#	6605#	6611#	6612	6637#
	6647#	6653#	6654	6671#	6682#	6701#	6714#	6727#	6740#	6753#	6765	6769#	6776#	6816#	6817
	6850#	6851	6876#	6880#	6881	6909#	6910	6935#	6939#	6940	6970#	6971	6996#	7000#	7001
	7033#	7040#	7041	7065#	7072#	7073	7089#	7099#	7109#	7121#	7130#	7138#	7146#	7151#	7160#
	7165#	7179#	7200#	7214#	7236#	7241#	7257#	7262#	7271#	7277#	7289#	7294#	7298#	7299	
M\$TAB	1#	1094#	1638#	1681#	1728#	1742#	1818#	1830#	1859#	1889#	1922#	1958#	1989#	2033#	2046#

CZDMTC.P 1 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

2062#	2069#	2250#	2295#	2303#	2330#	2339#	2347#	2353#	2435#	2565#	2576#	2583#	2745#	2792#
2795#	2805#	2809#	2817#	2821#	2829#	2833#	2841#	2845#	2853#	2857#	2871#	2875#	2886#	2890#
2899#	2903#	2909#	2917#	2921#	2927#	2933#	2938#	2942#	2956#	2960#	2969#	2973#	2982#	2986#
2996#	3000#	3012#	3021#	3025#	3033#	3037#	3380#	3410#	3414#	3418#	3422#	3441#	3479#	3492#
3495#	3497#	3507#	3510#	3525#	3527#	3543#	3575#	3588#	3591#	3594#	3601#	3611#	3652#	3662#
3679#	3686#	3692#	3727#	3737#	3754#	3761#	3767#	3802#	3812#	3829#	3836#	3842#	3877#	3887#
3904#	3911#	3917#	3952#	3962#	3979#	3986#	3992#	4027#	4037#	4054#	4061#	4067#	4086#	4093#
4098#	4107#	4114#	4132#	4139#	4143#	4152#	4158#	4225#	4229#	4236#	4242#	4246#	4250#	4259#
4268#	4271#	4303#	4311#	4326#	4346#	4350#	4366#	4384#	4388#	4411#	4419#	4426#	4432#	4450#
4456#	4461#	4468#	4482#	4484#	4502#	4510#	4515#	4527#	4531#	4536#	4541#	4548#	4549#	4558#
4559#	4568#	4576#	4595#	4609#	4630#	4639#	4655#	4665#	4681#	4686#	4695#	4700#	4709#	4730#
4738#	4746#	4755#	4759#	4765#	4771#	4793#	4810#	4814#	4822#	4832#	4840#	4858#	4885#	4895#
4906#	4912#	4941#	4954#	4971#	4983#	4998#	5011#	5030#	5044#	5059#	5073#	5083#	5094#	5104#
5121#	5139#	5153#	5166#	5176#	5183#	5201#	5215#	5228#	5239#	5246#	5263#	5277#	5290#	5301#
5308#	5321#	5332#	5343#	5354#	5366#	5378#	5380#	5390#	5402#	5416#	5425#	5438#	5449#	5458#
5468#	5470#	5478#	5489#	5499#	5510#	5512#	5529#	5539#	5547#	5555#	5560#	5571#	5584#	5590#
5595#	5600#	5606#	5612#	5623#	5637#	5651#	5664#	5675#	5688#	5701#	5714#	5724#	5737#	5748#
5759#	5773#	5786#	5796#	5803#	5824#	5834#	5842#	5853#	5871#	5882#	5892#	5899#	5918#	5932#
5942#	5952#	5962#	5972#	5982#	5992#	5998#	6015#	6029#	6039#	6049#	6059#	6069#	6079#	6089#
6095#	6114#	6128#	6142#	6155#	6169#	6182#	6195#	6209#	6222#	6235#	6248#	6261#	6288#	6296#
6299#	6315#	6325#	6336#	6342#	6358#	6372#	6382#	6393#	6399#	6419#	6430#	6440#	6450#	6461#
6468#	6485#	6496#	6508#	6514#	6547#	6556#	6563#	6596#	6605#	6612#	6637#	6647#	6654#	6671#
6682#	6701#	6714#	6727#	6740#	6753#	6765#	6769#	6776#	6817#	6851#	6876#	6881#	6910#	6935#
6940#	6971#	6996#	7001#	7033#	7041#	7065#	7073#	7089#	7099#	7109#	7121#	7130#	7138#	7146#
7151#	7160#	7165#	7179#	7200#	7214#	7236#	7241#	7257#	7262#	7271#	7277#	7289#	7294#	7299#
MS STL 1#	1094#	1638#	1681#	1728#	1742#	1818#	1830#	1859#	1889#	1922#	1958#	1989#	2033#	2046#
2062#	2069#	2250#	2295#	2303#	2330#	2339#	2347#	2353#	2435#	2565#	2576#	2583#	2745#	2792#
2795#	2805#	2809#	2817#	2821#	2829#	2833#	2841#	2845#	2853#	2857#	2871#	2875#	2886#	2890#
2899#	2903#	2909#	2917#	2921#	2927#	2933#	2938#	2942#	2956#	2960#	2969#	2973#	2982#	2986#
2996#	3000#	3012#	3021#	3025#	3033#	3037#	3380#	3410#	3414#	3418#	3422#	3441#	3479#	3492#
3495#	3497#	3507#	3510#	3525#	3527#	3543#	3575#	3588#	3591#	3594#	3601#	3611#	3652#	3662#
3679#	3686#	3692#	3727#	3737#	3754#	3761#	3767#	3802#	3812#	3829#	3836#	3842#	3877#	3887#
3904#	3911#	3917#	3952#	3962#	3979#	3986#	3992#	4027#	4037#	4054#	4061#	4067#	4086#	4093#
4098#	4107#	4114#	4132#	4139#	4143#	4152#	4158#	4225#	4229#	4236#	4242#	4246#	4250#	4259#
4268#	4271#	4303#	4311#	4326#	4346#	4350#	4366#	4384#	4388#	4411#	4419#	4426#	4432#	4450#
4456#	4461#	4468#	4482#	4484#	4502#	4510#	4515#	4527#	4531#	4536#	4541#	4548#	4549#	4558#
4559#	4568#	4576#	4595#	4609#	4630#	4639#	4655#	4665#	4681#	4686#	4695#	4700#	4709#	4730#
4738#	4746#	4755#	4759#	4765#	4771#	4793#	4810#	4814#	4822#	4832#	4840#	4858#	4885#	4895#
4906#	4912#	4941#	4954#	4971#	4983#	4998#	5011#	5030#	5044#	5059#	5073#	5083#	5094#	5104#
5121#	5139#	5153#	5166#	5176#	5183#	5201#	5215#	5228#	5239#	5246#	5263#	5277#	5290#	5301#
5308#	5321#	5332#	5343#	5354#	5366#	5378#	5380#	5390#	5402#	5416#	5425#	5438#	5449#	5458#
5468#	5470#	5478#	5489#	5499#	5510#	5512#	5529#	5539#	5547#	5555#	5560#	5571#	5584#	5590#
5595#	5600#	5606#	5612#	5623#	5637#	5651#	5664#	5675#	5688#	5701#	5714#	5724#	5737#	5748#
5759#	5773#	5786#	5796#	5803#	5824#	5834#	5842#	5853#	5871#	5882#	5892#	5899#	5918#	5932#
5942#	5952#	5962#	5972#	5982#	5992#	5998#	6015#	6029#	6039#	6049#	6059#	6069#	6079#	6089#
6095#	6114#	6128#	6142#	6155#	6169#	6182#	6195#	6209#	6222#	6235#	6248#	6261#	6288#	6296#
6299#	6315#	6325#	6336#	6342#	6358#	6372#	6382#	6393#	6399#	6419#	6430#	6440#	6450#	6461#
6468#	6485#	6496#	6508#	6514#	6547#	6556#	6563#	6596#	6605#	6612#	6637#	6647#	6654#	6671#
6682#	6701#	6714#	6727#	6740#	6753#	6765#	6769#	6776#	6817#	6851#	6876#	6881#	6910#	6935#
6940#	6971#	6996#	7001#	7033#	7041#	7065#	7073#	7089#	7099#	7109#	7121#	7130#	7138#	7146#
7151#	7160#	7165#	7179#	7200#	7214#	7236#	7241#	7257#	7262#	7271#	7277#	7289#	7294#	7299#
MS STL 1#	1094#	1165#	1174	1224#	1226	1227	1228	1229	1230	1231	1232	1233	1234	1235
1236	1237	1238	1239	1240	1241	1242	1243	1244	1245	1246	1247	1248	1249	1250
1251	1252	1253	1254	1255	1256	1257	1258	1259	1260	1261	1262	1263	1264	1265
1266	1267	1268	1269	1638#	1639	1640	1641	1681#	1682	1683	1684	1728#	1729	1730
1731	1742#	1743	1744	1745	1818#	1819	1820	1821	1830#	1831	1832	1833	1859#	1860

ZDMTC.F11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

	1861	1862	1889#	1890	1891	1892	1922#	1923	1924	1925	1958#	1959	1960	1961	1989#
	1990	1991	1992	2033#	2034	2035	2036	2046#	2047	2048	2049	2062#	2063	2064	2065
	2069#	2070	2071	2072	2250#	2251	2252	2253	2295#	2296	2297	2298	2435#	2436	2437
	2438	2745#	2746	2747	2748	3375#	3601#	3602	3603	3604	3652#	3653	3654	3655	3662#
	3663	3664	3665	3686#	3687	3688	3689	3727#	3728	3729	3730	3737#	3738	3739	3740
	3761#	3762	3763	3764	3802#	3803	3804	3805	3812#	3813	3814	3815	3836#	3837	3838
	3839	3877#	3878	3879	3880	3887#	3888	3889	3890	3911#	3912	3913	3914	3952#	3953
	3954	3955	3962#	3963	3964	3965	3986#	3987	3988	3989	4027#	4028	4029	4030	4037#
	4038	4039	4040	4061#	4062	4063	4064	4093#	4094	4095	4096	4139#	4140	4141	4142
	4225#	4226	4227	4228	4236#	4237	4238	4239	4246#	4247	4248	4249	4346#	4347	4348
	4349	4384#	4385	4386	4387	4510#	4511	4512	4513	4527#	4528	4529	4530	4536#	4549#
	4550	4551	4552	4559#	4560	4561	4562	4630#	4631	4632	4633	4681#	4682	4683	4684
	4695#	4696	4697	4698	4755#	4756	4757	4758	4765#	4766	4767	4768	4810#	4811	4812
	4813	4814#	4906#	4907	4908	4909	5301#	5302	5303	5304	5688#	5689	5690	5691	5992#
	5993	5994	5995	6089#	6090	6091	6092	6671#	6765#	6766	6767	6768	6776#	7322#	7327#
	7331#	7335#	7340#	7345#	7350#	7518	7519								
MSXFER	1#	1094#													
NEWST	1095#	3556	3618	3693	3768	3843	3918	3993	4068	4115	4160	4274	4579	4641	4710
	4774	4842	4914	5013	5124	5186	5249	5309	5515	5625	5807	5856	5902	6000	6097
	6265	6301	6344	6404	6470	6519	6567	6614	6656	6823	6882	6942	7003	7043	7074
OPEN	1#	1094#													
POINTE	1#	1094#	1124												
PRINTB	1#	1094#	2788	2801	2812	2824	2838	2850	2864	2880	2895	2906	2911	2924	2929
	2935	2949	2963	2976	2991	3007	3014	3030							
PRINTF	1#	1094#	2562	2573	2578	3673	3748	3823	3898	3973	4048	4101	4146		
PRINTS	1#	1094#													
PRINTX	1#	1094#	2324	2332	2343	2349	3606								
READBU	1#	1094#													
READEF	1#	1094#	3409	3413	3417	3421									
RFLAGS	1#	1094#													
SETPRI	1#	1094#	4425	4460	4540	4547	4557	4567	6850	6909	6970				
SETVEC	1#	1094#	3571	4407	4415	4446	4452								
SLASH	1#	1094#													
STARS	1#	1094#													
SVC	1#	1094#													
TOR	1599#	4256	4324	4364	4430	4466	4500	4727							
WFE	1691#	4829	4882	4980	5173	5236	5422	5721	5793	5839	5889	5949	5969	6046	6066
	6333	6390	6458	6505											
WFR	1689#	4308	4606	4662	4735	4819	4892	5568	5831	6322	6379	6447	6493	7197	
WFRO	1690#														
XFER	1#	1094#	3375#	4536#	4814#	6671#	6776#								
XFERF	1#	1094#													
XFERT	1#	1094#													
ZZ	3556#	3558	3618#	3620	3693#	3695	3768#	3770	3843#	3845	3918#	3920	3993#	3995	4068#
	4070	4115#	4117	4160#	4162	4274#	4276	4579#	4581	4641#	4643	4710#	4712	4774#	4776
	4842#	4844	4914#	4916	5013#	5015	5124#	5126	5186#	5188	5248#	5251	5309#	5311	5515#
	5517	5625#	5627	5807#	5809	5856#	5858	5902#	5904	6000#	6002	6097#	6099	6264#	6267
	6301#	6303	6344#	6346	6404#	6406	6470#	6472	6518#	6521	6566#	6569	6614#	6616	6656#
	6658	6823#	6825	6882#	6884	6942#	6944	7003#	7005	7043#	7045	7074#	7076		

. ABS. 037660 000

ERRORS DETECTED: 0

CZDMTC.P11 25-MAR-81 08:24

CROSS REFERENCE TABLE -- MACRO NAMES

CZDMTC.BIN,CZDMTC.SEQ/LRF/SOL=SVC34R.MAC,CZDMTC.P11
RUN-TIME: 32 40 5 SECONDS
RUN-TIME RATIO: 116/77=1.4
CORE USED: 21K (41 PAGES)