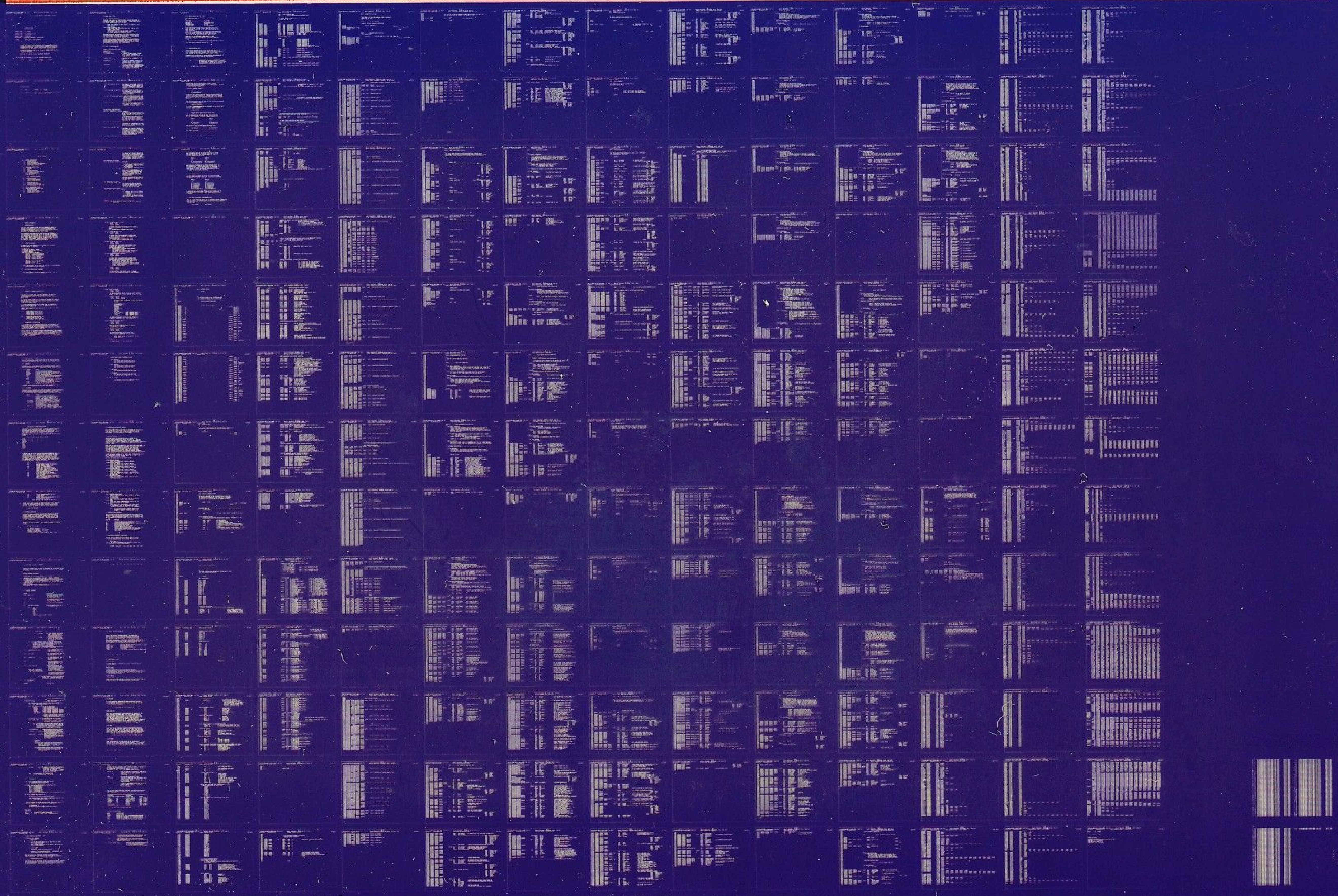
DMR11,DMC11 DMR/C11 DCLT
CZCLKA0

AH-F593A-MC
FICHE 1 OF 1

JUN 1980



```
                                  .TITLE  CZCLKA0 DMR,DMC-11 DATA COMM. LINK TEST




                                                      .REM &



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

                PRODUCT CODE:   AC-F591A-MC

                PRODUCT NAME:   CZCLKA0 DMR/C11 DCLT

                PRODUCT DATE:   17-APRIL-80

                MAINTAINER:     MERRIMACK DIAGNOSTIC ENGINEERING

                AUTHOR:         BRUCE LUHRS - BRUCE RIBOLINI



                      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 BY DIGITAL EQUIPMENT CORPORATION


                THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

                      DIGITAL         PDP             UNIBUS          MASSBUS
                      DEC             DECUS           DECTAPE
```

```
REVISION HISTORY:

REV      DATE            AUTHOR          REASON
---      ----            ------          ------

A        23-APR-80       BRUCE LUHRS     ORIGINAL ISSUE, DCLT FOR THE
                         BRUCE RIBOLINI    DMC OR DMR-11
```

TABLE OF CONTENTS

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS DCLT (DATA COMMUNICATION LINK TEST) PROGRAM IS MEANT TO PROVIDE FIELD SERVICE WITH A TOOL TO MAINTAIN DMR/DMC-11 TO DMR/DMC-11 COMMUNICATION LINKS. THIS DCLT PROGRAM WILL PROVIDE THE COVERAGE NECESSARY TO DETECT FAILURES TO THE COMPUTER EQUIPMENT, THE COMMUNICATION LINK, OR THE MODEM.

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, SLIDE AND PAPER TAPE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL (CHQUS?.SEQ WHERE ? IS REV. LEVEL OF THE MANUAL). THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

1.2 SYSTEM REQUIREMENTS

IN ORDER TO RUN THE DMR/DMC-11 DCLT PROGRAM, THE FOLLOWING MINIMUM HARDWARE IS REQUIRED:

```
- A PDP-11 CPU
- MINIMUM OF 24K WORDS OF MEMORY
- A WORKING, LINE OR REAL-TIME CLOCK
- A CONSOLE TERMINAL
- ANY XXDP+ SUPPORTED LOAD MEDIA
- ONE OF THESE DMR-11 OR DMC-11 CONFIGURATIONS:
        DMC11-AL - LOCAL MICROPROCESSOR
        DMC11-AL - REMOTE MICROPROCESSOR
        DMC11-DA - E.I.A. LINE UNIT
        DMC11-FA - CCITT V.35 LINE UNIT
        DMC11-MA - 1M BPS LINE UNIT
        DMC11-MD - 56K BPS LINE UNIT

        DMR11-AA - E.I.A. (RS 232/423)
        DMR11-AB - CCITT V.35
        DMR11-AC - LOCAL
        DMR11-AE - E.I.A. (RS 422)

IF DOWN-LINE-LOADING A DMC-11 SATELLITE, THE SATELLITE END REQUIRES:
        M9301-YJ/M9312 - BOOTSTRAP MODULE
```

1.3 RELATED DOCUMENTS AND STANDARDS

- XXDP+ USER'S MANUAL (CHQUS?.SEQ WHERE ? IS THE REV. LEVEL OF THE MANUAL - ''C'' IS THE CURRENT REV.).

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

THE GOAL OF THE DATA COMM. LINK TEST PROGRAM IS TO TEST THE COMMUNICATION LINK AND THEREFORE ASSUMES THAT THE CPU'S, CLOCKS, AND DMR OR DMC-11'S AT EACH END OF THE LINK HAVE ALREADY BEEN TESTED.

IF NO LINE OR REAL-TIME CLOCK IS FOUND, THE PROGRAM WILL CONTINUE BUT ANY OF THE PROGRAM THAT TIMES THE DEVICE WILL HANG IF THE DEVICE TIMES OUT. ALSO, THE EVENT LOG WILL CONTAIN A ZERO EVENT TIME FOR ALL EVENTS LOGGED.

IT IS NOT THE INTENTION OF A DATA COMM. LINK TEST PROGRAM TO TEST THE DMR OR DMC-11, BUT TO TEST THE COMMUNICATION LINK TO WHICH THEY ARE CONNECTED.

SOME OF THE DIAGNOSTICS THAT COULD BE RUN IF THE DMC-11 OR DMR-11 LOOKS BAD:

```
DMR:    CZDMIAO DMR-11 FCTNL DIAG
        CZDMPA1 M8207 STATIC DIAG #1
        CZDMQA2 M8207 STATIC DIAG #2
        CZDMRCO M8203 STATIC DIAG #1
        CZDMSCO M8203 STATIC DIAG #2

DMC:    CZDMCCO BSC W/R MICRO-PROC TST
        CZDMECO DDCMP MDLN UNIT TST
        CZDMGDO DMC-11 CROM + JMUP TEST
        MD-11-DZDMHB1  DMC-11 FREE RUNNING TEST
```

1.5 ASSUMPTIONS - RESTRICTIONS

IT IS ASSUMED THAT THE COMMUNICATIONS DEVICE (DMC OR DMR-11) HAS BEEN TESTED USING THE PREREQUISTE DIAGNOSTICS. THE OPERATOR SHOULD HAVE READ THE USER DOCUMENTATION PORTION OF THE LISTING TO FAMILIARIZE HIMSELF WITH THE COMMANDS AND CAPABILITIES AVAILABLE UNDER THE DIAGNOSTIC SUPERVISOR AND DCLT.

```
BECAUSE THE DMC-11 AND DMR-11 SUPPORT DDCMP OPERATION IN THE FIRMWARE,
THE PDP-11 D.C.L.T. PROGRAM IS UNABLE TO CONTROL OR KNOW EXACTLY WHAT
IS BEING TRANSMITTED AT ANY GIVEN TIME.  ALL DATA MESSAGES ARE ENCLOSED
IN A DDCMP ENVELOPE AND THERE MAY ALSO BE CONTROL MESSAGES
(AKS, NAKS,....) BEING TRANSMITTED.   BECAUSE OF THIS PLEASE BEWARE IF
IF YOU ARE SCOPING DATA.                                 ------ ------
```

2.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES. FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

2.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES (SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE ''STA'' INSTEAD OF ''START''.

2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY 'DDDDD''.

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

```
START/TESTS:1-5/PASS:1000/EOP:100
```

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE ''/TES:1-5'' INSTEAD OF ''/TESTS:1-5''.

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	'BELL'' ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)

IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A ''BELL'' ON ERROR, YOU MAY USE THE FOLLOWING STRING:

```
/FLAGS:LOE:IER:BOE
```

2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING ''CHANGE HW (L) ?'' YOU MUST ANSWER ''Y'' AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN ''PRELOADED'' USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A ''Y'', THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL).

THE DMR/DMC-11 DATA COMM. LINK TEST PROGRAM WILL NOT USE MORE THAN ONE UNIT. FOR THE DMC/DMR-11, THE HARDWARE INFORMATION REQUESTED WILL BE:

```
# UNITS (D) ? 1<CR>

UNIT 0
FULL DUPLEX OPERATION :  (L) Y ?
DMR,DMC-11 CSR ADDRESS : (O)  160170 ?
INTERRUPT VECTOR ADDRESS:  (O)  300  ?
INTERRUPT PRIORITY : (O)    5  ?
DEVICE OPTION TYPE : (0=DMC, 5=DMR-DMC MODE ,7=DMR) (O)  0 ?
```

2.5 DATA COMM. LINK TEST COMMANDS

THE 'DCLT>'' COMMAND LEVEL FOLLOWS THE ANSWERING OF THE HARDWARE P-TABLE QUESTIONS. THESE COMMANDS CAN BE TYPED WHEN THE 'DCLT> (A) ?'' PROMPT IS PRINTED.

MESSAGE COMMANDS AVAILABLE:

YOU ONLY HAVE TO TYPE ENOUGH CHARACTERS TO UNIQUELY SPECIFY A COMMAND.

THE COMMAND LINE IS INTERPRETED FROM LEFT TO RIGHT. THEREFORE, IF A QUALIFIER ON THE COMMAND LINE IS RELATED OR EFFECTS A QUALIFIER TO THE LEFT ON THE COMMAND LINE, THE QUALIFIER FARTHEREST TO THE RIGHT TAKES PRECEDENCE SINCE IT IS INTERPRETED LAST. (I.E. IF /CHECK..... .../NOCHECK APPEAR ON THE SAME LINE, NOCHECK WILL BE INDICATED IN THE PARAMETERS WORD.)

REFER TO SECTION 6.0 FOR A DESCRIPTION OF THE DIFFERENT MODES OF OPERATION AND THE TYPES OF MESSAGES AVAILABLE.

2.5.1 MESSAGE COMMANDS

```
COMMAND                               DESCRIPTION
---------                             ------------------------------

CLEAR   EXPECTLIST                    ZEROES THE EXPECTLIST (000'S)
                                       AND THEN PUTS DEFAULT ITEP MSG
                                       IN SO NOT REALLY EMPTY

CLEAR   TRANSMITLIST                  ZEROES TRANSMITLIST (000'S)
                                       AND THEN PUTS DEFAULT ITEP MSG
                                       IN SO NOT REALLY EMPTY

HELP                                  TYPES HELP INFO FOR OPERATOR
?

SET     EXPECTMSG=TYPE/QUAL           DEFINE A MESSAGE TO BE PUT ON
                                         THE EXPECTED LIST
        WHERE:  ''TYPE'' IS:
                =ONES
                =ZEROES
                =1ALT
                =0ALT
                =ITEP
                =CCITT
                =ALPHA
                =''A-Z,0-9,SPACES OR TABS IN QUOTES''
                WHERE THE OPTIONAL ''QUAL'' IS:

                            /SIZE=NNN       MAKE THE MESSAGE ''NNN'' BYTES
                                               LONG. (DEFAULT VALUE IS
                                               SIZE OF MESSAGE SPEC'D BY
                                               OPERATOR OR DEFAULTS.)
                            /COPY=NN        COPY THIS MESSAGE INTO THE
                                               BUFFER ''NN'' TIMES (DEFAULT
                                               IS 0 = PUT THE MESSAGE IN
                                               ONLY ONCE)

                NOTE: SET'S ADD MESSAGES TO THE LIST IN THE ORDER THEY'RE
                          DEFINED. ''NNN'' IS A DECIMAL NUMBER.  THE FIRST SET
                          OVERWRITES THE DEFAULT ITEP MESSAGE PLACED THERE BY
                          INITIALIZATION OR A ''CLEAR'' COMMAND.

                          SEE SECTION 6.2 FOR A DESCRIPTION OF THE PRE-DEFINED
                          MESSAGES THAT ARE AVAILABLE. (ZEROS,ONES ...)

        SET     TRANSMITMSG=TYPE/QUAL               DEFINE A MESSAGE TO BE PUT ON
                                                       THE TRANSMIT LIST
                                                       (SEE DESCRIPT FOR SET EXP)

        SHOW    EXPECTLIST                          LISTS THE MESSAGE SIZE AND TYPE
                                                      FOR THE MESSAGES IN THE
                                                      EXPECT LIST

        SHOW    TRANSMITLIST                        LISTS THE MESSAGE SIZE AND TYPE
                                                      FOR THE MESSAGES IN THE
                                                      TRANSMIT LIST

        PRINT                                       PRINTS THE EVENT LOG AFTER
                                                      ASKING THE OPERATOR IF HE
                                                      WANTS THE DMR/DMC-11 BASE
                                                      TABLE PRINTED

        DUMP    SSSSSS-EEEEEE/B                     PRINTS THE CONTENTS OF THE
                                                      MEMORY LOCATIONS BETWEEN
                                                      OCTAL ADDRESSES ''SSSSSS'' AND
                                                      ''EEEEEE'' WHERE ''SSSSSS'' IS
                WHERE ''/B'' IS OPTIONAL:             THE START ADDRESS AND
                  DEFAULT IS PRINT WORDS              ''-EEEEE'' IS THE END ADDRESS.
                  ''/B'' CAUSES PRINT BYTES
                                                      IF ''-EEEEEE'' IS NOT SPECIFIED
                                                      THEN THE CONTENTS OF ''SSSSSS''
                                                      IS PRINTED IN WORD FORMAT.

                            NOTE: THE DUMP COMMAND IS USEFUL FOR EXAMINING
                                      MESSAGE DATA.  STARTING ADDRESSES CAN
                                      BE FOUND BY LOOKING IN THE EVENT LOG.

        2.5.2 RUN COMMAND
              -----------

        COMMAND                                     DESCRIPTION
```

```
---------                                   ----------------------------

RUN MODE=MTYPE/QUAL                         STARTS DCLT EXECUTING IN THE
                                              MODE SPECIFIED

        NOTE: MODE=ACTIVE IS NOT DEFAULT, A MODE=MTYPE MUST BE TYPED
        -----    EACH TIME A RUN IS TYPED

        WHERE THE 'MTYPE'' IS ANY ONE OF THE FOLLOWING:
                 =ACTIVE         (FORCES /NOECHO ,NO LOOPING)
                 =PASSIVE        (FORCES NO LOOPING)
                 =RECEIVE        (FORCES /NOECHO ,NO LOOPING)
                 =LISTEN         (FORCES /NOECHO ,NO LOOPING, /NOCHECK)
                 =TRANSMIT       (FORCES /NOECHO ,NO LOOPING, /NOCHECK)
                 =TALK           (FORCES /NOECHO ,NO LOOPING, /NOCHECK)
                 =DOWNLINELOAD   (FORCES /NOECHO ,NO LOOPING, /NOCHECK,

                                 (FORCING NO LOOPING MEANS IT MUST BE
                                   SPECIFIED AS A QUALIFIER ANY TIME ITS
                                   DESIRED, THERE IS NO DEFAULT)

        AND OPTIONAL ''QUAL'' IS ANY COMBINATION OF THE FOLLOWING:

        /CHECK/NOCHECK          ENABLES/DISABLES CHECKING OF RECEIVED
                                  DATA AGAINST THE EXPECTED DATA

                NOTE: IF BOTH NODES IN ACTIVE AND ''/NOCHECK'' IS USED,
                ----- END-OF-PASS IS DEFINED AS RECEIVING 1 MESSAGE
                      AND COMPLETING THE TRANSMIT LIST.  WITH NO DATA
                      CHECKING, THERE IS NO WAY FOR DCLT TO KNOW HOW
                      MANY MESSAGES IT SHOULD EXPECT TO RECEIVE.

        /STATUS/NOSTATUS        ENABLES/DISABLES PRINTING OF PROGRAM
                                  STATUS MESSAGES TO THE OPERATOR
        /ECHO/NOECHO    ENABLES/DISABLES THE RETRANSMISSION OF
                                  THE DATA RECEIVED IN PASSIVE MODE.
                                  (IGNORED IN MODES OTHER THAN PASSIVE)
        /LOOP=LTYPE     SPECIFIES WHICH, IF ANY, TYPE OF
                                  MAINTENANCE LOOPBACK IS BEING USED.
                                  (IGNORED IN MODES OTHER THAN ACTIVE)
                                  MUST BE SPECIFIED EACH TIME ELSE NO
                                  LOOP IS USED.

                'LTYPE'' IS:
                 =INTERNALTTL
                 =CABLE
                 =LOCALMODEM (DMR IN DMR MODE AND RS449 MODEMS ONLY.
                        CAUSES A 'WRITE MODEM'' TO BE DONE TO SET UP
                        LOCAL-LOOPBACK (MAINT1) . ALSO CALLED
                        ANALOG-LOOPBACK.

                 =REMOTEMODEM (DMR IN DMR MODE AND RS449 MODEMS ONLY.
                        CAUSES A 'WRITE MODEM'' TO BE DONE TO SET UP
                        REMOTE-LOOPBACK (MAINT2) . ALSO CALLED
                        DIGITAL-LOOPBACK.
                /PASS=NN        SPECIFIES NUMBER OF ITERATIONS TO MAKE BEFORE
                                        END-OF-PASS.  DEFAULT VALUE OF 1
                                        WILL BE USED ON ANY RUN THAT A /PASS=N
                                        IS NOT ADDED TO THE ''RUN ...'' COMMAND.
                                        IF A ''-1'' IS TYPED, THEN THE PROGRAM
                                        RUN UNTIL A ^C IS TYPED.

                NOTE:   SEE SECTION 6.1 FOR A DESCRIPTION
                -----   OF THE ''RUN MODES'' AND ''LOOP MODES''


        2.5.3   DEFAULTS
                ---------

        IF NO ''SET'S'' THEN THE DEFAULT IS SAME AS IF TYPED:
                SET TRANSMITMSG=ITEP/SIZE=58/COPY=0
                SET EXPECTMSG=ITEP/SIZE=58/COPY=0

        THE DEFAULT COPY AND SIZE FOR EACH OF THE MESSAGE TYPES:
                ONES - /SIZE=64/COPY=0
                ZEROES - /SIZE=64/COPY=0
                0ALT - /SIZE=64/COPY=0
                1ALT - /SIZE=64/COPY=0
                CCITT - /SIZE=64/COPY=0
                ALPHA - /SIZE=65/COPY=0
                ITEP - /SIZE=58/COPY=0
                OPER. SPEC'D - /SIZE=LENGTH-OF-TEXT-TYPED-BETWEEN-QUOTES/COPY=0

        FOR THE RUN COMMAND THE DEFAULTS ARE:

                RUN MODE=ACTIVE/NOSTATUS/CHECK/NOECHO/PASS=1

                NOTE: MODE=ACTIVE IS NOT DEFAULT, A MODE=MTYPE MUST BE TYPED
                -----   EACH TIME A RUN IS TYPED


        IF THE DCLT PROGRAM IS RUN IN UNATTENDED MODE (UAM FLAG=1 OR CHAINED),
        THE DEFAULTS ARE AS IF THESE SETUP AND RUN COMMANDS WERE TYPED:

                SET TRANS=ITEP
                SET EXPECT=ITEP
                RUN MODE=ACTIVE/LOOP=INTERNAL/NOSTAT/CHECK/PASS=1


        OTHER NOTES:
        ------------
        ^C              ALWAYS RETURNS YOU TO ''DR>'' (THE SUPERVISOR)
        <CR>            IS SEEN AS A COMMAND TERMINATOR
        ''RUBOUT''      DELETE LAST CHAR. TYPED IN COMMAND STRING
```

2.6 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER THE LSI AND 50HZ (IF THERE IS A CLOCK) QUESTIONS
3. TYPE ''R NAME'', WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
4. TYPE ''START''
5. ANSWER THE ''CHANGE HW'' QUESTION WITH ''Y''
6. ANSWER ALL THE HARDWARE QUESTIONS. THE NUMBER OF UNITS THAT CAN DCLT CAN USE IS ALWAYS ''1''.

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS. THESE DEFAULTS ARE DESCRIBED IN SECTION 2.3.

7. AFTER THE ''DCLT> (A) ?'' PROMPT, TYPE ''RUN MOD=ACTIVE<CR>''

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING THE DEFAULT TRANSMIT AND EXPECTED MESSAGES. THE DEFAULT PASS COUNT AND ''RUN'' QUALIFIERS ARE ALSO BEING USED. THESE DEFAULTS ARE DESCRIBED IN SECTION 2.5.3.

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE ''IER'' FLAG IS SET (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

```
        NAME  TYPE  NUMBER  ON UNIT NUMBER  TST NUMBER PC:XXXXXX
        ERROR MESSAGE

,WHERE; NAME = DIAGNOSTIC NAME
        TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
        NUMBER = ERROR NUMBER
        UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
        TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
        PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL
```

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE ''IER'' OR ''IBE'' FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE ''IER'', ''IBE'' OR ''IXE'' FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

3.2 SPECIFIC ERROR MESSAGES

COMMAND LINE INTERPRETER ERRORS:

ERROR MESSAGE:	MEANING
?ILL CMD-BAD SYNTX?	A COMMAND WITH AN ILLEGAL CHAR WAS TYPED - RETYPE THE COMMAND. THE VALID COMMANDS AND THEIR SYNTAX ARE SHOWN IN SECTION 2.5.
?INCMPLTE CMD?	A REQUIRD PART OF A COMMAND WAS LEFT OUT.
?NUM TOO BIG?	THE VALUE OF A NUMERIC STRING IN THE COMMAND LINE WAS LARGER THAN 65535 OR 177777 OCTAL. (> 16 BITS).
?BAD RADIX?	A ''8'' OR ''9'' WAS TYPED WHEN AN OCTAL STRING WAS EXPECTED. PROBABLY OCCURRED WHEN TYPING A ''DUMP'' COMMAND WHERE OCTAL ADDRESSES ARE EXPECTED.

```
?''LOOP'' VALID ONLY IN ACTIVE?    THE ''/LOOP=..'' SWITCH WAS TYPED IN A
                                    RUN COMMAND BUT THE MODE WAS NOT SET
                                    TO ACTIVE.  MAINTENANCE LOOP IS ONLY
                                    POSSIBLE IF THE MODE OF OPERATION IS
                                    ACTIVE.

?''ECHO'' VALID ONLY IN PASSIVE?   THE ''/ECHO'' SWITCH WAS TYPED IN A
                                    RUN COMMAND BUT THE MODE WAS NOT SET
                                    TO PASSIVE. ECHOING OF RECEIVED DATA
                                    IS ONLY POSSIBLE IF THE MODE OF
                                    OPERATION IS PASSIVE.

?ILL CHR- ''A-Z,0-9,SP,TAB'' ONLY? A CHARACTER TYPED WITHIN QUOTES WHEN
                                     TRYING TO DEFINE THE CONTENTS OF A
                                     TRANSMIT OR EXPECT MESSAGE WAS NOT
                                     A ''A-Z,0-9,SPACE OR TAB''. RETYPE THE
                                     COMMAND WITH ONLY THESE CHARACTERS
                                     BETWEEN QUOTES.

?''SIZE=0'' NOT VALID?             A MESSAGE ZERO BYTES LONG CAN NOT BE
                                    BUILT. RETYPE THE COMMAND WITH A
                                    ''/SIZE=NNN''.  IF NO ''/SIZE='' IS TYPED
                                    A DEFAULT SIZE WILL BE USED.


DCLT OR DEVICE ERROR MESSAGES:
------------------------------
BAD CLOCK - PROGRAM WILL HANG ON ''TIMEOUT''!!
                                   THIS MEANS THAT EITHER NO CLOCK WAS
                                   ON THE SYSTEM OR THE ONE THAT WAS FOUND
                                   DID NOT INTERRUPT WHEN ASKED TO DO A
                                   ''TICK''.
                                   THE PROGRAM WILL STILL RUN, BUT ANY
                                   OF THE PROGRAM THAT TIMES THE DEVICE
                                   WILL HANG IF THE DEVICE TIMES OUT.
                                   ALSO, THE EVENT LOG WILL CONTAIN A
                                   ZERO EVENT TIME FOR ALL EVENTS LOGGED.
MAX. CHAR. MSG COUNT EXCEEDED - MSG. NOT BUILT !!

                                   THIS MEANS THAT THE TRANSMIT OR EXPECT
                                   BUFFER IS FULL. NO MORE MESSAGES CAN BE
                                   ADDED TO THAT BUFFER.
BUFFER FULL - MSG. NOT BUILT !!
                                   THIS MEANS THAT THE LAST MESSAGE YOU
                                   TRIED TO ADD TO EITHER THE TRANSMIT OR
                                   EXPECT BUFFER CAUSED THE TOTAL NUMBER
                                   OF MESSAGES TO BE EXCEEDED. NO MORE
                                   MESSAGES CAN BE ADDED TO THAT BUFFER.
                                   THE LIMIT IS DETERMINED BY THE SIZE OF
                                   THE MESSAGE POINTER TABLE.
CHAR. COUNT EXCEEDS BUFF LIMIT - MSG TRUNCATED
```

```
                                THIS MEANS THAT THE LAST MESSAGE YOU
                                TRIED TO ADD TO THE TRANSMIT OR EXPECT
                                BUFFER CAUSED THE TOTAL CHAR. COUNT
                                FOR THAT BUFFER TO EXCEED THE LIMIT.
                                THE MESSAGE WAS TRUNCATED TO COMPLETELY
                                FILL THE BUFFER. NO MORE MESSAGES CAN
                                BE ADDED TO THAT BUFFER.

DATA COMPARISON DATA ERROR
BYTE # IN MSG=XXX   EXPTD=YYY   RECVD=ZZZ

                                XXX= OFFSET OF THAT BYTE FROM THE START
                                   OF THE COMPARE OR EXPECT MESSAGE.
                                YYY= THE CONTENTS OF THAT BYTE IN THE
                                   EXPECTED MESSAGE
                                ZZZ= THE CONTENTS OF THAT BYTE IN THE
                                   RECEIVED MESSAGE

                                UP TO FIVE OF THESE ERRORS WILL BE
                                PRINTED PER MESSAGE COMPARED. ONLY
                                THE FIRST FIVE MISMATCHES WILL BE
                                INDIVIDUALLY REPORTED, BUT TOTAL
                                NUMBER OF MISTMATCHES IS REPORTED
                                BY ANOTHER ERROR.

                                PRINTING THE EVENT LOG AND USING THE
                                DCLT 'DUMP'' COMMAND WILL ALLOW YOU TO
                                FIND THE ADDRESS OF THE MESSAGE AND
                                EXAMINE IT.
DATA COMPARISON DATA ERROR
TOTAL MISMATCHES IN MSG = NNN
                                THIS MEANS THAT WHEN THE MESSAGE
                                RECEIVED WAS COMPARED AGAINST THE
                                MESSAGE THAT WAS EXPECTED, SOME OF
                                THE CHARS. WERE NOT THE SAME.

DATA COMPARISON LENGTH ERROR
COMPARE COUNT= XXX   RECEIVE COUNT= ZZZ

                                XXX= NUMBER OF BYTES IN THE COMPARE
                                        MESSAGE
                                ZZZ= NUMBER OF BYTES IN THE RECEIVED
                                        MESSAGE
                                THIS MEANS THAT THE MESSAGE RECEIVED
                                WAS A DIFFENT LENGTH THEN THE MESSAGE
                                THAT WAS EXPECTED.


*******
* NOTE *  - IN THE FOLLOWING ERROR DESCRIPTIONS XXXXX
********    REFERS TO THE OCTAL CONTENTS OF THE DEVICE REGISTERS
                SPECIFIED.
```

```
TIME OUT WAITING FOR RDI TO CLEAR
           SEL0     SEL2
           XXXXXX   XXXXXX

        THIS MEANS THAT A SOFTWARE TIMER EXPIRED BEFORE
        THE DEVICE CLEARED RDI IN RESPONSE TO THE DROPPING
        OF RQI.
        NOTE: PROGRAM RESETS TIMER AND WAITS AGAIN
              SO AN EFFECTIVE LOOP ON ERROR IS SETUP.

TIME OUT WAITING FOR RDI TO SET
           SEL0     SEL2
           XXXXXX   XXXXXX

        THIS MEANS THAT A SOFTWARE TIMER EXPIRED BEFORE
        THE DEVICE CAUSED AN INTERRUPT IN RESPONSE TO THE
        PROGRAM SETING RQI.
        NOTE: PROGRAM RESETS TIMER AND WAITS AGAIN
              SO AN EFFECTIVE LOOP ON ERROR IS SETUP.

TIME OUT WAITING FOR RUN TO SET
           SEL0     SEL2
           XXXXXX   XXXXXX

        THIS MEANS THAT A SOFTWARE TIMER EXPIRED BEFORE
        THE DEVICE SET THE RUN BIT IN RESPONSE TO THE
        PROGRAM SETING MASTER CLEAR.
        NOTE: PROGRAM RESETS TIMER AND ISSUES ANOTHER
              MASTER CLEAR AND WAITS AGAIN SO AN EFFECITVE
              LOOP ON ERROR IS SETUP.
              THIS ERROR COULD INDICATE WRONG ADDRESS FOR
              DMR/DMC-11 WAS GIVEN IN HARDWARE P TABLE.

TIME OUT WAITING FOR OUTPUT INTERRUPT
           SEL0     SEL2
           XXXXXX   XXXXXX

        THIS MEANS THAT A SOFTWARE TIMER EXPIRED BEFORE
        THE DEVICE SET OUTPUT INTERRUPT IN RESPONSE TO
        PROGRAM REQUESTING DEVICE TO TRANSMIT OR RECIEVE.
        NOTE: PROGRAM RESETS TIMER AND WAITS AGAIN SO AN
              EFFECTIVE LOOP ON ERROR IS SET UP.
              THIS ERROR WILL OCCUR WHEN ONE NODE IS STARTED
              IN RX OR TX MODE AND THE OTHER IS STILL BEING
              SET UP. IGNORE THIS ERROR IF PROGRAM CONTINUES
              WITHOUT FURTHER ERRORS.

INPUT INTERRUPT WHEN EXPECTING OUTPUT
           SEL0     SEL2
           XXXXXX   XXXXXX

        THIS WILL HAPPEN IF THE DEVICE IS BAD. IT MEANS
        THAT AFTER THE PROGRAM HAS ISSUED ALL INPUT REQUESTS
        TO THE DEVICE, THE DEVICE ISSUES AN INPUT INTERRUPT
```

```
ILLEGAL OUTPUT INTERRUPT
            SEL2      SEL6
            XXXXXX    XXXXXX

        THIS HAPPENS WHEN THE DEVICE ISSUES AN OUTPUT INTERRUPT
        WITHOUT SETTING 'RDO'. IF THIS HAPPENS THE DEVICE IS BAD.

CONTROL OUT INSTEAD OF BA-CC OUT
            SEL2      SEL6
            XXXXXX    XXXXXX    MMMMMMM

                WHERE 'MMMMM' IS ONE OF THE FOLLOWING MESSAGES
                THAT RESULT FROM INTERPRETING THE REGISTER CONTENTS
                FOR YOU:

                PROCEDURE ERROR/HALT
                NON EXIST MEM
                DDCMP START REC
                DISCONNECT
                LOST DATA
                DDCMP MAINT REC
                OVERRUN
                TIME OUT
                DATA CHECK
                RUN SET ILLEAGLLY    (DMR IN DMR-MODE ONLY)
                CD GLITCHED          (DMR IN DMR-MODE ONLY)
                RX IDLE              (DMR IN DMR-MODE ONLY)
                CTS FALILED          (DMR IN DMR-MODE ONLY)

        THIS ERROR OCCURS WHEN THE DEVICE SETS CONTROL OUT
        TO INDICATE ERROR CONTIDION. THE PROGRAM EXPECTS A
        BACC OUT.

TX BUFF COMPLETED AND SHOULD BE RX
            SEL4      SEL6
            XXXXXX    XXXXXX

        THIS ERROR OCCURS WHEN THE THE DEVICE HAS
        A BACC OUT WITH TX COMPLETED AND THE PROGRAM
        WAS EXPECTING A RX COMPLETED.

RX BUFF COMPLETED AND SHOULD BE TX
            SEL4      SEL6
            XXXXXX    XXXXXX

        THIS ERROR OCCURS WHEN THE THE DEVICE HAS
        A BACC OUT WITH RX COMPLETED AND THE PROGRAM
        WAS EXPECTING A TX COMPLETED.

                WHERE ''XXXXX'' IS THE OCTAL CONTENTS OF THAT
                DEVICE REGISTER.

DOWN LINE LOAD ABORTED

        THIS ERROR CAN ONLY OCCUR IN A NODE THAT
        IS A DLL ''HOST'' WHEN IT HAPPENS IT ALSO
        PRINTS ONE OF THE FOLLWING QUALIFERS:

        TX NOT COMPLETE
                HOST DEVICE DID NOT GIVE BACC OUT TX
                THIS SHOULD NOT HAPPEN BECAUSE DEVICE
                DOES NOT NEED AN ACK FOR MAINT MESGS.

        RX NOT COMPLETE
                HOST DEVICE DID NOT GIVE BACC OUT RX
                THIS CAN HAPPEN IF SATELLITE DOES NOT
                SEND THE SEC BOOT REQUEST MESSAGE.

        SEC REQ WORD1
                HOST RECIEVED A MESSAGE FROM SATELLITE
                BUT MESSAGE WAS NOT 1ST WORD OF SEC BOOT REQ.

        SEC REQ WORD2
                HOST RECIEVED A MESSAGE FROM SATELLITE
                BUT MESSAGE WAS NOT 2ND WORD OF SEC BOOT REQ.


CALLED FROM PC. XXXXXX

                THIS MESSAGE OCCURS WITH OTHER ERROR MESAGES
                TO INDICATE PC OF CALLING ROUTINE.
```

4.0 PERFORMANCE AND PROGRESS REPORTS

DCLT USES IT'S OWN METHOD FOR DETERMINING AN ''END OF PASS''
WHICH IS CALLED A ''DCLT END OF PASS''. THE NUMBER OF ''DCLT PASSES''
TO BE RUN IS SPECIFIED BY THE ''/PASS=XXX'' SWITCH ON THE DCLT
RUN COMMAND. THE TOTAL NUMBER OF ''DCLT ERRORS'' IS REPORTED
WHEN '' X NUMBER OF DCLT PASSES'' ARE COMPLETED.

4.1 PRINTING OF EVENT LOG

SIGNIFICANT EVENTS OR CHECK-POINTS WILL BE LOGGED IN A
''CIRCULAR QUEUE'' STORAGE AREA CALLED THE EVENT LOG. THE LAST
''N'' EVENTS ARE KEPT LOGGED AND CAN BE LISTED ON THE OPERATORS
CONSOLE BY GIVING A ''PRINT'' COMMAND AT THE ''DR>''(DIAGNOSTIC SUPERVISOR)
OR ''DCLT>'' (DCLT) LEVEL. THE EVENTS ARE PRINTED IN A ''LAST-IN
FIRST-OUT'' ORDER.

EVENT TIME IS TYPED OUT AS MMM:SS:TT (LIKE 254:36:07) WHERE MMM,SS,TT
REPRESENT THE NUMBER OF MINUTES, SECONDS, CLOCK TICKS SINCE THE LAST
START OR RESTART. IT SHOULD BE NOTED THAT THE TIMES ARE
RELATIVE SINCE WHILE THE PROCESSOR IS RUNNING AT PRIORITY 7
THE CLOCK CAN'T INTERRUPT TO KEEP TIME. THIS IS THE CASE
WHILE THE PROGRAM IS FETCHING DCLT COMMANDS FROM THE OPERATOR.
IT SHOULD ALSO BE NOTED THAT THERE ARE ONLY 8 BITS AVALIABLE TO STORE
RELATIVE MINUTES SO ''TIME'' WILL WRAP TO 000:00:00 AFTER 256:59:59.

A START OR RESTART COMMAND AT THE ''DR>'' LEVEL INITIALIZES THE EVENT
LOG. THEREFORE IT IS WISE TO DO A ''PRINT'' AT THE ''DR>'' LEVEL
BEFORE GIVING A ''START'' OR ''RESTART''.

THE TYPES OF EVENTS KEPT IN THE EVENT LOG ARE:

```
TRANSMIT MESSAGE QUEUED:
        EVENT TIME, ADDRESS OF 1ST BYTE OF MESSAGE,
        TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.
TRANSMIT MESSAGE COMPLETED:
        EVENT TIME, ADDRESS OF 1ST BYTE OF MESSAGE,
        TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.
RECEIVE SPACE QUEUED:
        EVENT TIME, ADDRESS OF 1ST BYTE OF MESSAGE,
        TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.
RECEIVE MESSAGE COMPLETED:
        EVENT TIME, ADDRESS OF 1ST BYTE OF MESSAGE,
        TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.
DATA COMPARISON STARTED:
        EVENT TIME, ADDRESS OF 1ST BYTE OF RECEIVED MSG.,
        TOTAL NO. OF BYTES IN RCV. MSG., TOTAL NO. OF BYTES
        IN EXPECT MSG.
DATA COMPARISON DATA ERROR:
        EVENT TIME, ADDRESS OF 1ST BYTE OF RECEIVED MSG.,
        TOTAL NO. OF BYTES IN RCV. MSG., TOTAL NO. OF
        COMPARISON FAILURES
```

DATA COMPARISON LENGTH ERROR:
EVENT TIME, ADDRESS OF 1ST BYTE OF RECEIVED MSG., TOTAL NO. OF BYTES IN RCV. MSG., TOTAL NO. OF BYTES IN EXPECT MSG.

DEVICE INIT AND SETUP:
EVENT TIME, MODE OF OPERATION, TYPE OF MAINTENANCE LOOP, 'DCLT'' PASS COUNT, 'RUN'' PARAMETERS

DEVICE ERROR:
EVENT TIME, DEVICE ERROR MESSAGE, CONTENTS OF TWO REGISTERS RELATING TO THE ERROR.

END OF PASS:
EVENT TIME, 'DCLT'' PASS COUNT, 'DCLT'' ERROR COUNT, NO. OF 'NOBUFF'S''(NO. OF CONTROL-OUTS WITH THE NO-BUFFER SET SINCE THE LAST 'DCLT RUN'' COMMAND.)

NOTE: IF THE NODES ON THE LINK ARE SIMILAR WITH RESPECT TO CONSOLE SPEED AND SETUP, THE NUMBER OF 'NOBUFFS'' SHOULD BE NEAR ZERO.

4.2 OPERATOR STATUS MESSAGES

THE ''/STATUS, /NOSTATUS'' QULAIFIERS FOR THE DCLT 'RUN'' COMMAND ENABLES/DISABLES THE PRINTING OF PROGRAM STATUS MESSAGES TO THE OPERATOR. THESE MESSAGES ARE INTENDED TO TELL THE OPERATOR WHAT THE DCLT PROGRAM IS CURRENTLY DOING. BELOW ARE THE MESSAGES THAT MIGHT BE PRINTED AND THEIR MEANING:

MESSAGE	MEANING
TXQ	DEVICE IS ABOUT START TRANSMITING A MESSAGE
TXC	TRANSMISSION OF MESSAGE COMPLETED
RXQ	DEVICE HAS QUEUED SPACE TO RECEIVE/ COMPLETED RECEIVE
ERR	DEVICE ERROR HAS OCCURRED
INI	DEVICE ABOUT TO BE INITIALIZED
MSC	ABNORMAL MODEM STATUS CHANGE
CMP	ABOUT TO DO DATA CHECKING OF RECVD VS. EXPTD DATA
CML	LENGTH ERROR OCCURRED DURING DATA COMPARISON
CMD	DATA ERROR OCCURRED DURING DATA COMPARISON
EOP	END OF PASS

4.3 PRINTING OF DMR/DMC-11 BASE TABLE

WHEN THE 'PRINT'' COMMAND IS GIVEN, BEFORE THE EVENT LOG IS PRINTED, THE DCLT DMR/DMC-11 DCLT PROGRAM ASKS IF YOU WISH TO HAVE THE CONTENTS OF THE DEVICE'S BASE TABLE PRINTED:

```
        BASE TABLE (L) N ?
```

IF A 'Y'' IS TYPED AS AN ANSWER, THE 256. BYTES OF BASE TABLE WILL BE PRINTED IN THE FOLLOWING FORMAT:

```
        017370:   000   001   002   003   004   005   006   007
        017400:   010   011   012   013   014   015   016   017
```

```
                                 ..      ..     ..     ..     ..     ..     ..     ..     ..
                                 ..      ..     ..     ..     ..     ..     ..     ..     ..
                             017760:    370    371    372    373    374    375    376    377
```

5.0 DEVICE INFORMATION TABLES

THIS IS THE DEFAULT HARDWARE P-TABLE. THE VALUES AND
SIZE ARE USED AS A ''TEMPLATE'' FOR CREATING ACTUAL P-TABLE
ENTRIES AND THE DEFAULT VALUES PROVIDED FOR THE OPERATOR.
SEE SECTION 2.4 FOR AN EXAMPLE OF THE HARDWARE QUESTIONS.

THE NUMBERS IN BRACKETS (I.E. [10]) INDICATES THE OFFSET OF THE
WORD INTO THE HARDWARE P-TABLE. THE OFFSETS MUST MATCH THE P-TABLE
OFFSETS USED IN THE HARDWARE PARAMETER CODING SECTION WHERE THE
''GET PARAMETER'' CALLS ARE USED TO FILL THE P-TABLE.

```
.WORD   1               ;[0] FULL OR HALF DUPLEX FLAG (BIT0=1 IF FULL)
.WORD   160170          ;[2] CSR ADDRESS
.WORD   300             ;[4] INTERRUPT VECTOR
.WORD   240             ;[6] INTERRUPT PRIORITY (5)
.WORD   0               ;[10] SPARE
.WORD   0               ;[12] OPTION TYPE(0=DMC,5=DMR-DMC MODE,7=DMR)
```

6.0 MODE AND MESSAGE DESCRIPTIONS

6.1 MODE DESCRIPTIONS

TRANSMIT MODE

A LIST OF MESSAGES IS TRANSMITTED WITHOUT EXPECTING ANY DATA
TO BE RECEIVED.

RECEIVE MODE

SPACE IS QUEUED FOR THE DEVICE TO RECEIVE MESSAGES.
AFTER RECEIVING AN ''EXPECTED'' NUMBER OF MESSAGES, THE DATA RECEIVED
CAN BE COMPARED AGAINST A LIST OF ''EXPECT TO RECEIVE'' MESSAGES
IF DATA-CHECKING IS ENABLED.

PASSIVE MODE

THEN EVERY TIME A MESSAGE IS RECEIVED, A MESSAGE IS TRANSMITTED.
DATA CHECKING CAN BE DONE ON THE RECEIVED DATA.THE ''/ECHO, /NOECHO''
ENABLES/DISABLES THE RETRANSMISSION OF THE DATA RECEIVED.

ACTIVE MODE

A LIST OF MESSAGES IS TRANSMITTED AND MESSAGES ARE RECEIVED. AFTER RECEIVING AN ''EXPECTED'' NUMBER OF MESSAGES, THE DATA RECEIVED CAN BE COMPARED AGAINST A LIST OF ''EXPECT TO RECEIVE'' MESSAGES IF DATA-CHECKING IS ENABLED.

NOTE: IF BOTH ENDS OF THE LINK ARE IN ACTIVE MODE, THEN THE
LINK MUST BE A FULL DUPLEX LINK!

DOWN-LINE-LOAD

THE ''HOST'' OR ORIGINATING STATION REQUESTS THE ''SATELLITE'' OR BOOT STATION TO ENTER MOP MODE. THE SATELLITE THEN SENDS A ''SECONDARY BOOT REQUEST MESSAGE''. THE ''HOST'' THEN CHECKS THE RECEIVED MESSAGE TO SEE THAT IT IS A ''SECONDARY BOOT REQUEST''. THEN THE HOST SENDS A ''MEMORY LOAD WITH TRANSFER ADDRESS'' THAT CONTAINS IMAGE DATA TO BE LOADED BY THE SATELLITE'S M9301-YJ/M9312 STARTING AT LOC. 0. THIS IMAGE DATA WILL CONTAIN A CODE THAT PRINTS A MESSAGE SAYING DOWN-LINE-LOAD WAS SUCESSFUL. THE BOOTING PROCESS OVERWRITES PART OF THE ''VECTOR'' AREA SO THE DCLT PROGRAM MUST BE RELOADED IN THE ''SATELLITE'' SYSTEM.

TALK MODE

THE ''TALK'' END OF THE LINK TRANSMITS OPERATOR-TYPED MESSAGES UNTIL A ''EXIT'' MESSAGE IS TYPED. AT THAT POINT, THE NODE GOES INTO ''LISTEN'' MODE. AN ''EXIT MESSAGE'' IS A MESSAGE WHOSE FIRST FOUR CHARACTERS ARE ''EXIT''. SINCE ONLY THE FIRST FOUR CHARACTERS NEED TO BE ''EXIT'', MORE CHARACTERS CAN BE ADDED SO THAT A MESSAGE MAY BE SENT AND THE MODE SWITCHED ALL AT ONCE. FOR EXAMPLE:

```
        TLK> EXIT ALL OF THIS LINE IS SENT THEN MODE SWTICHED
```

LISTEN MODE

THE ''LISTEN'' END OF THE LINK PRINTS ALL OF THE MESSAGES RECEIVED BY THE DEVICE ON THE OPERATOR'S CONSOLE. IF THE MESSAGE RECEIVED IS AN ''EXIT'' MESSAGE, THEN THE NODE ENTERS ''TALK'' MODE. AN ''EXIT MESSAGE'' IS A MESSAGE WHOSE FIRST FOUR CHARACTERS ARE ''EXIT''.

MAINTENANCE ''LOOP'' MODES

REMEMBER THAT THE WHENEVER A ''RUN'' COMMAND IS TYPED, THE DEFAULT IS NO LOOPBACK AND THAT A LOOP MODE MUST BE SPECIFIED BY A ''/LOOP=..'' IF A LOOP MODE IS DESIRED.
LOOP MODES ARE ONLY VALID IF THE MODE TO RUN IS ACTIVE !

INTERNALTTL — THE ''LU LOOP'' BIT IS SET SO THAT THE UNIT'S SERIAL LINE OUT IS LOOPED BACK TO THE SERIAL LINE IN AT THE TTL LEVEL BEFORE LEVEL CONVERSION.

CABLE — NOT USED BY DMR OR DMC-11 CODE.

LOCALMODEM — FOR DMR-11 IN DMR MODE AND RS449 MODEMS ONLY. CAUSES A ''WRITE MODEM'' TO BE DONE TO SET UP LOCAL-LOOPBACK (MAINT1) . ALSO CALLED ANALOG-LOOPBACK.

REMOTEMODEM — FOR DMR-11 IN DMR MODE AND RS449 MODEMS ONLY. CAUSES A ''WRITE MODEM'' TO BE DONE TO SET UP REMOTE-LOOPBACK (MAINT2) . ALSO CALLED DIGITAL-LOOPBACK.

THE FOLLOWING TABLE SUMMARIZES THE MODES THAT CAN BE RUN TOGETHER WHEN THE DCLT PROGRAM IS RUNNING ON TWO PROCESSORS (ONE AT EACH END OF THE LINK):

STATION A ''HOST'' NODE	''/LOOP'' ALLOWED?	STATION B ''REMOTE'' NODE	DUPLEX
TALK	NO	LISTEN*, RECEIVE	HALF OR FULL
LISTEN	NO	TALK*, TRANSMIT	HALF OR FULL
TRANSMIT	NO	RECEIVE*, LISTEN	HALF OR FULL
RECEIVE	NO	TRANSMIT*, TALK	HALF OR FULL
PASSIVE	NO	ACTIVE*	HALF OR FULL
ACTIVE	YES	ACTIVE*	FULL
		PASSIVE*	HALF OR FULL
DOWNLINELOAD	NO	PASSIVE	HALF FORCED

*= MOST LIKELY TO BE IN THAT MODE

6.2 MESSAGE DESCRIPTIONS

NAME	DESCRIPTION
ZEROES	MESSAGE OF ALL 0'S (00000000,00000000,00000000,....)
ONES	MESSAGE OF ALL 1'S (11111111,11111111,11111111,....)
1ALT	MESSAGE OF ALTERNATING 1'S (10101010,10101010,.....)
0ALT	MESSAGE OF ALTERNATING 0'S (01010101,01010101,.....)
CCITT	''CCITT'' 512-BIT (VS. 511 BITS) TEST PATTERN

```
ITEP            ''INTERPROCESSOR TEST PROGRAM'S (ITEP)'' MESSAGE 1(DP1:)
                (<177><177>/$A THE QUICK BROWN FOX JUMPED OVER THE
                LAZY DOG.<15><12><001><177><177><177><177>)

ALPHA           ALPHA-NUMERICS (OR FUTURE COMM TURNAROUND MSG)
                (#$!'' (AMPERSAND)'()*+,-.0123456789:;<=>?@ABCDEFGHIJK
                LMNOPQRSTUVWXYZ/[\]^_%)

''A-Z,0-9,SPACES,TABS''         THESE ARE THAT THE CHARACTERS THAT CAN
                                BE TYPED BETWEEN QUOTATION MARKS (''..'')
                                TO SPECIFIY A UNIQUE MESSAGE.
                                (CALLED AN OPERATOR SPECIFIED MESSAGE.)
```

6.3 INTERFACING TO AN ''ITEP'' NODE

WHEN DCLT IS USED TO INTERFACE TO AN ITEP NODE.
THE TABLE BELOW APPLIES:

ITEP NODE	DCLT NODE
ONE-WAY-OUT	RECEIVE OR LISTEN
ONE-WAY-IN	TRANSMIT OR TALK
INTERNAL LOOP	ACTIVE
EXTERNAL LOOP	ACTIVE OR PASSIVE

NOTE: WHEN INTERFACING TO ITEP IF THE RX BUFFER ON THE ITEP SIDE IS ONLY 10 BYTES LARGER THAN THE TX BUFFER YOU HAVE SELECTED, SO BE SURE TO SET THE TX BUFFER ON THE DCLT NODE ACCORDINGLY.

WHEN ITEP IS IN A MODE THAT IT IS EXPECTING TO BE TRANSMITTED TO, A SOFT ERROR ''BASE TABLE ERR COUNTS NON-ZERO'' WILL OCCUR. THIS IS DUE TO THE SPEED DIFFERENCES IN THE SOFTWARE.

WHEN DCLT IS IN LISTEN MODE THE RX BUFFER IS ONLY 82 BYTES LONG THEREFORE DO NOT SEND THE DCLT NODE ITEP MSG. 3 FROM THE ITEP NODE OR A ''LOST DATA'' ERROR WILL OCCUR

BE SURE ITEP NODE HAS INCORPERATED PATCH FROM DEPO# MD-11-DZDMO-A1

ITEP NODE SHOULD ALWAYS BE RUN WITH SW 4 = TO 0

6.4 TROUBLESHOOTING HINTS

LISTED BELOW ARE SOME SETUPS THAT COULD BE USED FOR ISOLATING FAULTS. THESE ARE BY NO MEANS THE ONLY WAYS DCLT CAN BE USED !!!!!!! DCLT IS MEANT TO BE A VERY FLEXIBLE TOOL! THIS SECTION IS MEANT TO GIVE SOMEONE NOT TOO FAMILIAR WITH DCLT A PLACE TO START.

REMEMBER THAT THE PRINTING OF STATUS MESSAGES AND PRINTING OF THE EVENT LOG CAN PROVIDE A LOT OF INFORMATION ABOUT THE SEQUENCE OF EVENTS AND HOW THE DEVICE AND LINK ARE BEHAVING.

NOTE: IF BOTH NODES IN ACTIVE AND ''/NOCHECK'' IS USED,
----- END-OF-PASS IS DEFINED AS RECEIVING 1 MESSAGE AND COMPLETING THE TRANSMIT LIST. WITH NO DATA CHECKING, THERE IS NO WAY FOR DCLT TO KNOW HOW MANY MESSAGES IT SHOULD EXPECT TO RECEIVE.

1.) INTERNAL LOOP AT EACH NODE

RUN EACH END OF THE LINK IN ACTIVE MODE WITH LOOP=INTERNAL.
TRANSMIT TWO OR THREE MESSAGES WITH NO DATA CHECKING.
STATUS PRINTING COULD BE TURNED OFF IF ON, BUT SEEING THE SEQUENCE
OF EVENTS MIGHT BE INFORMATIVE.

A POSSIBLE COMMMAND SEQUENCE IS:

```
        C E
        C T
        SE T=ONES/S=20/C=2
        R M=A/LO=I/NOCH/STAT
```

THIS GIVES YOU A IDEA IF THE COMM. DEVICE CAN EVEN TRANSMIT AND
RECEIVE. ANY ERRORS REPORTED WILL PROBABLY BE DUE TO INCORRECT
DEVICE ADDRESSES BEING USED OR A FAULTY DEVICE. CHECK ADDRESSES
WITH 'DISPLAY'' AND RUN THE PREREQUISTE DIAGNOSTICS FOR THE COMM.
DEVICE.

NOW TRY RUNNING EACH NODE THE SAME WAY WITH DATA CHECKING ENABLED.
A POSSIBLE COMMMAND SEQUENCE IS:

```
        R M=A/LO=I/CH/PAS=3
```

IF A CABLE TURNAROUND CONNECTOR IS AVAILABLE, PUT IT ON THE END OF
THE CABLE JUST BEFORE THE MODEM AND RUN IN ACTIVE MODE WITH NO LOOP.
POSSIBLE COMMAND SEQUENCE IS:

```
        R M=A/CH/PAS=3
```

2.)   TRANSMIT ON ONE NODE RECEIVE ON THE OTHER

NOW TRY TRANSMIITING FROM ONE END AND RECEIVING ON THE
OTHER. MAYBE WITH NO DATA CHECKING AT FIRST TO ESTABLISH
IF THE LINK IS WORKING. POSSIBLE COMMAND SEQUENCES ARE:

```
        NODE A                          NODE B
        ------                          ------
        C E                             C E
        C T                             C T
        SE T=1ALT/S=250                 SE E=1ALT/S=250
        R M=TR/PAS=3                    R M=R/NOCH/PAS=3
```

NOW TRY DOING DATA CHECKING ON THE MESSAGE(S) BEING
TRANSMITTED. POSSIBLE COMMAND SEQUENCES ARE:

```
        R M=TR/PAS=3                    R M=R/CH/PAS=3
```

NOW RUN THRU THE SEQUENCE AGAIN WITH NODE A RECEIVING
AND NODE B RECEIVING.

3.)   ONE NODE ACTIVE THE OTHER NODE PASSIVE

NOW TRY RUNNING ONE NODE IN ACTIVE MODE WHILE THE OTHER
END RUNS IN PASSIVE. DATA CHECKING SHOULD BE TURNED OFF
IF THE MESSAGE LISTS ARE NOT THE SAME.
POSSIBLE COMMAND SEQUENCES ARE:

```
        NODE A                  NODE B
        ------                  ------
        C E                     C E
        C T                     C T
        SE T=CCITT/S=10/C=2     SE T=1ALT/S=20/C=2
        R M=ACT/NOCH/PAS=3      R M=P/NOCH/PAS=3
```

NOW USE DATA CHECKING WITH THE ''EXPECT MESSAGE LISTS'' SET
UP APPROPRIATELY. ANOTHER VARIATION IS TO HAVE LARGE SIZE
MESSAGES ON ONE SIDE WITH SMALL MESSAGES ON THE OTHER.

THEN REVERSE THE SETUP SO THAT THE NODE RUNNING IN ACTIVE
IS RUNNING IN PASSIVE AND VICE VERSA.

4.) BOTH NODES ACTIVE

NOW BOTH NODES CAN BE RUN IN ACTIVE WITH DATA CHECKING ON.
STATUS PRINTING COULD BE TURNED OFF IF YOU'RE NOT INTERESTED
IN THEM.

```
        NODE A                  NODE B
        ------                  ------
        C E                     C E
        C T                     C T
        SE T=0ALT/S=10          SE E=0ALT/S=10
        SE T=CCITT/S=20         SE E=CCITT/S=20
        SE T=ALPHA/S=30         SE E=ALPHA/S=30
        SE E=ZERO/S=11          SE T=ZERO/S=11
        SE E=ONES/S=21          SE T=ONES/S=21
        SE E=ITEP/S=31          SE T=ITEP/S=31
        R M=A/CH/NOST/PAS=3     R M=A/CH/NOST/PAS=3
```

A VARIATION THAT CAN BE USED IS FOR ONE END TO SEND A LOT OF
SMALL MESSAGES AND THE OTHER TO SEND A FEW LARGE MESSAGES.
THE ''END-OF-PASS'' POINT WILL BE OUT OF SYNC BUT THIS IS NOT
A PROBLEM.

5.) TALK AND LISTEN MODES FOR COMMUNICATING

TALK AND LISTEN MODES ARE USEFUL IF THE OPERATORS WISH TO COMMUNICATE
WITH EACH OTHER. JUST SETUP A TIME THAT EACH WILL GO TO THEIR MODE,
TALK OR LISTEN, AND SEND MESSAGES OVER THE LINK. POSSIBLE COMMAND
SEQUENCES ARE.

```
        R M=LIS/NOST            R M=TA/NOST
```

```
                LIS>                    TLK>

                                            &
```

```

                                  .SBTTL  PROGRAM HEADER

      002000                              BGNMOD





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

      002000                              POINTER BGNRPT,BGNAU,BGNDU



      002000                              HEADER  CZCLK,A,0,1800.,0
      002000                                                                    L$NAME::
      002000       103                                                                  .ASCII /C/
      002001       132                                                                  .ASCII /Z/
      002002       103                                                                  .ASCII /C/
      002003       114                                                                  .ASCII /L/
      002004       113                                                                  .ASCII /K/
      002005       000                                                                  .BYTE   0
      002006       000                                                                  .BYTE   0
      002007       000                                                                  .BYTE   0
      002010                                                                    L$REV::
      002010       101                                                                  .ASCII  /A/
      002011                                                                    L$DEPO::
      002011       060                                                                  .ASCII  /0/
      002012                                                                    L$UNIT::
      002012    000000                                                                  .WORD   0
      002014                                                                    L$TIML::
      002014    003410                                                                  .WORD   1800.
      002016                                                                    L$HPCP::
      002016    035764                                                                  .WORD   L$HARD
      002020                                                                    L$SPCP::
      002020    000000                                                                  .WORD   0
      002022                                                                    L$HPTP::
      002022    002130                                                                  .WORD   L$HW
      002024                                                                    L$SPTP::
      002024    000000                                                                  .WORD   0
      002026                                                                    L$LADP::
      002026    036346                                                                  .WORD   L$LAST
      002030                                                                    L$STA::
      002030    000000                                                                  .WORD   0
      002032                                                                    L$CO::
      002032    000000                                                                  .WORD   0
      002034                                                                    L$DTYP::
      002034    000000                                                                  .WORD   0
      002036                                                                    L$APT::
      002036    000000                                                                  .WORD   0
      002040                                                                    L$DTP::
      002040    002124                                                                  .WORD   L$DISPATCH
```

```
002042                 L$PRIO::
002042  000000                   .WORD   0
002044                 L$ENVI::
002044  000000                   .WORD   0
002046                 L$EXP1::
002046  000000                   .WORD   0
002050                 L$MREV::
002050     003                   .BYTE   C$REVISION
002051     003                   .BYTE   C$EDIT
002052                 L$EF::
002052  000000                   .WORD   0
002054  000000                   .WORD   0
002056                 L$SPC::
002056  000000                   .WORD   0
002060                 L$DEVP::
002060  012026                   .WORD   L$DVTYP
002062                 L$REPP::
002062  024364                   .WORD   L$RPT
002064                 L$EXP4::
002064  000000                   .WORD   0
002066                 L$EXP5::
002066  000000                   .WORD   0
002070                 L$AUT::
002070  025316                   .WORD   L$AU
002072                 L$DUT::
002072  025310                   .WORD   L$DU
002074                 L$LUN::
002074  000000                   .WORD   0
002076                 L$DESP::
002076  012042                   .WORD   L$DESC
002100                 L$LOAD::
002100  104035                  EMT      E$LOAD
002102                 L$ETP::
002102  000000                   .WORD   0
002104                 L$ICP::
002104  024400                   .WORD   L$INIT
002106                 L$CCP::
002106  025266                   .WORD   L$CLEAN
002110                 L$ACP::
002110  025264                   .WORD   L$AUTO
002112                 L$PRT::
002112  024372                   .WORD   L$PROT
002114                 L$TEST::
002114  000000                   .WORD   0
002116                 L$DLY::
002116  000000                   .WORD   0
002120                 L$HIME::
002120  000000                   .WORD   0
```

```
                              .SBTTL  DISPATCH TABLE

                              ;++
                              ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
                              ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                              ;--

002122                                  DISPATCH 1
002122  000001                                                                  .WORD   1
002124                                                                   L$DISPATCH::
002124  025324                                                                  .WORD   T1

```

```
                              .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 HARDWARE P-TABLES,
                              ; AND IS USED AS A ''TEMPLATE'' FOR BUILDING THE P-TABLES.
                              ;--

002126                                BGNHW   DFPTBL
002126  000010                                                                          .WORD   L10000-L$HW/2
002130                                                                          L$HW::
002130                                                                          DFPTBL::


                              ;INDEPENDENT SECTION
                              ;       THE NUMBERS IN BRACKETS ARE THE OFFSET VALUES USED IN THE PARAMETER
                              ;       CODING SECTION.


002130  000001                        .WORD   1               ;[0] FULL OR HALF DUPLEX FLAG (BIT0=1 IF FULL)



                              ;DEVICE DEPENDENT SECTION
                              ;       ADDING OR REMOVING WORDS FROM THIS TABLE EFFECTS THE ''GET'' CALLS IN
                              ;       THE HARDWARE PARAMTER CODING SECTION BY CHANGING ''OFFSETS''

002132  160170                        .WORD   160170          ;[2] CSR ADDRESS
002134  000300                        .WORD   300             ;[4] INTERRUPT VECTOR
002136  000240                        .WORD   240             ;[6] INTERRUPT PRIORITY (5)
002140  000000                        .WORD   0               ;[10] DEVICE PARAMETERS WORD
                                                              ;  (ENABLE CRC, STRIP SYNC, COMPATIBLE MODE...)
002142  000000                        .WORD   0               ;[12] DEVICE OPTION TYPE(0=DMC,5=DMR-DMC MODE,
                                                              ; 7=DMR.
002144  000004                        .WORD   4               ;[14] BAUD RATE (0=2.4K, 1=4.8K, 2=9.6K, 3= 19.2K,
                                                              ; 4=56K, 5=250K, 6=500K, 7=1 MEGA-BAUD)
002146  000000                        .WORD   0               ;[16] LINE INTERFACE (422, V.35, INT, EIA...)


002150                                ENDHW
002150                                                                          L10000:
```

```




                                  .SBTTL  GLOBAL EQUATES SECTION




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


 002150                                   EQUALS
                                  ;
                                  ; BIT DIFINITIONS
                                  ;
              100000              BIT15== 100000
              040000              BIT14== 40000
              020000              BIT13== 20000
              010000              BIT12== 10000
              004000              BIT11== 4000
              002000              BIT10== 2000
              001000              BIT09== 1000
              000400              BIT08== 400
              000200              BIT07== 200
              000100              BIT06== 100
              000040              BIT05== 40
              000020              BIT04== 20
              000010              BIT03== 10
              000004              BIT02== 4
              000002              BIT01== 2
              000001              BIT00== 1
                                  ;
              001000              BIT9==  BIT09
              000400              BIT8==  BIT08
              000200              BIT7==  BIT07
              000100              BIT6==  BIT06
              000040              BIT5==  BIT05
              000020              BIT4==  BIT04
              000010              BIT3==  BIT03
              000004              BIT2==  BIT02
              000002              BIT1==  BIT01
              000001              BIT0==  BIT00
                                  ;
                                  ; EVENT FLAG DEFINITIONS
                                  ;    EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
                                  ;
              000040              EF.START==        32.                        ; START COMMAND WAS ISSUED
              000037              EF.RESTART==      31.                        ; RESTART COMMAND WAS ISSUED
              000036              EF.CONTINUE==     30.                        ; CONTINUE COMMAND WAS ISSUED
              000035              EF.NEW==          29.                        ; A NEW PASS HAS BEEN STARTED
              000034              EF.PWR==          28.                        ; A POWER-FAIL/POWER-UP OCCURRED
                                  ;
 1617                             ;
 1618                             ; PRIORITY LEVEL DEFINITIONS
 1619                             ;
 1620          000340             PRI07== 340
 1621          000300             PRI06== 300
 1622          000240             PRI05== 240
 1623          000200             PRI04== 200
 1624          000140             PRI03== 140
 1625          000100             PRI02== 100
 1626          000040             PRI01== 40
 1627          000000             PRI00== 0
 1628                             ;
 1629                             ;OPERATOR FLAG BITS
 1630                             ;
 1631          000004             EVL==        4
 1632          000010             LOT==       10
 1633          000020             ADR==       20
 1634          000040             IDU==       40
 1635          000100             ISR==      100
 1636          000200             UAM==      200
 1637          000400             BOE==      400
 1638          001000             PNT==     1000
 1639          002000             PRI==     2000
 1640          004000             IXE==     4000
 1641          010000             IBE==    10000
 1642          020000             IER==    20000
 1643          040000             LOE==    40000
 1644          100000             HOE==   100000
 1645
```

```
                                ;***** INDEPENDENT EQUATES

        001000                          BUFLIM=512.                             ;MAX BUFFER SIZE IN BYTES
                                                                                ;  APPLIES TO TX,RX AND CMP BUFFS
        000017                          MSGLIM=15.                              ;MAX NO. OF MESSAGES PER BUFFER
                                                                                ;  (FOR EACH INCREMENT (+1) TO MSGLIM,
                                                                                ;   ADD 6 WORDS TO THE POINTER TABLE
                                                                                ;   (PTRTAB:) SINCE THIS MEANS 2 MORE
                                                                                ;   'POINTER'' WORDS PER BUFFER.

                                ;MODE OF OPERATION EQUATES
        000000                          REC=0                                   ;RECEIVE MODE
        000001                          TRA=1                                   ;TRANSMIT MODE
        000002                          PAS=2                                   ;PASSIVE MODE
        000003                          ACT=3                                   ;ACTIVE MODE
        000004                          DOW=4                                   ;DOWN-LINE-LOAD MODE
        000005                          TAL=5                                   ;TALK MODE
        000006                          LIS=6                                   ;LISTEN MODE
                                ;MAINT LOOP TYPE EQUATES
        000000                          NONE=   0                               ;NO LOOP
        000001                          TTL=    1                               ;INTERNAL TTL
        000002                          CABLE=  2                               ;CABLE LOOP
        000003                          MODLOC= 3                               ;MODMEM LOCAL
        000004                          MODREM= 4                               ;MODEM REMOTE
        000005                          MOP=    5                               ;MOP



                                ;CLOCK ENABLE VALUES TO BE LOADED IN CLK'S CSR
        000100                          LCLKEN= 100                 ;L-CLOCK CSR VALUE TO ENABLE THE CLOCK
        000111                          PCLKEN= 111                 ;P-CLOCK CSR VALUE TO ENABLE THE CLOCK
        001600                          PCLKCT= 1600                ;P-CLOCK COUNT SET REGISTER FOR COUNTER

                                ;PARAM WORD EQUATES

        000001                          STATB=  BIT0                ;OPERATOR AWAKE ASKED FOR
        000002                          DATCKB= BIT1                ;DATA CHECK BIT
        000004                          ECHOB=  BIT2                ;ECHO BIT
        000020                          CRCB=   BIT4                ;CRC CALCUALTE ASKED FOR
        000040                          PROTOB= BIT5                ;PROTOCOL PROCESSING ASKED FOR

                                ;OPTION TYPE EQUATES

        000000                          DMC=    0                   ;DMC
        000004                          DMRC6=  4                   ;8206 DMR IN DMC MODE
        000005                          DMRC7=  5                   ;8207 DMR IN DMC MODE
        000006                          DMR6=   6                   ;8206 DMR IN DMR MODE
        000007                          DMR7=   7                   ;8207 DMR IN DMR MODE

                                ;EVENT LOG MESSAGE TYPES (USED TO LOCATE EVENT DESCRIPTION IN EVENT TABLE
                                ;  AND DISPATCHING TO SEPERATE SECTIONS OF THE EVENT REPORTING SECTION)
        000000                          TXQ=    0                   ;TRANSMIT MESSAGE QUEUED
        000002                          TXC=    2                   ;TRANSMIT COMPLETE
        000004                          RXQ=    4                   ;RECEIVE BUFFER QUEUED
        000006                          RXC=    6                   ;RECEIVE COMPLETE
        000010                          DER=    10                  ;DEVICE INFORMATION
        000012                          DVI=    12                  ;DEVICE ABOUT TO INIT
        000014                  DCK=    14              ;DATA COMPARISON RESULTS

        000020                  DLE=    20              ;DATA COMPARISON LENGH ERROR
        000022                  DDE=    22              ;DATA COMPARISON DATA ERROR
        000024                  EOP=    24              ;END OF PASS

                        ;;;;EQUATES FOR FLAG WORD;;;;;;

        000001                  ININT= 1                ;INPUT INT. REC.
        000002                  OTINT= 2                ;OUTPUT INT REC
        000004                  QRX=   4                ;RX QUED /COMPL
        000010                  QTX=    10              ;TX QUED/COMPL
        000020                  CTX=    20              ;TX COMPL AND IN TXSEL4 AND TSEL6
        000040                  CRX=    40              ;RX COMPL AND IN TSEL4 AND TSEL6
        000100                  ERX=    100             ;EXPECT TO GET A RX COMPLED
        000200                  ETX=    200             ;EXPECT TO GET A TX COMPLETED
        000400                  DLLGA=  400             ;DOWN LINE LOAD GO AHEAD BIT
        001000                  DMRRUN= 1000            ;DMR RUN MODE EXPECTED

                        ; SPECIAL CLI CODES FOR ''CHAR'' ARGUMENT IN CLI CALLS
                        ;   (COMMAND LINE INTERPRETER DEFINITIONS)
        000000                  CLIERR= 0
        000001                  CLIEXI= 1
        000002                  CLIBR=  2
        000003                  CLIBIF= 3
        000004                  CLISPA= 4
        000005                  CLINUM= 5
        000006                  CLIALP= 6
        000007                  CLIALN= 7
        000010                  CLIOCT= 8.
        000011                  CLIDEC= 9.
        000012                  CLISTR= 10.

                        ; DEFS FOR COMMAND LINE INTERPRETATION ACTION VALUES
        000000                  NULL=0
        000001                  CLEAR=1
        000002                  SHOW=2
        000003                  CHECK=3
        000004                  RUN=4
        000005                  HLP=5
        000006                  CSHEXP=6
        000007                  CSHTRN=7
        000010                  SETEXP=10
        000011                  SETTRN=11
        000012                  SIZE=12
        000013                  QCOPY=13
        000014                  NUM=14
        000015                  OPRMSG=15
        000016                  STATUS=16
        000017                  ENDQO=17
        000020                  CMSG0=20
        000021                  CMSG1=21
        000022                  CMSG2=22
        000023                  CMSG3=23
        000024                  CMSG4=24
        000025                  CMSG5=25
        000026                  CMSG6=26
        000027                  ATVMOD=27
        000030                  PASMOD=30
        000031                  RECMOD=31
        000032                  LISMOD=32
        000033                  DLLMOD=33
        000034                  TRAMOD=34
        000035                  TALMOD=35
        000036                  NO=36
        000037                  ECHO=37
        000040                  CRC=40
        000041                  PROTO=41
        000042                  PASC=42
        000043                  MOP=43
        000044                  TTLLOP=44
        000045                  CBLLOP=45
        000046                  LMDLOP=46
        000047                  RMDLOP=47
        000050                  NOTNUF=50
        000051                  BADCHR=51
        000052                  DMPS=52
        000053                  DMPE=53
        000054                  DMPQ=54
        000055                  PRNT=55


                        ;***** DEVICE DEPENDENT EQUATES
                        ; MODEM SIGNAL BIT DEFINITONS
                        ;       IF SIGNAL AVAILABLE IN DEVICE, EQUATE NAME TO BIT POSITION,
                        ;       ELSE EQUATE IT TO = 0

        000004                  CTS=    BIT2            ;CLEAR TO SEND (CIRCUIT CB)
        000010                  DSR=    BIT3            ;DATA SET READY (CIRCUIT CC)
        000001                  DCD=    BIT0            ;DATA CARRIER DETECT (CIRCUIT CF)
        000040                  RTS=    BIT5            ;REQUEST TO SEND (CIRCUIT CA)
        000200                  RI=     BIT7            ;RING INDICATOR (CIRCUIT CE)
        040000                  SQD=    BIT14           ;SIGNAL QUALITY DETECT (CIRCUIT CG)
        001000                  TM=     BIT9            ;MODEM IN TEST MODE (RS 449 ONLY CIRCUIT TM)



                        ; DEVICE SIGNALS
        000040                  RQI=    BIT5            ;REQUEST IN
        000200                  RDI=    BIT7            ;READY IN
        000200                  RDO=    BIT7
        000001                  BACC=   BIT0            ;BUFFER ADDR. CHAR COUNT
        040000                  MCLR=   BIT14           ;MASTER CLEAR
        004000                  LULOOP= BIT11           ;LINE UNIT LOOP(TTL)
        000400                  MAINTB= BIT8            ;MAINT MODE BIT
        002000                  HALFDB= BIT10           ;HALF DUPLEX BIT
        000004                  RXBIT=  BIT2            ;RX BIT
        000100                  IEO=    BIT6            ;ENABLE OUTPUT INTERRUPT BIT

```

```
                                .SBTTL  GLOBAL DATA SECTION
                                .SBTTL          DEFAULT MESSAGE DEFINITIONS AND TABLES

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

                                ;MESSAGE BYTE COUNT TABLE

002150                          DMSGCT:
002150  000001                  MSG0C:  .WORD   EMSG0-MSG0      ;BYTE COUNT OF MESSAGE #0
002152  000001                  MSG1C:  .WORD   EMSG1-MSG1      ;BYTE COUNT OF MESSAGE #1
002154  000001                  MSG2C:  .WORD   EMSG2-MSG2      ;BYTE COUNT OF MESSAGE #2
002156  000001                  MSG3C:  .WORD   EMSG3-MSG3      ;BYTE COUNT OF MESSAGE #3
002160  000100                  MSG4C:  .WORD   EMSG4-MSG4      ;BYTE COUNT OF MESSAGE #4
002162  000072                  MSG5C:  .WORD   EMSG5-MSG5      ;BYTE COUNT OF MESSAGE #5
002164  000101                  MSG6C:  .WORD   EMSG6-MSG6      ;BYTE COUNT OF MESSAGE #6
002166  000000                  OPCNT:  .WORD   0               ;BYTE COUNT FOR OPERATOR SPEC'D MSG.
002170  000001                  MSG8C:  .WORD   EMSG8-MSG8      ;BYTE COUNT OF RECEIVE BUFFER FILL PATTERN
002172  000005                  DLLM1C: .WORD   DLLM1E-DLLM1    ;DLL MSG 1 COUNT
002174  000206                  DLLM2C: .WORD   DLLM2E-DLLM2    ;DLL MSG 2 COUNT

                                ;MESSAGE ADDRESS TABLE

002176                          DMSGAD:
002176  002220                          MSG0                    ;ADDRESS OF MESSAGE #0
002200  002221                          MSG1                    ;ADDRESS OF MESSAGE #1
002202  002222                          MSG2                    ;ADDRESS OF MESSAGE #2
002204  002223                          MSG3                    ;ADDRESS OF MESSAGE #3
002206  002224                          MSG4                    ;ADDRESS OF MESSAGE #4
002210  002324                          MSG5                    ;ADDRESS OF MESSAGE #5
002212  002416                          MSG6                    ;ADDRESS OF MESSAGE #6
002214  002524                          OPBUF                   ;ADDRESS OF OPERATOR SPEC'D MSG.
002216  002646                          MSG8                    ;ADDRESS OF RECEIVE BUFFER FILL PATTERN

002220     000                  MSG0:   .BYTE   000             ;MESSAGE OF ALL 0'S
002221                          EMSG0:
002221     377                  MSG1:   .BYTE   377             ;MESSAGE OF ALL 1'S
002222                          EMSG1:
002222     252                  MSG2:   .BYTE   252             ;MESSAGE OF ALTERNATING 1'S
002223                          EMSG2:
002223     125                  MSG3:   .BYTE   125             ;MESSAGE OF ALTERNATING 0'S
002224                          EMSG3:
002224                          MSG4:                           ;''CCITT'' 512-BIT (VS. 511 BITS) TEST PATTERN
002224  177603  157427  031011          .WORD   177603,157427,031011,047321,163715,105221,143325,142304
002232  047321  163715  105221
002240  143325  142304
002244  040041  014116  052606          .WORD   040041,014116,052606,172334,105025,123754,111337,111523
002252  172334  105025  123754
002260  111337  111523
002264  030030  145064  137642          .WORD   030030,145064,137642,143531,063617,135075,066730,026575
002272  143531  063617  135075
002300  066730  026575
002304  052012  053627  070071          .WORD   052012,053627,070071,151172,165044,031605,166632,016741
002312  151172  165044  031605
```

```
002320  166632  016741
002324                            EMSG4:
002324                            MSG5:                             ;''INTERPROCESSOR TEST PROGRAM'S (ITEP)'' MESSAGE
                                                                    ; #1, (DP1:)
002324  077577  040444  052040            .ASCII  <177><177>/$A THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG./
002332  042510  050440  044525
002340  045503  041040  047522
002346  047127  043040  054117
002354  045040  046525  042520
002362  020104  053117  051105
002370  052040  042510  046040
002376  055101  020131  047504
002404  027107
002406  005015  077401  077577            .ASCIZ  <15><12><001><177><177><177><177>
002414  000177
002416                            EMSG5:
002416                            MSG6:                             ;ALPHA-NUMERICS (OR FUTURE COMM TURNAROUND MSG)
002416  022043  021041  023040            .ASCII  /#$!'' &'()*+,-.0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ/
002424  024047  025051  026053
002432  027055  030460  031462
002440  032464  033466  034470
002446  035472  036474  037476
002454  040500  041502  042504
002462  043506  044510  045512
002470  046514  047516  050520
002476  051522  052524  053526
002504  054530     132
002507     057  056133  057135            .ASCIZ  ?/[\]^_%?
002514  022537     000
002517                            EMSG6:
        002520                            .EVEN

                                  ; ***********************************************************************
                                  ;THESE THREE STORAGE AREAS MUST NOT BE SEPERATED !!!!

002520  047045  040445            OPBFPT: .ASCII  /%N%A/
002524  000122                    OPBUF:  .BLKB   82.               ;BUFFER FOR OPERATOR SPEC'D MESSAGES
002646                            OPEND:

                                  ; THE ABOVE THREE LINES MUST BE KEPT TOGETHER
                                  ; ***********************************************************************

002646     033                    MSG8:   .BYTE   33                ;RECEIVE BUFFER FILL PATTERN
002647                            EMSG8:

                                  ; DOWN-LINE-LOAD MESSAGE DEFINITIONS

002647     006                    DLLM1:  .BYTE   6
002650     000                            .BYTE   0
002651     000                            .BYTE   0
002652     000                            .BYTE   0
002653     000                            .BYTE   0
002654                            DLLM1E:
002654     000                    DLLM2:  .BYTE   0         ;CODE
002655     000                            .BYTE   0         ;LOAD NUMBER
002656     006                            .BYTE   6         ;LOAD ADDRESS LSB
```

```
 002657      000                          .BYTE   0
 002660      006                          .BYTE   6
 002661      000                          .BYTE   0               ;LOAD ADDRESS
                                  ;
                                  ;IMAGE DATA
                                  ;
 002662  005037  000006                   CLR     @#6
 002666  000005                           RESET
 002670  012706  001000                   MOV     #1000,SP
 002674  012701  177560                   MOV     #177560,R1         ;SET UP TTY
 002700  010700                           MOV     PC,R0              ;MAKE ADDR.PIC
 002702  062700  000034                   ADD     #<MSG-.>,R0        ;ADDRESS MSG.
 002706  105761  000004           1$:     TSTB    4(R1)              ;TTY READY?
 002712  100375                           BPL     1$                 ;WAIT TIL YES
 002714  112061  000006                   MOVB    (R0)+,6(R1)        ;TYPE A CHAR
 002720  001372                           BNE     1$                 ;KEEP GOING
 002722  012737  000026  000024           MOV     #26,@#24           ;SET UP POWER FAIL
 002730  005037  000026                   CLR     @#26               ;MAKE SURE T BIT CLAER
 002734  000777                           BR      .                  ;JUMP ON YOURSELF
 002736  006412  047502  052117   MSG:    .ASCII  <12><15>/BOOT MESSAGE WAS RECEIVED SUCCESSFULLY -END OF TEST!!/
 002744  046440  051505  040523
 002752  042507  053440  051501
 002760  051040  041505  044505
 002766  042526  020104  052523
 002774  041503  051505  043123
 003002  046125  054514  026440
 003010  047105  020104  043117
 003016  052040  051505  020524
 003024     041
 003025     012  027015  027056           .ASCIZ  <12><15>/....RELOAD PROGRAM..../
 003032  051056  046105  040517
 003040  020104  051120  043517
 003046  040522  027115  027056
 003054  000056
 003056     006                           .BYTE   6
 003057     000                           .BYTE   0
 003060     000                           .BYTE   0
 003061     000                           .BYTE   0
 003062                           DLLM2E:
                                          .EVEN
```

```
                                        ;COMMAND LINE BUFFER, DATA LOCATIONS AND MESSAGES FOR ACTION ROUTINES

003062  000122                          CMDBUF: .BLKB   82.             ;BUFFER FOR OPERATOR COMMANDS
003204  000000                          KEYWD1: .WORD   0               ;THIS LOC WILL =1 IF CLEAR TYPED, 2 FOR SHOW,
                                                                        ; A 4 IF RUN WAS TYPED, 5 IF HELP WAS TYPED
003206  000000                          QUALFG: .WORD   0               ;THIS LOC HOLDS QUALIFIER VALUE (SIZE OR COPY)
003210  000000                          QUALVL: .WORD   0
003212  012525                          HLPTAB: .WORD   HLP1
003214  012540                                  .WORD   HLP2
003216  012646                                  .WORD   HLP3
003220  012733                                  .WORD   HLP4
003222  013012                                  .WORD   HLP4A
003224  013070                                  .WORD   HLP5
003226  013152                                  .WORD   HLP6
003230                                  HLPEND:
003230  013305  013314  013321          SHTYTB: .WORD   SHTYP0,SHTYP1,SHTYP2,SHTYP3,SHTYP4,SHTYP5,SHTYP6,SHTYP7
003236  013326  013333  013341
003244  013346  013354

                                        ; THE LIST OF BYTES BELOW ARE THE FIRST BYTES OF THE PREDEFINED MESSAGES
                                        ;  USED TO 'SHOW' THE TRANSMIT AND COMPARE BUFFER CONTENTS.

003250     000     377     252          SHTAB:  .BYTE   0,377,252,125,203,177,043
003253     125     203     177
003256     043
003257                                  SHTEND:
        003260                                  .EVEN

003260  013365                          MODES:  .WORD   M00     ;ADDRESSES OF MODE TYPES IN ASCII
003262  013375                                  .WORD   M01
003264  013406                                  .WORD   M02
003266  013416                                  .WORD   M03
003270  013425                                  .WORD   M04
003272  013442                                  .WORD   M05
003274  013447                                  .WORD   M06

003276  013456                          LOOPS:  .WORD   LP0     ;ADDRESSES OF LOOP TYPES IN ASCII
003300  013466                                  .WORD   LP1
003302  013477                                  .WORD   LP2
003304  013505                                  .WORD   LP3
003306  013520                                  .WORD   LP4

                                        ;COMMAND LINE TRAVERSE LOCATIONS (USED BY 'P$TRV')

003310  000000                          P$BUFA: .WORD   0               ;LOC. TO HOLD ADDR. OF CMD LINE BUFFER
003312  000000                          P$TREE: .WORD   0               ;LOC. TO HOLD ADDR. OF PARSING TREE
003314  000000                          P$ACT:  .WORD   0               ;LOC. TO HOLD ADDR. OF ACTION ROUTINE
003316  000000                          P$CNT:  .WORD   0               ;LOC. TO BE A COUNTER LOCATION
003320  000000                          P$NUM:  .WORD   0               ;LOC. TO HOLD NUMERIC VALUE FROM PARSE
003322  000000                          P$RADX: .WORD   0               ;LOC. TO HOLD RADIX USED(LO) AND +/-(HI BYTE)
003324     000                          P$NNUF: .BYTE   0               ;RETURN =0 IF ENOUGH OF COMMAND FOUND
003325     000                          P$GDBD: .BYTE   0               ;RETURN CODE 0 IF NO ERROR FOUND

```

```
                                .SBTTL          MESSAGE BUFFERS AND POINTER TABLES

        003326  001000          TXBUF:  .BLKB   BUFLIM  ;TRANSMITTER BUFFERS
        004326  001000          RXBUF:  .BLKB   BUFLIM  ;RECEIVER BUFFERS
        005326  001000          CMPBUF: .BLKB   BUFLIM  ;COMPARISON BUFFERS
        006326  000264          PTRTAB: .BLKW   180.    ;TABLE FOR MESSAGE ADDRS. & BYTE COUNTS
        007076                  PTREND:                 ; END OF MSG. PTR. TABLE

        007076  000000          RXPTR:  .WORD   0       ;RECEIVER MESSAGE POINTER
        007100  000000          TXPTR:  .WORD   0       ;TRANMITTER BUFFER POINTER
        007102  000000          CMPPTR: .WORD   0       ;COMPARISON BUFFER POINTER
        007104  000000          CMPTOT: .WORD   0       ;CMP MSG TOTAL
        007106  000000          CTOTCC: .WORD   0       ;COMPARE BUFFER CHAR. COUNT
        007110  000000          CCURAD: .WORD   0       ;CURRENT ADDR OF CMP BUFF TO ADD AT

        007112  000000          DVTXA:  .WORD   0       ;DEVICE TX ADDR
        007114  000000          DVTCC:  .WORD   0       ;DEVICE TX CHAR COUNT
        007116  000000          DVTCT:  .WORD   0       ;DEVICE TX MESSAGE COUNT
        007120  000000          TXMTOT: .WORD   0       ;TX MSG TOTAL
        007122  000000          TTOTCC: .WORD   0       ;TX BUFFER CHAR. COUNT
        007124  000000          TCURAD: .WORD   0       ;CURRENT ADDR. OF TX BUFF TO ADD AT

        007126  000000          DVRXA:  .WORD   0       ;DEVICE RX ADDR
        007130  000000          DVRCC:  .WORD   0       ;DEVICE RX CHAR COUNT
        007132  000000          DVRCT:  .WORD   0       ;DEVICE RX MESSAGE COUNT
        007134  000000          RXMTOT: .WORD   0       ;RX MSG TOTAL

        007136  000000          LNCNT:  .WORD   0       ;NUMBER OF OPERATOR AWAKE MSGS
        007140  000000          NOBUF:  .WORD   0       ;NUMBER OF NO BUFFS
        007142  000000          PSCNT:  .WORD   0       ;PASS COUNTER
        007144  000000          ERRCNT: .WORD   0       ;ERROR COUNTER
        007146  000000          STADD:  .WORD   0       ;START ADDR.
        007150  000000          ENADD:  .WORD   0       ;END ADDR. FOR DUMP
        007152  000000          BYTBIT: .WORD   0       ;BYTE BIT FOR DUMP ROUTINE

                                ;OTHER MESSSAGE RELATED STORAGE LOCATIONS

        007154  000000          MSGTYP: .WORD   0       ;TYPE OF DATA 0=0'S,1=1'S,2=10'S,3=01'S
                                                                ;4=CCITT,5=QUICK FOX,6=ALPHA/NUM,7=OPER
        007156  000000          CURCC:  .WORD   0       ;TX/RX/CMP CHAR COUNT
        007160  000000          CPTRR:  .WORD   0       ;CURRENT RX POINTER
        007162  000000          CPTR:   .WORD   0       ;CURRENT POINTER
        007164  000000          CURADD: .WORD   0       ;CURRENT TX/RX/CMP START ADDD
        007166  000000          TOTCC:  .WORD   0       ;TOTAL CHAR COUNT NOT MORE THEN 'BUFLIM''
        007170  000000          OFSET:  .WORD   0       ;OFFSET COUNT
        007172  000000          TEMP:   .WORD   0       ;TEMPORARY LOCATIONS (USED A LOT)
        007174  000000          TEMP1:  .WORD   0
        007176  000000          TEMP2:  .WORD   0
        007200  000000          TEMP3:  .WORD   0
        007202  000000          TEMP4:  .WORD   0
        007204  000000          CONOTM: .WORD   0       ;CONTROL OUT ERROR MSG. ADDRESS
        007206  000000          CONTIN: .WORD   0       ;WORD FOR CONTORL IN
        007210     000          GOOD:   .BYTE   0       ;BYTE TO HOLD EXPECTED MESSAGE DATA BYTE FOR ERR REPORT
        007211     000          BAD:    .BYTE   0       ;BYTE TO HOLD RECEIVED MESSAGE DATA BYTE FOR ERR REPORT
```

```
                                ;MORE INDEPENDENT CODE STORAGE LOCATIONS

 007212  000000                 LOGUNT: .WORD   0       ;LOC. TO HOLD LOGICAL UNIT NUMBER
 007214  000000                 PCADD:  .WORD   0       ;LOC. HOLD PC OF CALLIN ROUTINE
 007216  000000                 RESFLG: .WORD   0       ;LOC TO HOLD FLAG (-1) THAT A RESTART WAS GIVEN
 007220  000000                 MODTYP: .WORD   0       ;DCLT MODE OF OPERATION TYPE
                                                        ;  (0=REC-ONLY, 1=TX-ONLY, 2=PASSIVE-LOOPBK,
                                                        ;   3=ACTIVE-LOOPBK, 4=DOWN L.L., 5=TALK, 6=LISTEN)
 007222  000000                 MLTYP:  .WORD   0       ;MAINTENANCE LOOP TYPE (0=NONE, 1=INTERNAL TTL,
                                                        ;  2=CABLE, 3=MODEM-ANALOG LOOPBK (LOCAL),
                                                        ;  4=MODEM-DIGITAL LOOPBK (REMOTE), 5=MOP)
 007224  000000                 FHDPLX: .WORD   0       ;FULL OR HALF DUPLEX FLAG (1=FULL FROM P-TABLE)
 007226  000002                 PARAM:  .WORD   2       ;PROGRAM PARAMETERS
                                                        ; BIT0= STATUS MSGS TO OPR PRINTED (1=YES)
                                                        ; BIT1= DATA CHECKING DONE ON RCVD MSGS (1=YES)
                                                        ; BIT2= ECHO (TRANSMIT) RCV'D MSG.(PASSIVE)(1=YES)
                                                        ; BIT3= SPARE
                                                        ; BIT4= CRC CALC./CHECK DONE (1=YES)
                                                        ; BIT5= PROTOCOL EMULATION (1=YES)
                                                        ; BIT6= SPARE
 007230  000000                 RPASS:  .WORD   0       ;PASS NUMBER FROM RUN COMMAND
 007232  000000                 FLAG:   .WORD   0       ;DEVICE FLAG WORD

                                ;MODE DISPATCH TABLE
 007234  031014                 MODE:   .WORD   RXONLY  ;RX ONLY DISPATCH
 007236  031046                         .WORD   TXONLY  ;TX ONLY DISPATCH
 007240  031106                         .WORD   PLCK    ;PASSIVE LOOP BACK DISP
 007242  031142                         .WORD   ALCK    ;ACTIVE LOOP BACK DISP
 007244  032262                         .WORD   DLL     ;DOWN LINE LOAD DISP
 007246  032750                         .WORD   TALCK   ;TALK MODE DISPATCH
 007250  033170                         .WORD   LISCK   ;LISTEN MODE DISPATCH


                                .SBTTL          CLOCK TABLES, EVENT LOG AND POINTERS
 007252  000000                 CLKCSR: .WORD   0       ;CLOCK CSR ADDRESS
 007254  000000                 CLKBR:  .WORD   0       ;CLOCK INTERRUPT LEVEL
 007256  000000                 CLKVEC: .WORD   0       ;CLOCK INTERRUPT VECTOR
 007260  000074                 CLKHZ:  .WORD   60.     ;CLOCK'S HERTZ RATE
 007262  000000                 CLKEN:  .WORD   0       ;CLOCK'S CSR VALUE TO INTRPT. ENABLE IT

 007264  000000                 TIMMIN: .WORD   0       ;PLACE TO KEEP TIME-SINCE-START
 007266  000000                 TIMSEC: .WORD   0
 007270  000000                 TIMTCK: .WORD   0       ;PLACE TO KEEP # OF TICKS/SEC

 007272  000000                 TIMER1: .WORD   0       ;EVENT TIMER #1 (TICKS)
 007274  000000                 TIMER2: .WORD   0       ;EVENT TIMER #2 (TICKS)
 007276  000000                 TIMERS: .WORD   0       ;EVENT TIMER #3 (SECONDS)
```

```
                                ;EVENT LOG TABLE AND ITS NEXT ENTRY POINTER
007300  007302                  EVTPTR: .WORD   EVTLOG  ;POINTER TO NEXT FREE SPACE IN EVENT LOG
007302  000341                  EVTLOG: .BLKW   225.    ;EVENT LOG BUFFER
010204  000001                  EVTEND: .BLKW   1.      ;APPROXIMATE END OF EVENT TABLE (ALLOWS CIRCULAR QUE)

                                .SBTTL          MODEM DATA SECTION

010206  000000                  MODS:   .WORD   0       ;MODEM STATUS

                                ;TABLE OF MODEM SIGNAL BIT DEFINITIONS

010210  000004                  MOBITS: .WORD   CTS                     ;CLEAR TO SEND (CIRCUIT CB)
010212  000010                          .WORD   DSR                     ;DATA SET READY (CIRCUIT CC)
010214  000001                          .WORD   DCD                     ;DATA CARRIER DETECT (CIRCUIT CF)
010216  000040                          .WORD   RTS                     ;REQUEST TO SEND (CIRCUIT CA)
010220  000200                          .WORD   RI                      ;RING INDICATOR (CIRCUIT CE)
010222  040000                          .WORD   SQD                     ;SIGNAL QUALITY DETECT (CIRCUIT CG)
010224  001000                          .WORD   TM                      ;MODEM IN TEST MODE (RS 449 ONLY CIRCUIT TM)
010226                          MOBITE:

                                ;TABLE OF ADDRESSES OF MODEM SIGNAL MESSAGE POSITIONS

010226  016004                  MOMSGS: .WORD   EVMCTS                  ;CLEAR TO SEND (CIRCUIT CB)
010230  016010                          .WORD   EVMDSR                  ;DATA SET READY (CIRCUIT CC)
010232  016014                          .WORD   EVMDCD                  ;DATA CARRIER DETECT (CIRCUIT CF)
010234  016020                          .WORD   EVMRTS                  ;REQUEST TO SEND (CIRCUIT CA)
010236  016024                          .WORD   EVMRI                   ;RING INDICATOR (CIRCUIT CE)
010240  016030                          .WORD   EVMSQD                  ;SIGNAL QUALITY DETECT (CIRCUIT CG)
010242  016034                          .WORD   EVMTM                   ;MODEM IN TEST MODE (RS 449 ONLY CIRCUIT TM)

                                ;TABLE OF ADDRESSES OF EVENT DESCRIPTION MESSAGES
                                ;       ORDER CORRESPONDS TO MESSAGE TYPE VALUES

010244  014430                  EVTLST: .WORD   EDTXQ   ;TRANSMIT MESSAGE QUEUED
010246  014454                          .WORD   EDTXC   ;TRANSMIT OF MESSAGE COMPLETE
010250  014503                          .WORD   EDRXQ   ;RECEIVE MESSAGE SPACE QUEUED
010252  014530                          .WORD   EDRXC   ;MESSAGE RECEIVED - RECEIVE COMPLETE
010254  014556                          .WORD   EDDER   ;DEVICE INFORMATION
010256  014623                          .WORD   EDDVI   ;DEVICE INITIALIZE STARTED
010260  014573                          .WORD   EDDCK   ;DATA COMPARISON DONE
010262  013456                          .WORD   LP0     ;NULL STRING
010264  014651                          .WORD   EDDLE   ;DATA COMPARE LENGTH ERROR
010266  014706                          .WORD   EDDDE   ;DATA COMPARE DATA ERROR
010270  014741                          .WORD   EDEOP   ;END OF PASS

                                ;LOCATIONS USED DURING EVENT REPORTING

010272  000000                  EVTSEC: .WORD   0       ;TEMPORARY LOCS TO KEEP EVENT TIME WHILE REPORTING
010274  000000                  EVTMIN: .WORD   0
010276  000000                  EVTTCK: .WORD   0
010300  000000                  EVTADD: .WORD   0       ;TEMP. LOC. TO HOLD ADDRESS DURING EVENT REPORTING
010302  000000                  EVTBCT: .WORD   0       ;       "       "  BYTE COUNT  "      "      "
010304  000000                  EVTTMP: .WORD   0       ;       "       "  OTHER DATA  "      "      "

                                ;REPORT CODING DISPATCH TABLE

```

```
010306  021702          RPTDSP: .WORD   RPTTXQ  ;TRANSMIT QUEUED ENTRY DECODING
010310  021702                  .WORD   RPTTXQ  ;TRANSMIT COMPLETE ENTRY DECODING
010312  021702                  .WORD   RPTTXQ  ;RECEIVER QUEUED ENTRY DECODING
010314  021702                  .WORD   RPTTXQ  ;RECEIVER COMPLETE ENTRY DECODING
010316  021754                  .WORD   RPTDER  ;DEVICE ERROR ENTRY DECODING
010320  022050                  .WORD   RPTDVI  ;DEVICE INIT ENTRY DECODING
010322  022244                  .WORD   RPTDCK  ;DATA COMPARISON ENTRY DECODING
010324  021530                  .WORD   RPT     ;PLACE HOLDER
010326  022244                  .WORD   RPTDLE  ;DATA COMPARISON LENGH ERROR
010330  022170                  .WORD   RPTDDE  ;DATA COMPARISON DATA ERROR
010332  022114                  .WORD   RPTEOP  ;END OF PASS


010334  000000          DEV1:   .WORD   0               ;TEMP LOCS TO HOLD DATA FOR EVENT REPORTING
010336  000000          DEV2:   .WORD   0               ;  AND SHOW MODE,... SUBROUTINE
010340  000000          DEV3:   .WORD   0
010342  000000          DEV4:   .WORD   0

```

```
                                   .SBTTL          COMMAND LINE ACTION TREE

                                   ;SAMPLE CLI TREE NODE   (ALWAYS AT LEAST 1 WORD)
                                   ;  ----------------------
                                   ;  ! ACTION  ! CHAR CODE !
                                   ;  ----------------------
                                   ;  !  MISS DISPLACEMENT  !          ONLY IF 'MISS'' ARGUMENT DEFINED
                                   ;  ----------------------
                                   ;  ! NEXT NODE DISPLMNT  !          ONLY IF ''ASCII'' ARGUMENT DEFINED
                                   ;  ----------------------
                                   ;  ! ASCIZ MATCH STRING  !          ONLY IF ''ASCII'' ARGUMENT DEFINED
                                   ;  !             (.EVEN) !
                                   ;  ----------------------


        010344                     CLITRE:

                                   ;FIRST KEYWORD
        010344                             CLI     CLISPA,0,N10$                       ;SKIP ANY LEADIN SPACES
        010350                     N10$:   CLI     <'?>,HLP,N42$                       ;IS THE FIRST NON-SP CHAR A '?''
        010354                             CLI     CLIEXI,0                            ;   IF YES DO 'HLP'' AND EXIT
        010356                     N42$:   CLI     CLISTR,HLP,N43$,<'HELP'>            ;ELSE, IS FIRST WORD A 'HELP''
        010372                             CLI     CLIEXI,0                            ;   IF YES DO 'HLP'' AND EXIT
        010374                     N43$:   CLI     CLISTR,PRNT,N45$,<'PRINT'>          ;ELSE, IS FIRST WORD A 'PRINT''
        010410                             CLI     CLIEXI,0                            ;   IF YES DO 'PRINT'' AND EXIT
        010412                     N45$:   CLI     CLISTR,RUN,N46$,<'RUN'>             ;ELSE, IS FIRST WORD A 'RUN''
        010424                             CLI     CLIBR,0,N80$                        ;   IF YES DO 'RUN'' & GOTO N80$
        010430                     N46$:   CLI     CLISTR,NOTNUF,N40$,<'DUMP'>         ;ELSE, IS FIRST WORD A 'DUMP''
        010444                             CLI     CLIBR,0,N50$                        ;   IF YES GOTO N80$
        010450                     N40$:   CLI     CLISTR,CLEAR,N20$,<'CLEAR'>         ;ELSE, IS FIRST WORD A ''CLEAR''
        010464                             CLI     CLIBR,NOTNUF,N100$                  ;   IF YES DO ''CLR'' & GOTO N100$
        010470                     N20$:   CLI     <'S>,NOTNUF,N30$                    ;ELSE, IS FIRST CHAR. A 'S''
        010474                             CLI     CLISTR,SHOW,N25$,<'HOW'>            ;   IF YES IS REST OF WORD 'HOW''
        010506                             CLI     CLIBR,0,N100$                       ;      IF YES, DO 'SHOW'',BR N100$
        010512                     N25$:   CLI     CLISTR,0,N30$,<'ET'>                ;   ELSE, IS REST OF WORD 'ET''
        010524                             CLI     CLIBR,0,N110$                       ;      IF YES, DO 'SET'', BR N110$
        010530                     N30$:   CLI     CLIERR,0                            ;OTHERWISE ''ILL CMD'' - EXIT

                                   ;SECOND KEYWORD (MODE=) FOR RUN COMMAND

        010532                     N80$:   CLI     CLISPA,0,N30$                       ;SKIP LEADING SPS, IF NONE-ERR
        010536                     N81$:   CLI     CLISTR,NOTNUF,N30$,<'MODE'>         ;IS NEXT WORD 'MODE=''
        010552                             CLI     <'=>,0,N30$                         ;   IF NO, IT'S WRONG -ERR -EXIT
        010556                             CLI     CLISTR,ATVMOD,N82$,<'ACTIVE'>       ;IS NEXT WORD 'ACTIVE''
        010574                             CLI     CLIBR,0,N115$                       ;   IF YES, DO 'ACTIVE'',BR N115$
        010600                     N82$:   CLI     CLISTR,PASMOD,N83$,<'PASSIVE'>      ;IS NEXT WORD 'PASSIVE''
        010616                             CLI     CLIBR,0,N115$                       ;   IF YES, DO 'PASSVE'',BR N115$
        010622                     N83$:   CLI     CLISTR,RECMOD,N84$,<'RECEIVE'>      ;IS NEXT WORD 'RECEIVE''
        010640                             CLI     CLIBR,0,N115$                       ;   IF YES, DO 'RECVE'',BR N115$
        010644                     N84$:   CLI     CLISTR,LISMOD,N85$,<'LISTEN'>       ;IS NEXT WORD 'LISTEN''
        010662                             CLI     CLIBR,0,N115$                       ;   IF YES, DO 'LISTEN'',BR N115$
        010666                     N85$:   CLI     CLISTR,DLLMOD,N86$,<'DOWNLINELOAD'>     ;IS NEXT WORD 'DOW...''
        010712                             CLI     CLIBR,0,N115$                       ;   IF YES, DO 'DWNLL'',BR N115$
        010716                     N86$:   CLI     <'T>,0,N30$                         ;IS NEXT CHAR A 'T''
        010722                             CLI     CLISTR,TRAMOD,N87$,<'RANSMIT'>      ; IS REST OF WORD 'RANSMIT''
        010740                             CLI     CLIBR,0,N115$                       ;   IF YES, DO 'TRANSM'',BR N115$
```

```
010744                  N87$:   CLI     CLISTR,TALMOD,N30$,<'ALK'>          ; IS REST OF WORD ''ALK''
010756                          CLI     CLIBR,0,N115$                       ;  IF YES, DO ''TALK'',BR N115$
                                                                            ; IF NO, ERROR - EXIT

                        ;SECOND KEYWORD (FOR CLEAR OR SHOW)
010762                  N100$:  CLI     CLISPA,0,N30$                       ;SKIP LEADING SPACES, NONE=ERR
010766                  N102$:  CLI     CLISTR,CSHEXP,N104$,<'EXPECTBUFF'>           ;IS NEXT WORD 'EXPE...''
011010                          CLI     CLIEXI,0                                     ; IF YES, DO CLR-EXP,EXIT
011012                  N104$:  CLI     CLISTR,CSHTRN,N30$,<'TRANSMITBUFF'>          ;IS NEXT WORD ''TRANS..''
011036                          CLI     CLIEXI,0                                     ; IF YES, DO CLR-TRN,EXIT
                                                                                     ;IF NO - ERROR - EXIT


                        ;SECOND KEYWORD (FOR SET)
011040                  N110$:  CLI     CLISPA,0,N30$
011044                  N111$:  CLI     CLISTR,SETEXP,N112$,<'EXPECTMSG'>
011064                          CLI     CLIBR,0,N120$
011070                  N112$:  CLI     CLISTR,SETTRN,N30$,<'TRANSMITMSG'>
011112                          CLI     CLIBR,0,N120$

                        ;GET ADDRESSES FOR DUMP COMMAND
011116                  N50$:   CLI     CLIALP,0,N51$
011122                  N51$:   CLI     CLISPA,0,N52$
011126                  N52$:   CLI     CLIOCT,DMPS,N30$
011132                          CLI     <'->,NOTNUF,N125$
011136                          CLI     CLIOCT,DMPE,N30$
011142                          CLI     <'/>,NOTNUF,N125$
011146                          CLI     <'B>,DMPQ,N30$
011152                          CLI     CLIBR,0,N125$

                        ;QUALIFIERS FOR THE RUN COMMAND
011156                  N115$:  CLI     CLIALP,0,N114$
011162                  N114$:  CLI     <'/>,NOTNUF,N125$
011166                          CLI     CLISTR,NO,N116$,<'NO'>
011200                  N116$:  CLI     <'C>,0,N117$
011204                          CLI     CLISTR,CHECK,N117$,<'HECK'>
011220                          CLI     CLIBR,0,N115$


                        ;N113$: CLI     CLISTR,CRC,N30$,<'RC16'>
                        ;       CLI     CLIBR,0,N115$

011224                  N117$:  CLI     CLISTR,STATUS,N118$,<'STATUS'>
011242                          CLI     CLIBR,0,N115$
011246                  N118$:  CLI     CLISTR,ECHO,N130$,<'ECHO'>
011262                          CLI     CLIBR,0,N115$


011266                  N130$:  CLI     CLISTR,0,N131$,<'PASS'>
011302                          CLI     CLIBR,0,N150$
011306                  N131$:  CLI     CLISTR,0,N30$,<'LOOP'>
011322                          CLI     CLIBR,0,N140$

                        ;GET MESSAGE TYPE FOR SET MESSAGE COMMANDS
011326                  N120$:  CLI     <'=>,0,N30$

```

```
                        ;     LOOK FOR DEFAULT MESSAGE NAME
011332                  N60$:     CLI       CLISTR,CMSG1,N61$,<'ONES'>
011346                            CLI       CLIBR,0,N121$
011352                  N61$:     CLI       CLISTR,CMSG0,N62$,<'ZEROES'>
011370                            CLI       CLIBR,0,N121$
011374                  N62$:     CLI       CLISTR,CMSG2,N63$,<'1ALT'>
011410                            CLI       CLIBR,0,N121$
011414                  N63$:     CLI       CLISTR,CMSG3,N64$,<'0ALT'>
011430                            CLI       CLIBR,0,N121$
011434                  N64$:     CLI       CLISTR,CMSG5,N65$,<'ITEP'>
011450                            CLI       CLIBR,0,N121$
011454                  N65$:     CLI       CLISTR,CMSG4,N66$,<'CCITT'>
011470                            CLI       CLIBR,0,N121$
011474                  N66$:     CLI       CLISTR,CMSG6,N67$,<'ALPHA'>
011510                            CLI       CLIBR,0,N121$


                        ;     LOOK FOR QUOTED MESSAGE
011514                  N67$:     CLI       <'">,OPRMSG,N30$
011520                  N70$:     CLI       <'">,ENDQO,N71$
011524                            CLI       CLIBR,0,N121$
011530                  N71$:     CLI       CLISPA,0,N72$
011534                  N72$:     CLI       CLIALN,0,N73$               ;ONLY A-Z,SP,TAB, OR 0-9 BETWEEN '"S
011540                            CLI       CLIBR,0,N70$
011544                  N73$:     CLI       CLIERR,BADCHR               ;PRINT ERROR IF NONE LEGAL CHAR FOR '"S

                        ;GET QUALIFIERS (SIZE OR COPY) FOR SET MESSAGE COMMANDS
011546                  N121$:    CLI       CLIALP,0,N123$
011552                  N123$:    CLI       <'/>,NOTNUF,N125$
011556                            CLI       CLISTR,SIZE,N122$,<'SIZE'>
011572                            CLI       CLIBR,0,N126$
011576                  N122$:    CLI       CLISTR,QCOPY,N30$,<'COPY'>
011612                            CLI       CLIBR,0,N126$

                        ;NUMER FOR SIZE OR COPY
011616                  N126$:    CLI       <'=>,0,N30$
011622                            CLI       CLIDEC,NUM,N30$
011626                            CLI       CLIBR,0,N121$

                        ;GET MAINTENANCE LOOP TYPE FOR RUN 'LOOP' QUIALIFIER
011632                  N140$:    CLI       <'=>,0,N30$


011636                  N141$:    CLI       CLISTR,TTLLOP,N142$,<'INTERNALTTL'>
011660                            CLI       CLIBR,0,N115$
011664                  N142$:    CLI       CLISTR,CBLLOP,N143$,<'CABLE'>
011700                            CLI       CLIBR,0,N115$
011704                  N143$:    CLI       CLISTR,LMDLOP,N144$,<'LOCALMODEM'>
011726                            CLI       CLIBR,0,N115$
011732                  N144$:    CLI       CLISTR,RMDLOP,N30$,<'REMOTEMODEM'>
011754                            CLI       CLIBR,0,N115$

                        ;GET LINE NUMBER FOR 'PASS' RUN QUALIFIER
011760                  N150$:    CLI       <'=>,0,N30$
011764                            CLI       CLIDEC,PASC,N30$
011770                            CLI       CLIBR,0,N115$

```

```
2363
2364
2365                                   ;END-OF-LINE
2366  011774                           N125$:  CLI     CLIEXI,0
2367
```

```


                              ;DEVICE DEPENDENT STORAGE LOCATIONS FOR
                              ;  CURRENT DEVICE PARAMTERS

 011776                       SEL0:
 011776  000000               BSEL0:  .WORD   0            ;ADDRESSES OF REGISTERS SEL0 THRU BSEL7
 012000  000000               BSEL1:  .WORD   0
 012002                       SEL2:
 012002  000000               BSEL2:  .WORD   0
 012004  000000               BSEL3:  .WORD   0
 012006                       SEL4:
 012006  000000               BSEL4:  .WORD   0
 012010  000000               BSEL5:  .WORD   0
 012012                       SEL6:
 012012  000000               BSEL6:  .WORD   0
 012014  000000               BSEL7:  .WORD   0


 012016  000000               INVEC:  .WORD   0            ;INPUT INTERRUPT VECTOR ADDRESS
 012020  000000               OUTVEC: .WORD   0            ;OUTPUT INTERRUPT VECTOR ADDRESS
 012022  000000               INTPRI: .WORD   0            ;INTERRUPT PRIORITY
 012024  000000               OPTYP:  .WORD   0            ;DEVICE OPTION TYPE(0=DMC,5=DMR-DMC MODE
                                                           ;7=DMR).





                              ;       ERRTBL
```

```
                                   .SBTTL  GLOBAL TEXT SECTION

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

                                   .SBTTL          DEVICE SUPPORTED
                                   ;
                                   ; NAMES OF DEVICES SUPPORTED BY PROGRAM
                                   ;
012026                                     DEVTYP  <DMR,DMC-11>
012026                                                                             L$DVTYP::
012026  046504  026122  046504                                                              .ASCIZ  /DMR,DMC-11/
012034  026503  030461     000
        012042                                                                              .EVEN



                                   .SBTTL          PROGRAM IDENTIFICATION
                                   ; TEST DESCRIPTION
                                   ;
012042                                     DESCRIPT        <CZCLKAO DMR, DMC-11 DATA COMM. LINK TEST>
012042                                                                             L$DESC::
012042  055103  046103  040513                                                              .ASCIZ  /CZCLKAO DMR, DM
012050  020060  046504  026122
012056  042040  041515  030455
012064  020061  040504  040524
012072  041440  046517  027115
012100  046040  047111  020113
012106  042524  052123     000
        012114                                                                              .EVEN
                                           .EVEN




```

```
                                                .SBTTL          GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO


        012114  041504  052114  000076  CLI$PM: .ASCIZ  /DCLT>/
        012122  047045  040445  044477  CLIERM: .ASCIZ  /%N%A?ILL CMD-BAD SYNTX?/
        012130  046114  041440  042115
        012136  041055  042101  051440
        012144  047131  054124  000077
        012152  047045  040445  044477  CLINUF: .ASCIZ  /%N%A?INCMPLTE CMD?/
        012160  041516  050115  052114
        012166  020105  046503  037504
        012174     000
        012175     045  022516  037501  CLINBG: .ASCIZ  /%N%A?NUM TOO BIG?/
        012202  052516  020115  047524
        012210  020117  044502  037507
        012216     000
        012217     045  022516  037501  CLIBRX: .ASCIZ  /%N%A?BAD RADIX?/
        012224  040502  020104  040522
        012232  044504  037530     000
        012237     045  022516  037501  CLIBDL: .ASCIZ  /%N%A?"LOOP" VALID ONLY IN ACTIVE?/
        012244  046042  047517  021120
        012252  053040  046101  042111
        012260  047440  046116  020131
        012266  047111  040440  052103
        012274  053111  037505     000
        012301     045  022516  037501  CLINPS: .ASCIZ  /%N%A?"ECHO" VALID ONLY IN PASSIVE?/
        012306  042442  044103  021117
        012314  053040  046101  042111
        012322  047440  046116  020131
        012330  047111  050040  051501
        012336  044523  042526  000077
        012344  047045  040445  044477  CLIBCR: .ASCIZ  /%N%A?ILL CHR- "A-Z,0-9,SP,TAB" ONLY?/
        012352  046114  041440  051110
        012360  020055  040442  055055
        012366  030054  034455  051454
        012374  026120  040524  021102
        012402  047440  046116  037531
        012410     000
        012411     045  022516  037501  CLISE0: .ASCIZ  /%N%A?"SIZE=0" NOT VALID?/
        012416  051442  055111  036505
        012424  021060  047040  052117
        012432  053040  046101  042111
        012440  000077
        012442  047045  040445  044124  HLP0:   .ASCIZ  /%N%ATHIS IS DCLT. TYPE "H" OR "?" FOR DETAILS/
        012450  051511  044440  020123
        012456  041504  052114  020056
        012464  054524  042520  021040
        012472  021110  047440  020122
        012500  037442  020042  047506
        012506  020122  042504  040524
        012514  046111  000123
        012520  047045  052045     000  HLPF:   .ASCIZ  /%N%T/
        012525     104  046103  020124  HLP1:   .ASCIZ  /DCLT CMDS:/
        012532  046503  051504  000072
        012540  041440  042514  051101  HLP2:   .ASCII  / CLEAR OR SHOW  EXPECTLIST OR TRANSMITLIST/<15><12>
        012546  047440  020122  044123
```

```
 2491  012554  053517  020040  054105
 2492  012562  042520  052103  044514
 2493  012570  052123  047440  020122
 2494  012576  051124  047101  046523
 2495  012604  052111  044514  052123
 2496  012612  005015
 2497  012614  050040  044522  052116                .ASCII  / PRINT/<15><12>
 2498  012622  005015
 2499  012624  042040  046525  020120                .ASCIZ  ? DUMP START-END/B?
 2500  012632  052123  051101  026524
 2501  012640  047105  027504  000102
 2502  012646  051440  052105  042440  HLP3:   .ASCIZ  ? SET EXPECTMSG OR TRANSMITMSG=TYPE/SIZE=N OR /COPY=N?
 2503  012654  050130  041505  046524
 2504  012662  043523  047440  020122
 2505  012670  051124  047101  046523
 2506  012676  052111  051515  036507
 2507  012704  054524  042520  051457
 2508  012712  055111  036505  020116
 2509  012720  051117  027440  047503
 2510  012726  054520  047075     000
 2511  012733     040  020040  054524  HLP4:   .ASCIZ  ?   TYPE=ONES,ZEROES,1ALT,0ALT,ITEP,CCITT,ALPHA?
 2512  012740  042520  047475  042516
 2513  012746  026123  042532  047522
 2514  012754  051505  030454  046101
 2515  012762  026124  040460  052114
 2516  012770  044454  042524  026120
 2517  012776  041503  052111  026124
 2518  013004  046101  044120  000101
 2519  013012  020040  020040  020040  HLP4A:  .ASCIZ  /        OR ''OPR SPCD=A-Z,SP,TAB,0-9 IN QUOTES''/
 2520  013020  047440  020122  047442
 2521  013026  051120  051440  041520
 2522  013034  036504  026501  026132
 2523  013042  050123  052054  041101
 2524  013050  030054  034455  044440
 2525  013056  020116  052521  052117
 2526  013064  051505  000042
 2527  013070  051040  047125  046440  HLP5:   .ASCIZ  ? RUN MODE=MTYP/LOOP=LTYP/CHECK,STATUS,ECHO,PASS=N?
 2528  013076  042117  036505  052115
 2529  013104  050131  046057  047517
 2530  013112  036520  052114  050131
 2531  013120  041457  042510  045503
 2532  013126  051454  040524  052524
 2533  013134  026123  041505  047510
 2534  013142  050054  051501  036523
 2535  013150  000116
 2536  013152  020040  046440  054524  HLP6:   .ASCII  /   MTYP=TRAN,REC,ACT,PAS,TAL,LIS,DOWN/<15><12>
 2537  013160  036520  051124  047101
 2538  013166  051054  041505  040454
 2539  013174  052103  050054  051501
 2540  013202  052054  046101  046054
 2541  013210  051511  042054  053517
 2542  013216  006516     012
 2543  013221     040  020040  052114                .ASCIZ  /   LTYP=INT,CAB,LOC,REM/
 2544  013226  050131  044475  052116
 2545  013234  041454  041101  046054
 2546  013242  041517  051054  046505
```

```
013250     000

013251     045  022516  046501  SHMSG:   .ASCIZ  ?%N%AMSG: TYPE=%T%A/SIZE=%D3?
013256  043523  020072  054524
013264  042520  022475  022524
013272  027501  044523  042532
013300  022475  031504     000
013305     132  051105  042517  SHTYP0: .ASCIZ  /ZEROES/
013312  000123
013314  047117  051505     000  SHTYP1: .ASCIZ  /ONES/
013321     061  046101  000124  SHTYP2: .ASCIZ  /1ALT/
013326  040460  052114     000  SHTYP3: .ASCIZ  /0ALT/
013333     103  044503  052124  SHTYP4: .ASCIZ  /CCITT/
013340     000
013341     111  042524  000120  SHTYP5: .ASCIZ  /ITEP/
013346  046101  044120  000101  SHTYP6: .ASCIZ  /ALPHA/
013354  050117  020122  050123  SHTYP7: .ASCIZ  /OPR SPEC/
013362  041505     000
013365     122  041505  044505  M00:    .ASCIZ  /RECEIVE/
013372  042526     000
013375     124  040522  051516  M01:    .ASCIZ  /TRANSMIT/
013402  044515  000124
013406  040520  051523  053111  M02:    .ASCIZ  /PASSIVE/
013414  000105
013416  041501  044524  042526  M03:    .ASCIZ  /ACTIVE/
013424     000
013425     104  053517  046116  M04:    .ASCIZ  /DOWNLINELOAD/
013432  047111  046105  040517
013440  000104
013442  040524  045514     000  M05:    .ASCIZ  /TALK/
013447     114  051511  042524  M06:    .ASCIZ  /LISTEN/
013454  000116
013456     000                  LP0:    .ASCIZ  //
013457     057  047514  050117  LP00:   .ASCIZ  ?/LOOP=?
013464  000075
013466  047111  042524  047122  LP1:    .ASCIZ  ?INTERNAL?
013474  046101     000
013477     103  041101  042514  LP2:    .ASCIZ  ?CABLE?
013504     000
013505     114  041517  046101  LP3:    .ASCIZ  ?LOCALMODEM?
013512  047515  042504  000115
013520  042522  047515  042524  LP4:    .ASCIZ  ?REMOTEMODEM?
013526  047515  042504  000115
013534  047516                  PNST:   .ASCII  /NO/
013536  052123  052101  051525  PST:    .ASCIZ  /STATUS/
013544     000
013545     116     117          PNCK:   .ASCII  /NO/
013547     103  042510  045503  PCK:    .ASCIZ  /CHECK/
013554     000
013555     116     117          PNEC:   .ASCII  /NO/
013557     105  044103  000117  PEC:    .ASCIZ  /ECHO/


013564  047045  040445  044514  LISP:   .ASCIZ  /%N%ALIS>/
013572  037123     000
013575     124  045514  000076  OPRMM:  .ASCIZ  /TLK>/
```

```
013602  044124  051511  040440  L5060:   .ASCIZ  /THIS A 50. OR 60. HZ. LSI-11:/
013610  032440  027060  047440
013616  020122  030066  020056
013624  055110  020056  051514
013632  026511  030461  000072
                                         .EVEN




                                ;
                                ;  FORMAT STATEMENTS USED IN PRINT CALLS
                                ;

013640  047045  040445  047504  DLLCM:   .ASCIZ  /%N%ADOWN LINE LOAD COMPLETED SUCCESSFULLY/
013646  047127  046040  047111
013654  020105  047514  042101
013662  041440  046517  046120
013670  052105  042105  051440
013676  041525  042503  051523
013704  052506  046114  000131
013712  047045  040445  040502  NOCLK:   .ASCIZ  /%N%ABAD CLOCK - PROGRAM WILL HANG ON "TIMEOUT"!!/
013720  020104  046103  041517
013726  020113  020055  051120
013734  043517  040522  020115
013742  044527  046114  044040
013750  047101  020107  047117
013756  021040  044524  042515
013764  052517  021124  020441
013772     000
013773     115  054101  020056  TABEX:   .ASCIZ  /MAX. CHAR. MSG COUNT EXCEEDED -/
014000  044103  051101  020056
014006  051515  020107  047503
014014  047125  020124  054105
014022  042503  042105  042105
014030  026440     000
014033     102  043125  042506  BUFEX:   .ASCIZ  /BUFFER FULL -/
014040  020122  052506  046114
014046  026440     000
014051     045  022516  022524  MSGTRN:  .ASCIZ  /%N%T%A MSG. NOT BUILT !!/
014056  020101  051515  027107
014064  047040  052117  041040
014072  044525  052114  020440
014100  000041
014102  047045  040445  044103  MSGTRU:  .ASCIZ  /%N%ACHAR. COUNT EXCEEDS BUFF LIMIT - MSG TRUNCATED/
014110  051101  020056  047503
014116  047125  020124  054105
014124  042503  042105  020123
014132  052502  043106  046040
014140  046511  052111  026440
014146  046440  043523  052040
014154  052522  041516  052101
014162  042105     000
014165     045  022516  032523  SHFO:    .ASCIZ  ?%N%S5%AMODE=%T%T%T%A/PASS=%Z5?
014172  040445  047515  042504
014200  022475  022524  022524
014206  022524  027501  040520
```

```
 2659  014214  051523  022475  032532
 2660  014222     000
 2661
 2662
 2663  014223     045  022516  032523  SHF1:    .ASCIZ   ?%N%S5%S5%S5%A/%T%A/%T%A/%T?
 2664  014230  051445  022465  032523
 2665  014236  040445  022457  022524
 2666  014244  027501  052045  040445
 2667  014252  022457  000124
 2668
 2669  014256  051445  022465  052101  EFM2:    .ASCIZ   /%S5%ATOTAL MISMATCHES IN MSG = %D5/
 2670  014264  052117  046101  046440
 2671  014272  051511  040515  041524
 2672  014300  042510  020123  047111
 2673  014306  046440  043523  036440
 2674  014314  022440  032504     000
 2675  014321     045  022516  031523  PCPM:    .ASCIZ   /%N%S3%ACALLED FROM PC=%O6/
 2676  014326  040445  040503  046114
 2677  014334  042105  043040  047522
 2678  014342  020115  041520  022475
 2679  014350  033117     000
 2680  014353     045  032523  040445  EFM11:   .ASCIZ   /%S5%ACOMPARE COUNT=%D5%S3%ARECEIVE COUNT=%D5/
 2681  014360  047503  050115  051101
 2682  014366  020105  047503  047125
 2683  014374  036524  042045  022465
 2684  014402  031523  040445  042522
 2685  014410  042503  053111  020105
 2686  014416  047503  047125  036524
 2687  014424  042045  000065
 2688
 2689
 2690                                  ;EVENT DESCRIPTION MESSAGES
 2691
 2692  014430  051124  047101  046523  EDTXQ:   .ASCIZ   /TRANSMIT MSG QUEUED/
 2693  014436  052111  046440  043523
 2694  014444  050440  042525  042525
 2695  014452  000104
 2696  014454  051124  047101  046523  EDTXC:   .ASCIZ   /TRANSMIT MSG COMPLETED/
 2697  014462  052111  046440  043523
 2698  014470  041440  046517  046120
 2699  014476  052105  042105     000
 2700  014503     122  041505  044505  EDRXQ:   .ASCIZ   /RECEIVE SPACE QUEUED/
 2701  014510  042526  051440  040520
 2702  014516  042503  050440  042525
 2703  014524  042525  000104
 2704  014530  042522  042503  053111  EDRXC:   .ASCIZ   /RECEIVE MSG COMPLETED/
 2705  014536  020105  051515  020107
 2706  014544  047503  050115  042514
 2707  014552  042524  000104
 2708  014556  042504  044526  042503  EDDER:   .ASCIZ   /DEVICE ERROR/
 2709  014564  042440  051122  051117
 2710  014572     000
 2711  014573     104  052101  020101  EDDCK:   .ASCIZ   /DATA COMPARISON STARTED/
 2712  014600  047503  050115  051101
 2713  014606  051511  047117  051440
 2714  014614  040524  052122  042105
```

```
014622     000
014623     104  053105  041511  EDDVI:  .ASCIZ  /DEVICE INIT AND SETUP/
014630  020105  047111  052111
014636  040440  042116  051440
014644  052105  050125     000
014651     104  052101  020101  EDDLE:  .ASCIZ  /DATA COMPARISON LENGTH ERROR/
014656  047503  050115  051101
014664  051511  047117  046040
014672  047105  052107  020110
014700  051105  047522  000122
014706  040504  040524  041440  EDDDE:  .ASCIZ  /DATA COMPARISON DATA ERROR/
014714  046517  040520  044522
014722  047523  020116  040504
014730  040524  042440  051122
014736  051117     000
014741     105  042116  047440  EDEOP:  .ASCIZ  /END OF PASS/
014746  020106  040520  051523
014754     000


                                ;*************************************************************
                                ;THESE TWO STORAGE AREAS MUST NOT BE SEPERATED !!!!

                                ;EVENT REPORTING MESSAGES
014755     040  041040  051501  BASM1A: .ASCIZ  /  BASE TABLE/
014762  020105  040524  046102
014770  000105

014772  051445  022463  031517  BASM3:  .ASCIZ  /%S3%03/
015000     000
015001     045  031523  047445  BASM2:  .ASCIZ  /%S3%06/
015006  000066
015010  047045  047445  000066  BASM1:  .ASCIZ  /%N%06/




015016  047045  040445  044124  NULEVT: .ASCIZ  /%N%ATHE EVENT LOG IS EMPTY/
015024  020105  053105  047105
015032  020124  047514  020107
015040  051511  042440  050115
015046  054524     000
015051     045  022516  037101  EVTF0:  .ASCIZ  /%N%A>>> EVENT LOG ENTRY <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<</
015056  037076  042440  042526
015064  052116  046040  043517
015072  042440  052116  054522
015100  036040  036074  036074
015106  036074  036074  036074
015114  036074  036074  036074
015122  036074  036074  036074
015130  036074  036074  036074
015136  036074  036074  036074
015144  036074     000
015147     045  022516  032504  EVTF1:  .ASCIZ  /%N%D5%A:%Z2%A:%Z2%S3%T/
015154  040445  022472  031132
015162  040445  022472  031132
015170  051445  022463  000124
```

```
 2771  015176  047045  051445  022463  EVTF2:  .ASCIZ  /%N%S3%AADDR OF MSG=%O6%S3%ABYTE COUNT=%D5/
 2772  015204  040501  042104  020122
 2773  015212  043117  046440  043523
 2774  015220  022475  033117  051445
 2775  015226  022463  041101  052131
 2776  015234  020105  047503  047125
 2777  015242  036524  042045  000065
 2778  015250  047045  051445  022463  EVTF3:  .ASCIZ  /%N%S3%T%N/
 2779  015256  022524  000116
 2780  015262  051445  022463  033117  EVTF3C: .ASCIZ  /%S3%O6%S3%O6/
 2781  015270  051445  022463  033117
 2782  015276     000
 2783  015277     045  031523  047445  EVTF3D: .ASCIZ  /%S3%O6%S3%O6%S3%T/
 2784  015304  022466  031523  047445
 2785  015312  022466  031523  052045
 2786  015320     000
 2787  015321     045  022516  031523  EVTF4:  .ASCIZ  /%N%S3%AADDR OF MSG=%O6%S3%ABYTE COUNT=%D5%S3%ANO. OF CMP ERRS=%D5/
 2788  015326  040445  042101  051104
 2789  015334  047440  020106  051515
 2790  015342  036507  047445  022466
 2791  015350  031523  040445  054502
 2792  015356  042524  041440  052517
 2793  015364  052116  022475  032504
 2794  015372  051445  022463  047101
 2795  015400  027117  047440  020106
 2796  015406  046503  020120  051105
 2797  015414  051522  022475  032504
 2798  015422     000
 2799  015423     045  022516  031523  EVTF4A: .ASCIZ  /%N%S3%AADDR OF MSG=%O6%S3%ARX BYTES=%D5%S3%ACOMPARE BYTES=%D5/
 2800  015430  040445  042101  051104
 2801  015436  047440  020106  051515
 2802  015444  036507  047445  022466
 2803  015452  031523  040445  054122
 2804  015460  041040  052131  051505
 2805  015466  022475  032504  051445
 2806  015474  022463  041501  046517
 2807  015502  040520  042522  041040
 2808  015510  052131  051505  022475
 2809  015516  032504     000
 2810  015521     045  022516  031523  EVTF4B: .ASCIZ  /%N%S3%APASS=%D5%S3%AERRORS=%D5%S3%ANOBUFFS=%D5/
 2811  015526  040445  040520  051523
 2812  015534  022475  032504  051445
 2813  015542  022463  042501  051122
 2814  015550  051117  036523  042045
 2815  015556  022465  031523  040445
 2816  015564  047516  052502  043106
 2817  015572  036523  042045  000065
 2818  015600  051445  022465  041101  EVTF5A: .ASCIZ  /%S5%ABYTE # IN MSG.=%D5%S3%AEXPTD=%O3%S3%ARECVD=%O3/
 2819  015606  052131  020105  020043
 2820  015614  047111  046440  043523
 2821  015622  036456  042045  022465
 2822  015630  031523  040445  054105
 2823  015636  052120  036504  047445
 2824  015644  022463  031523  040445
 2825  015652  042522  053103  036504
 2826  015660  047445  000063
```

```

015664  047045  051445  022471  EVMOCG: .ASCIZ  /%N%S9%ACHANGED TO:/
015672  041501  040510  043516
015700  042105  052040  035117
015706     000

                                ; ****************************************************************************
                                ;DO NOT SEPERATE THE NEXT LIST OF MESSAGES - MODEM SIGNAL HEADER AND REPORT

015707     045  022516  034123  EVMOHD: .ASCIZ  /%N%S8%AMODEM STATUS: CTS DSR DCD RTS RI  SQD TM/
015714  040445  047515  042504
015722  020115  052123  052101
015730  051525  020072  052103
015736  020123  051504  020122
015744  041504  020104  052122
015752  020123  044522  020040
015760  050523  020104  046524
015766     000
015767     045  022516  034523  EVMOST: .ASCII  /%N%S9%S9%S5%A/
015774  051445  022471  032523
016002  040445
016004     130     040     040  EVMCTS: .BYTE   'X,40,40,40
016007     040
016010     130     040     040  EVMDSR: .BYTE   'X,40,40,40
016013     040
016014     130     040     040  EVMDCD: .BYTE   'X,40,40,40
016017     040
016020     130     040     040  EVMRTS: .BYTE   'X,40,40,40
016023     040
016024     130     040     040  EVMRI:  .BYTE   'X,40,40,40
016027     040
016030     130     040     040  EVMSQD: .BYTE   'X,40,40,40
016033     040
016034     130     040     040  EVMTM:  .BYTE   'X,40,40,40
016037     040
016040     000                          .BYTE   0
        016042                          .EVEN


                                ;EXECUTION STATUS MESSAGES TO BE PRINTED TO KEEP OPERATOR AWAKE
016042  047045     000          CR:     .ASCIZ  /%N/            ;CR FOR LINES IN A ROW
016045     045  031523  040445  STXQ:   .ASCIZ  /%S3%ATXQ/      ;ABOUT TO TRANSMIT
016052  054124  000121
016056  051445  022463  052101  STXC:   .ASCIZ  /%S3%ATXC/      ;TX COMPLETED
016064  041530     000
016067     045  031523  040445  SRXQ:   .ASCIZ  /%S3%ARXQ/      ;ABOUT TO RECEIVE
016074  054122  000121
016100  051445  022463  042501  SDVE:   .ASCIZ  /%S3%AERR/      ;DEVICE ERROR
016106  051122     000
016111     045  031523  040445  SCM:    .ASCIZ  /%S3%ACMP/      ;ABOUT TO DO DATA CHECKING OF RECVD VS. EXPTD
016116  046503  000120
016122  051445  022463  044501  SDVI:   .ASCIZ  /%S3%AINI/      ;DEVICE ABOUT TO BE INITIALIZED
016130  044516     000
016133     045  031523  040445  SCML:   .ASCIZ  /%S3%ACML/      ;COMPARE LENGTH ERROR
016140  046503  000114
016144  051445  022463  041501  SCMD:   .ASCIZ  /%S3%ACMD/      ;COMPARE DATA ERROR
016152  042115     000
```

```
  2883  016155       045  031523  040445  SEOP:   .ASCIZ  /%S3%AEOP/      ;END  OF PASS
  2884  016162  047505  000120
  2885                                            .EVEN
  2886
```

```

                                        ;DEVICE ERROR MESSAGES

016166  044524  042515  047440  DVEM0:  .ASCII  /TIME OUT WAITING FOR RDI TO CLEAR/
016174  052125  053440  044501
016202  044524  043516  043040
016210  051117  051040  044504
016216  052040  020117  046103
016224  040505     122
016227     015  020012  020040          .ASCIZ  <15><12>/   SEL0     SEL2  /
016234  042523  030114  020040
016242  020040  051440  046105
016250  020062  000040
016254  044524  042515  047440  DVEM1:  .ASCII  /TIME OUT WAITING FOR RDI TO SET/
016262  052125  053440  044501
016270  044524  043516  043040
016276  051117  051040  044504
016304  052040  020117  042523
016312     124
016313     015  020012  020040          .ASCIZ  <15><12>/   SEL0     SEL2  /
016320  042523  030114  020040
016326  020040  051440  046105
016334  020062  000040
016340  044524  042515  047440  DVEM3:  .ASCII  /TIME OUT WAITING FOR RUN TO SET/
016346  052125  053440  044501
016354  044524  043516  043040
016362  051117  051040  047125
016370  052040  020117  042523
016376     124
016377     015  020012  020040          .ASCIZ  <15><12>/   SEL0     SEL2  /
016404  042523  030114  020040
016412  020040  051440  046105
016420  020062  000040
016424  044524  042515  047440  DVEM4:  .ASCII  /TIME OUT WAITING FOR OUTPUT INTERRUPT/
016432  052125  053440  044501
016440  044524  043516  043040
016446  051117  047440  052125
016454  052520  020124  047111
016462  042524  051122  050125
016470     124
016471     015  020012  020040          .ASCIZ  <15><12>/   SEL0     SEL2  /
016476  042523  030114  020040
016504  020040  051440  046105
016512  020062  000040
016516  047111  052520  020124  DVEM5:  .ASCII  /INPUT INTERRUPT WHEN EXPECTING OUTPUT/
016524  047111  042524  051122
016532  050125  020124  044127
016540  047105  042440  050130
016546  041505  044524  043516
016554  047440  052125  052520
016562     124
016563     015  020012  020040          .ASCIZ  <15><12>/   SEL0     SEL2  /
016570  042523  030114  020040
016576  020040  051440  046105
016604  020062  000040
016610  046111  042514  040507  DVEM6:  .ASCII  /ILLEGAL OUTPUT INTERRUPT/
```

```
016616  020114  052517  050124
016624  052125  044440  052116
016632  051105  052522  052120
016640  005015  020040  051440          .ASCIZ  <15><12>/   SEL2     SEL6  /
016646  046105  020062  020040
016654  020040  042523  033114
016662  020040     000
016665     103  047117  051124  DVEM7:  .ASCII  /CONTROL OUT INSTEAD OF BA-CC OUT/
016672  046117  047440  052125
016700  044440  051516  042524
016706  042101  047440  020106
016714  040502  041455  020103
016722  052517     124
016725     015  020012  020040          .ASCIZ  <15><12>/   SEL2     SEL6  /
016732  042523  031114  020040
016740  020040  051440  046105
016746  020066  000040

016752  054124  041040  043125  DVEM8:  .ASCII  /TX BUFF COMPLETED AND SHOULD BE RX/
016760  020106  047503  050115
016766  042514  042524  020104
016774  047101  020104  044123
017002  052517  042114  041040
017010  020105  054122
017014  005015  020040  051440          .ASCIZ  <15><12>/   SEL4     SEL6  /
017022  046105  020064  020040
017030  020040  042523  033114
017036  020040     000
017041     122  020130  052502  DVEM9:  .ASCII  /RX BUFF COMPLETED AND SHOULD BE TX/
017046  043106  041440  046517
017054  046120  052105  042105
017062  040440  042116  051440
017070  047510  046125  020104
017076  042502  052040     130
017103     015  020012  020040          .ASCIZ  <15><12>/   SEL4     SEL6  /
017110  042523  032114  020040
017116  020040  051440  046105
017124  020066  000040
017130  042040  053517  020116  DLLAB:  .ASCII  / DOWN LINE LOAD ABORTED/
017136  044514  042516  046040
017144  040517  020104  041101
017152  051117  042524     104
017157     015  020012  020040          .ASCIZ  <15><12>/   RXBUF    TXBUF /
017164  054122  052502  020106
017172  020040  052040  041130
017200  043125  000040

017204  051120  041517  042105  PROEM:  .ASCIZ  /PROCEDURE ERROR/
017212  051125  020105  051105
017220  047522  000122
017224  047516  020116  054105  NXMM:   .ASCIZ  /NON EXIST MEM/
017232  051511  020124  042515
017240  000115
017242  042104  046503  020120  DDCSRM: .ASCIZ  /DDCMP START REC/
017250  052123  051101  020124
017256  042522  000103
```

```
2999 017262 044504 041523 047117  DISCOM: .ASCIZ  /DISCONNECT/
3000 017270 042516 052103    000
3001 017275    114 051517 020124  LOSDAM: .ASCIZ  /LOST DATA/
3002 017302 040504 040524    000
3003 017307    104 041504 050115  DDCMRM: .ASCIZ  /DDCMP MAINT REC/
3004 017314 046440 044501 052116
3005 017322 051040 041505    000
3006 017327    124 046511 020105  TIMOM:  .ASCIZ  /TIME OUT/
3007 017334 052517 000124
3008 017340 040504 040524 041440  DATCKM: .ASCIZ  /DATA CHECK/
3009 017346 042510 045503    000
3010
3011        017354                        .EVEN
3012
```

```

                                        ;THIS SECTION IS USED BY A M9301-YJ BOOT ROM FOR DOING DOWN-LINE-LOAD ?????
                                        ;MUST BE IN THE AREA OF '017370 + 256. BYTES'' + A FEW

         017370                                 .=17370
 017370  000400                         BASE:   .BLKB   256.            ;BASE TABLE ADDRESS

         020000                                 .=20000
 3021  020000  052522  020116  042523  RUNSBM: .ASCIZ  /RUN SET ILLEAGLLY/
 3022  020006  020124  046111  042514
 3023  020014  043501  046114  000131
 3024  020022  054122  044440  046104  RXIDM:  .ASCIZ  /RX IDLE/
 3025  020030  000105
 3026  020032  042103  043440  044514  CDGLM:  .ASCIZ  /CD GLITCHED/
 3027  020040  041524  042510  000104
 3028  020046  052103  020123  040506  CTSFM:  .ASCIZ  /CTS FALILED/
 3029  020054  044514  042514  000104
 3030  020062  054124  047040  052117  TXNC:   .ASCIZ  /TX NOT COMPLETE/
 3031  020070  041440  046517  046120
 3032  020076  052105  000105
 3033  020102  054122  047040  052117  RXNC:   .ASCIZ  /RX NOT COMPLETE/
 3034  020110  041440  046517  046120
 3035  020116  052105  000105
 3036  020122  042523  020103  042522  RXM1:   .ASCIZ  /SEC REQ ERR WORD 1/
 3037  020130  020121  051105  020122
 3038  020136  047527  042122  030440
 3039  020144     000
 3040  020145     123  041505  051040  RXM2:   .ASCIZ  /SEC REQ ERR WORD 2/
 3041  020152  050505  042440  051122
 3042  020160  053440  051117  020104
 3043  020166  000062
 3044                                          .EVEN
 3045
 3046
 3047
 3048
 3049
 3050
 3051
```

```
                                        .SBTTL  GLOBAL ERROR REPORT SECTION

                                        ;++
                                        ; THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
                                        ; USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION.  PRINTB
                                        ; (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
                                        ;--






020170                                          BGNMSG  ERR1
020170                                                                                  ERR1::
020170                                          PRINTB  #EVTF5A,OFSET,<B,GOOD>,<B,BAD>  ;INDIVIDUAL DATA COMPARE ERROR
020170  005046                                                                                  CLR     -(SP)
020172  153716  007211                                                                          BISB    BAD,(SP)
020176  005046                                                                                  CLR     -(SP)
020200  153716  007210                                                                          BISB    GOOD,(SP)
020204  013746  007170                                                                          MOV     OFSET,-(SP)
020210  012746  015600                                                                          MOV     #EVTF5A,-(SP)
020214  012746  000004                                                                          MOV     #4,-(SP)
020220  010600                                                                                  MOV     SP,R0
020222  104414                                                                                  TRAP    C$PNTB
020224  062706  000012                                                                          ADD     #12,SP
020230                                          ENDMSG
020230                                                                                  L10001:
020230  104423                                                                                  TRAP    C$MSG

020232                                          BGNMSG  ERR2
020232                                                                                  ERR2::
020232                                          PRINTB  #EFM2,TEMP4                     ;TOTAL DATA COMPARE FAILS ERROR
020232  013746  007202                                                                          MOV     TEMP4,-(SP)
020236  012746  014256                                                                          MOV     #EFM2,-(SP)
020242  012746  000002                                                                          MOV     #2,-(SP)
020246  010600                                                                                  MOV     SP,R0
020250  104414                                                                                  TRAP    C$PNTB
020252  062706  000006                                                                          ADD     #6,SP
020256                                          ENDMSG
020256                                                                                  L10002:
020256  104423                                                                                  TRAP    C$MSG

020260                                          BGNMSG  ERR10
020260                                                                                  ERR10::
020260                                          PRINTB  #EFM11,R4,TEMP3
020260  013746  007200                                                                          MOV     TEMP3,-(SP)
020264  010446                                                                                  MOV     R4,-(SP)
020266  012746  014353                                                                          MOV     #EFM11,-(SP)
020272  012746  000003                                                                          MOV     #3,-(SP)
020276  010600                                                                                  MOV     SP,R0
020300  104414                                                                                  TRAP    C$PNTB
020302  062706  000010                                                                          ADD     #10,SP
020306                                          ENDMSG
020306                                                                                  L10003:
020306  104423                                                                                  TRAP    C$MSG

```

```
020310                          BGNMSG  ERR8
020310                                                          ERR8::
020310                          PRINTB  #EVTF3D,TEMP3,TEMP4,CONOTM
020310  013746  007204                                                  MOV     CONOTM,-(SP)
020314  013746  007202                                                  MOV     TEMP4,-(SP)
020320  013746  007200                                                  MOV     TEMP3,-(SP)
020324  012746  015277                                                  MOV     #EVTF3D,-(SP)
020330  012746  000004                                                  MOV     #4,-(SP)
020334  010600                                                          MOV     SP,R0
020336  104414                                                          TRAP    C$PNTB
020340  062706  000012                                                  ADD     #12,SP
020344                          PRINTB  #PCPM,PCADD
020344  013746  007214                                                  MOV     PCADD,-(SP)
020350  012746  014321                                                  MOV     #PCPM,-(SP)
020354  012746  000002                                                  MOV     #2,-(SP)
020360  010600                                                          MOV     SP,R0
020362  104414                                                          TRAP    C$PNTB
020364  062706  000006                                                  ADD     #6,SP
020370                          ENDMSG
020370                                                          L10004:
020370  104423                                                          TRAP    C$MSG

020372                          BGNMSG  ERR9
020372                                                          ERR9::
020372                          PRINTB  #EVTF3C,TEMP3,TEMP4
020372  013746  007202                                                  MOV     TEMP4,-(SP)
020376  013746  007200                                                  MOV     TEMP3,-(SP)
020402  012746  015262                                                  MOV     #EVTF3C,-(SP)
020406  012746  000003                                                  MOV     #3,-(SP)
020412  010600                                                          MOV     SP,R0
020414  104414                                                          TRAP    C$PNTB
020416  062706  000010                                                  ADD     #10,SP
020422                          PRINTB  #PCPM,PCADD
020422  013746  007214                                                  MOV     PCADD,-(SP)
020426  012746  014321                                                  MOV     #PCPM,-(SP)
020432  012746  000002                                                  MOV     #2,-(SP)
020436  010600                                                          MOV     SP,R0
020440  104414                                                          TRAP    C$PNTB
020442  062706  000006                                                  ADD     #6,SP
020446                          ENDMSG
020446                                                          L10005:
020446  104423                                                          TRAP    C$MSG

020450                          BGNMSG  ERR13
020450                                                          ERR13::
020450                          PRINTB  #EVTF3C,TEMP3,TEMP4
020450  013746  007202                                                  MOV     TEMP4,-(SP)
020454  013746  007200                                                  MOV     TEMP3,-(SP)
020460  012746  015262                                                  MOV     #EVTF3C,-(SP)
020464  012746  000003                                                  MOV     #3,-(SP)
020470  010600                                                          MOV     SP,R0
020472  104414                                                          TRAP    C$PNTB
020474  062706  000010                                                  ADD     #10,SP
020500                          ENDMSG
020500                                                          L10006:
020500  104423                                                          TRAP    C$MSG
```

```

020502                          BGNMSG  ERR14
020502                                                                      ERR14::
020502                          PRINTB  #EVTF3D,TEMP3,TEMP4,CONOTM
020502  013746  007204                                                               MOV     CONOTM,-(SP)
020506  013746  007202                                                               MOV     TEMP4,-(SP)
020512  013746  007200                                                               MOV     TEMP3,-(SP)
020516  012746  015277                                                               MOV     #EVTF3D,-(SP)
020522  012746  000004                                                               MOV     #4,-(SP)
020526  010600                                                                       MOV     SP,R0
020530  104414                                                                       TRAP    C$PNTB
020532  062706  000012                                                               ADD     #12,SP
020536                          ENDMSG
020536                                                                      L10007:
020536  104423                                                                       TRAP    C$MSG

020540                          EXIT    MSG
020540  000167                                                                       .WORD   J$JMP
020542  177772                                                                       .WORD   L10007-2-.


```

```
                                .SBTTL  GLOBAL SUBROUTINES SECTION

                                ;++
                                ; THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
                                ; THAT ARE USED IN MORE THAN ONE TEST.
                                ;--


                                .SBTTL          CLOCK SETUP SUBROUTINE

                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;       THIS SUBROUTINE SETS UP THE CLOCK INFORMATION TABLE FOLLOWING A ''CLOCK''
                                ;       CALL EXECUTED IN THE INITIALIZATION CODE. BUT SINCE THE ''CLOCK'' CALL
                                ;       SAYS NOTHING ABOUT AN LSI-11'S CLOCK, THIS ROUTINE IS ONLY USED IF A
                                ;       LINE OR P-CLOCK IS FOUND.
                                ;
                                ; INPUTS:
                                ;       R1= POINTS TO SUPERVISOR SPACE WHERE CLOCK INFO WAS RETURNED
                                ;       R2= POINTS TO ''CLK'' TABLE WHERE CLOCK INFO WILL BE KEPT
                                ;
                                ; IMPLICIT INPUTS:
                                ;       THE SUPERVISOR  SPACE WHERE CLOCK INFO WAS RETURNED BY THE ''CLOCK'' CALL
                                ;
                                ; OUTPUTS:
                                ;       ''CLKCSR'' GETS LOADED WITH THE CLOCK'S CSR ADDRESS
                                ;       ''CLKBR'' GETS LOADED WITH THE CLOCK'S INTERRUPT LEVEL
                                ;       ''CLKVEC'' GETS LOADED WITH THE CLOCK'S INTERRUPT VECTOR
                                ;       ''CLKHZ'' GETS LOADED WITH THE LINE FREQ. (HERTZ RATE) WHICH DETERMINES
                                ;               THE NUMBER OF TICKS IN A SECOND
                                ;
                                ; CALLING SEQUENCE:
                                ;       JSR     PC,CLKSET                       ;CALL CLOCK SETUP WITH R1 & R2 SETUP
                                ;--

 020544                         CLKSET:
 020544  012122                         MOV     (R1)+,(R2)+                     ;LOAD CLOCK'S CSR ADDR. INTO ''CLKCSR''
 020546  012112                         MOV     (R1)+,(R2)                      ;LOAD CLOCK'S INT. LEVEL INTO ''CLKBR''
 020550  006312                         ASL     (R2)                            ;ADJUST THE INT. LEVEL FOR LOADING INTO
 020552  006312                         ASL     (R2)                            ;   THE PSW WITH A ''SETVEC'' CALL
 020554  006312                         ASL     (R2)
 020556  006312                         ASL     (R2)
 020560  006322                         ASL     (R2)+
 020562  012122                         MOV     (R1)+,(R2)+                     ;LOAD CLOCK'S INT. VECTOR INTO ''CLKVEC''
 020564  012122                         MOV     (R1)+,(R2)+                     ;LOAD CLOCK'S HERTZ RATE INTO ''CLKHZ''
 020566  000207                         RTS     PC
```

```
                                        .SBTTL          CLOCK INTERRUPT SERVICE ROUTINE
                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       THIS IS THE CLOCK INTERRUPT SERVICE ROUTINE WHICH TAKES CARE OF
                                        ;       KEEPING THE 'TIME-SINCE-START'' AND COUNTING DOWN ANY OF THE
                                        ;       'EVENT'' TIMERS. THE TIMERS ARE USED TO TIME COMPLETION OF  DEVICE
                                        ;       REQUESTS. THE 'TIME-SINCE-START'' IS USED TO BE LOGGED WITH EACH ENTRY
                                        ;       INTO THE EVENT LOG.
                                        ;
                                        ; IMPLICIT INPUTS:
                                        ;       TIMTCK: THE CURRENT NO. OF TICKS LEFT TO BE COUNTED UNTIL A SECOND
                                        ;               HAS BEEN COUNTED OFF
                                        ;       CLKHZ:  THE NO. OF TICKS IN A SECOND, DETERMINED BY THE SYS. LINE FREQ.
                                        ;       TIMMIN & TIMSEC: CURRENT VALUE OF 'TIME-SINCE-START''
                                        ;                                IN MINUTES & SECONDS
                                        ;       TIMER 1,2, & S: CURRENT VALUES OF THE 'EVENT TIMERS''
                                        ;
                                        ; IMPLICIT OUTPUTS:
                                        ;       NEW VALUE OF EVENT TIMER '1'' DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
                                        ;       NEW VALUE OF EVENT TIMER '2'' DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
                                        ;       NEW VALUE OF EVENT TIMER 'S'' DECREMENTED BY 1 SECOND IF IT WAS NON-ZERO
                                        ;
                                        ; FUNCTIONAL SIDE EFFECTS:
                                        ;       THE CLOCK IS DISABLED UPON ENTRY AND REENABLED WHEN LEAVING
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;       THIS ROUTINE IS CALLED WHEN THE CLOCK INTERRUPTS THRU ''CLKVEC''.
                                        ;       THE ADDRESS OF THIS ROUTINE WAS LOADED INTO THE CLOCK'S INTERRUPT
                                        ;       VECTOR WITH A SUPERVISOR ''SETVEC'' CALL.
                                        ;--

 020570                                         BGNSRV  CLKINT
 020570                                                                         CLKINT::

 020570  005077  166456                         CLR     @CLKCSR         ;DISABLE THE CLOCK FORM INTERRUPTING
 020574  005337  007270                         DEC     TIMTCK          ;DECREMENT THE # OF TICKS/SEC.
 020600  001015                                 BNE     1$              ;GO CHECK TIMERS (1&2-TICKS, 3-SECONDS)
 020602  013737  007260  007270                 MOV     CLKHZ,TIMTCK    ;RESET THE # OF TICKS/SEC.
 020610  005237  007266                         INC     TIMSEC          ;INC # OF SECS-SINCE-START
 020614  022737  000074  007266                 CMP     #60.,TIMSEC     ;SEE IF WE'VE COUNTED 60 SECS. YET
 020622  001004                                 BNE     1$              ;IF NOT, GO CHECK TIMERS
 020624  005237  007264                         INC     TIMMIN          ; ELSE INC MINUTES-SINCE-START
 020630  005037  007266                         CLR     TIMSEC          ; AND RESTART SECOND COUNTER

 020634  005737  007272                 1$:     TST     TIMER1          ;SEE IF TIMER #1, TIMING ANYTHING
 020640  001402                                 BEQ     2$              ; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
 020642  005337  007272                         DEC     TIMER1          ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
 020646  005737  007274                 2$:     TST     TIMER2          ;SEE IF TIMER #2, TIMING ANYTHING
 020652  001402                                 BEQ     3$              ; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
 020654  005337  007274                         DEC     TIMER2          ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
 020660  005737  007276                 3$:     TST     TIMERS          ;SEE IF TIMER #3, TIMING ANYTHING
 020664  001406                                 BEQ     4$              ; IF=0, NOTHING BEING TIMED, LEAVE
 020666  023737  007260  007270                 CMP     CLKHZ,TIMTCK    ;SEE IF A SECOND HAS BEEN COUNTED OFF
 020674  001002                                 BNE     4$              ; BR IF NO
 020676  005337  007276                         DEC     TIMERS          ; ELSE DECREMENT THE TIMER VALUE (BY 1 SEC.)
```

```
020702  013777  007262  166342  4$:     MOV     CLKEN,@CLKCSR   ;REENABLE THE CLOCK TO INTERRUPT
020710                                  ENDSRV
020710                                                                                  L10010:
020710  000002                                                                                  RTI
```

```
                                        .SBTTL          EVENT LOG SUBROUTINES

                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       THIS SUBROUTINE HAS A DIFFERENT ENTRY POINT
                                        ;       FOR EACH EVENT TO BE LOGGED AND ALWAYS PRINTS
                                        ;       THE SHORT 'OPERATOR AWAKE'' MESSAGE TO CONSOLE THEN LOGS THE
                                        ;       EVENT TYPE, TIME, AND THE OTHER 3 WORDS OF INFO PASSED TO THE
                                        ;       SUBROUTINE AT CALLING TIME
                                        ;
                                        ; INPUTS:
                                        ;       TIMMIN & TIMSEC:        CURRENT VALUE OF 'TIME-SINCE-START''
                                        ;       TEMP2:  WORD #1 OF EVENT LOG INFORMATION (FOR MOST EVENT TYPES)
                                        ;       TEMP3:  WORD #2 OF EVENT LOG INFORMATION
                                        ;       TEMP4:  WORD #3 OF EVENT LOG INFORMATION
                                        ;       MODS:   CURRENT VALUE OF THE MODEM SIGNALS AVAILABLE FROM THE DEVICE
                                        ;
                                        ; OUTPUTS:
                                        ;       'OPERATOR AWAKE'' MESSAGE SENT TO THE CONSOLE
                                        ;       NEW EVENT LOGGED IN 'EVTLOG'' (EVENT LOG)
                                        ;       UPDATED 'EVTPTR'' (EVENT LOG ENTRY POINTER)
                                        ;
                                        ; SUBORDINATE ROUTINES USED:
                                        ;       'DVMODS'' THE DEVICE SUBROUTINE THAT RETURNS MODEM STATUS IN 'MODS''
                                        ;               (FOR SOME EVENT TYPES)
                                        ;
                                        ; FUNCTIONAL SIDE EFFECTS:
                                        ;       TEMP:   USED TO STORE ADDRESS OF 'OPERATOR AWAKE'' MESSAGE
                                        ;       TEMP1:  USED TO SETUP THE VALUE OF THE 'EVENT TYPE'' BYTE FOR LOGGING
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;       JSR     PC,LOGTXQ       ;CALL THE LOG EVENT SUBROUTINE WITH TEMP,TEMP1,
                                        ;        ''      ''  ''     ; TEMP2, TEMP3, AND TEMP4 SETUP
                                        ;        ''      ''  ''
                                        ;       JSR     PC,LOGCMP
                                        ;--

020712                                  LOGTXQ:
020712  012737  016045  007174                  MOV     #STXQ,TEMP1     ;SET UP MSG. TO PRINT
020720  012737  000000  007172                  MOV     #TXQ,TEMP       ;SET UP EVENT TYPE
020726  000510                                  BR      LOGS1           ;GO LOG EVENT AND TIME

020730                                  LOGTXC:
020730  012737  016056  007174                  MOV     #STXC,TEMP1     ;SET UP MSG. TO PRINT
020736  012737  000002  007172                  MOV     #TXC,TEMP       ;SET UP EVENT TYPE
020744  000501                                  BR      LOGS1           ;GO LOG EVENT AND TIME

020746                                  LOGRXQ:
020746  012737  016067  007174                  MOV     #SRXQ,TEMP1     ;SET UP MSG. TO PRINT
020754  012737  000004  007172                  MOV     #RXQ,TEMP       ;SET UP EVENT TYPE
020762  000472                                  BR      LOGS1           ;GO LOG EVENT AND TIME

020764                                  LOGRXC:
020764  012737  000006  007172                  MOV     #RXC,TEMP       ;SET UP EVENT TYPE
020772  000466                                  BR      LOGS1           ;GO LOG EVENT AND TIME
020774                                  LGDVE:
```

```
020774  012737  016100  007174          MOV     #SDVE,TEMP1     ;SET UP MSG. TO PRINT
021002  012737  000010  007172          MOV     #DER,TEMP       ;SET UP EVENT TYPE
021010  000474                          BR      LOGS3           ;GO LOG EVENT AND TIME

021012                          LOGDVI:
021012  012737  016122  007174          MOV     #SDVI,TEMP1     ;SET UP MSG. TO PRINT
021020  012737  000012  007172          MOV     #DVI,TEMP       ;SET UP EVENT TYPE
021026  113737  007220  007176          MOVB    MODTYP,TEMP2
021034  113737  007222  007177          MOVB    MLTYP,TEMP2+1
021042  013737  007230  007200          MOV     RPASS,TEMP3
021050  013737  007226  007202          MOV     PARAM,TEMP4     ;SET UP EVNT ENTRIES
021056  000451                          BR      LOGS3           ;GO LOG EVENT AND TIME

021060                          LOGCMP:
021060  012737  016111  007174          MOV     #SCM,TEMP1      ;SET UP MSG. TO PRINT
021066  012737  000014  007172          MOV     #DCK,TEMP       ;SET UP EVENT TYPE
021074  000442                          BR      LOGS3
021076                          LOGCML:
021076  012737  016133  007174          MOV     #SCML,TEMP1
021104  012737  000020  007172          MOV     #DLE,TEMP       ;SET UP MSG. AND TYPE
021112  000433                          BR      LOGS3           ;GO LOG EVENT AND TIME
021114                          LOGCMD:
021114  012737  016144  007174          MOV     #SCMD,TEMP1
021122  012737  000022  007172          MOV     #DDE,TEMP
021130  000424                          BR      LOGS3           ;GO LOG MSG TYPE AND TIME
021132                          LOGEOP:
021132  012737  016155  007174          MOV     #SEOP,TEMP1
021140  012737  000024  007172          MOV     #EOP,TEMP
021146  000415                          BR      LOGS3           ;GO LOG MSG TYPE AND TIME

021150  013746  007144          LOGS1:  MOV     ERRCNT,-(SP)    ;SAVE CURRENT ERROR COUNT
021154  004737  034064                  JSR     PC,DVMODS       ; GO GET MODEM STATUS
021160  012604                          MOV     (SP)+,R4        ;GET SAVED ERRCNT VALUE
021162  020437  007144                  CMP     R4,ERRCNT       ;WHERE ANY ERRORS FOUND
021166  001402                          BEQ     1$              ; BR IF NONE
021170  000137  021404                  JMP     LOGEX           ; ELSE, LEAVE WITHOUT LOGGING ANYTHING
                                                                ;    BUT THE DEVICE ERROR FROM 'DVMODS'
021174  013737  010206  007202  1$:     MOV     MODS,TEMP4      ;AND PUT IT IN TEMP4

021202                          LOGS3:
021202  022737  000006  007172          CMP     #RXC,TEMP
021210  001434                          BEQ     LOGS5           ;IF RXC DONT PRINT
021212  032737  000001  007226          BIT     #STATB,PARAM
021220  001430                          BEQ     LOGS5           ;IF NO STATUS SELECTED
                                                                ;GO TO 5

021222  022737  000010  007136          CMP     #10,LNCNT       ;HAVE WE DONE 10?
021230  001012                          BNE     LOGS4           ;IF NOT GO TO 4
021232  005037  007136                  CLR     LNCNT           ;ESLE CLEAR IT

021236                                  PRINTF  #CR             ;ELSE PRINT CR
021236  012746  016042                                                                  MOV     #CR,-(SP)
021242  012746  000001                                                                  MOV     #1,-(SP)
021246  010600                                                                          MOV     SP,R0
021250  104417                                                                          TRAP    C$PNTF
021252  062706  000004                                                                  ADD     #4,SP
```

```
021256                          LOGS4:
021256  005237  007136                  INC     LNCNT           ;INC COUNTER OF # OF AWAKE MSGS
021262                                  PRINTF  TEMP1           ;PRINT OPERATOR AWAKE MSG.
021262  013746  007174                                                                          MOV     TEMP1,-(SP)
021266  012746  000001                                                                          MOV     #1,-(SP)
021272  010600                                                                                  MOV     SP,R0
021274  104417                                                                                  TRAP    C$PNTF
021276  062706  000004                                                                          ADD     #4,SP
021302  010346                  LOGS5:  MOV     R3,-(SP)        ;SAVE R3 ON THE STACK
021304  013703  007300                  MOV     EVTPTR,R3
021310  113723  007172                  MOVB    TEMP,(R3)+      ;LOG EVENT
021314  013737  007260  007172          MOV     CLKHZ,TEMP
021322  163737  007270  007172          SUB     TIMTCK,TEMP
021330  113723  007172                  MOVB    TEMP,(R3)+      ;LOG TIME SINCE START
021334  113723  007266                  MOVB    TIMSEC,(R3)+
021340  113723  007264                  MOVB    TIMMIN,(R3)+    ;TICKS,SECS AND MINS.
021344  013723  007176                  MOV     TEMP2,(R3)+     ;LOG EVNT ENTRY 3
021350  013723  007200                  MOV     TEMP3,(R3)+     ;LOG EVNT ENTRY 4
021354  013723  007202                  MOV     TEMP4,(R3)+     ;LOG EVNT ENTRY 5
021360  020327  010204                  CMP     R3,#EVTEND
021364  103404                          BLO     LOGS2           ;IF EVENT LOG FULL GO
                                                                ;CONTINUE;ELSE GO TO 2
021366  012713  177777                  MOV     #-1,(R3)        ;LOG A TABLE END
021372  012703  007302                  MOV     #EVTLOG,R3      ;PUT R3 TO START OF TABLE
021376  010337  007300          LOGS2:  MOV     R3,EVTPTR       ;RESTORE POINTER
021402  012603                          MOV     (SP)+,R3        ;RESTORE R3
021404  000207                  LOGEX:  RTS     PC
```

```
                                         .SBTTL          DUMP EVENT LOG AND BASE TABLE


021406  010246                           REPORT: MOV     R2,-(SP)              ;SAVE R2,R3,R4 ON THE STACK
021410  010346                                   MOV     R3,-(SP)
021412  010446                                   MOV     R4,-(SP)
021414  005037  007172                           CLR     TEMP
021420                                           GMANIL  #BASM1A,TEMP,1,YES
021420  104443                                                                                       TRAP    C$GMAN
021422  000404                                                                                       BR      10000$
021424  007172                                                                                       .WORD   TEMP
021426  000130                                                                                       .WORD   T$CODE
021430  014755                                                                                       .WORD   #BASM1A
021432  000001                                                                                       .WORD   1
021434                                                                                       10000$:
021434  005737  007172                           TST     TEMP
021440  001413                                   BEQ     BASN1                 ;IF NO BASE PRINT GO TO 1

021442  012737  017370  007146                   MOV     #BASE,STADD
021450  012737  017767  007150                   MOV     #BASE+255.,ENADD
021456  012737  000001  007152                   MOV     #1,BYTBIT
021464  004737  022442                           JSR     PC,DUMPSR             ;GO DUMP BASE TABLE
021470  013702  007300                   BASN1:  MOV     EVTPTR,R2             ;MAKE R2 A POINTER TO EVENT TABLE
021474  023727  007302  177777                   CMP     EVTLOG,#-1            ;SEE IF EVENT TABLE IS EMPTY
021502  001034                                   BNE     RPT0                  ;BR IF NO
021504                                           PRINTS  #NULEVT               ;IF EMPTY TELL OPRERATOR.
021504  012746  015016                                                                               MOV     #NULEVT,-(SP)
021510  012746  000001                                                                               MOV     #1,-(SP)
021514  010600                                                                                       MOV     SP,R0
021516  104416                                                                                       TRAP    C$PNTS
021520  062706  000004                                                                               ADD     #4,SP
021524  000137  022320                           JMP     ENDEVT                ;AND END

021530  162702  000012                   RPT:    SUB     #12,R2                ;NOW POINT BACK TO TOP OF ENTRY U
                                                                               ;JUST PRINTED

021534  020227  007302                           CMP     R2,#EVTLOG            ;POINTING TO TOP OF EVNT LOG QUEUE?
021540  001010                                   BNE     RPT1                  ; BR IF NO
021542  012702  010204                           MOV     #EVTEND,R2            ;SET R2 TO POINT TO BOTTOM OF LOG
021546  026227  177776  177777                   CMP     -2(R2),#-1
021554  001007                                   BNE     RPT0                  ;IF END OF LOG IS NOT EMPTY
021556  000137  022320                           JMP     ENDEVT                ;CONTINUE...ELSE EXIT

021562  020237  007300                   RPT1:   CMP     R2,EVTPTR             ;ARE WE BACK TO POINTER?
021566  001002                                   BNE     RPT0                  ;IF NOT CONTINUE
021570  000137  022320                           JMP     ENDEVT                ;IF SO EXIT....

021574  162702  000012                   RPT0:   SUB     #12,R2                ;POINT R2 TO START OF ENTRY
021600                                   RPTAA:  PRINTS  #EVTF0                ;PRINT EVENT ENTRY HEADER
021600  012746  015051                                                                               MOV     #EVTF0,-(SP)
021604  012746  000001                                                                               MOV     #1,-(SP)
021610. 010600                                                                                       MOV     SP,R0
021612  104416                                                                                       TRAP    C$PNTS
021614  062706  000004                                                                               ADD     #4,SP
021620  112203                                   MOVB    (R2)+,R3              ;PUT EVENT TYPE INTO R3
021622  112237  010276                           MOVB    (R2)+,EVTTCK
```

```
021626  112237  010272            MOVB    (R2)+,EVTSEC      ;PUT EVENT TIME (TICKS,SECS,MINS IN TEMP LOC.S)
021632  112237  010274            MOVB    (R2)+,EVTMIN
021636                            PRINTS  #EVTF1,EVTMIN,EVTSEC,EVTTCK,EVTLST(R3)  ;PRINT EVENT TIME AND DESCRIPT.
021636  016346  010244                                                                  MOV     EVTLST(R3),-(SP)
021642  013746  010276                                                                  MOV     EVTTCK,-(SP)
021646  013746  010272                                                                  MOV     EVTSEC,-(SP)
021652  013746  010274                                                                  MOV     EVTMIN,-(SP)
021656  012746  015147                                                                  MOV     #EVTF1,-(SP)
021662  012746  000005                                                                  MOV     #5,-(SP)
021666  010600                                                                          MOV     SP,R0
021670  104416                                                                          TRAP    C$PNTS
021672  062706  000014                                                                  ADD     #14,SP
021676  000173  010306            JMP     @RPTDSP(R3)       ;DISPATCH TO DECODING SECTION FOR SPECIFIC TYPE

021702  012237  010300   RPTTXQ:  MOV     (R2)+,EVTADD      ;STORE MESSAGE ADDRESS FOR PRINTING
021706  012237  010302            MOV     (R2)+,EVTBCT      ;STORE BYTE COUNT FOR PRINTING
021712  012203                    MOV     (R2)+,R3          ;STORE MODEM STATUS FOR PRINTING
021714                            PRINTS  #EVTF2,EVTADD,EVTBCT    ;PRINT ADDR,BYTE CNT
021714  013746  010302                                                                  MOV     EVTBCT,-(SP)
021720  013746  010300                                                                  MOV     EVTADD,-(SP)
021724  012746  015176                                                                  MOV     #EVTF2,-(SP)
021730  012746  000003                                                                  MOV     #3,-(SP)
021734  010600                                                                          MOV     SP,R0
021736  104416                                                                          TRAP    C$PNTS
021740  062706  000010                                                                  ADD     #10,SP
021744  004737  022330            JSR     PC,RPTMSB         ;GO PRINT MODEM STATUS
021750  000137  021530            JMP     RPT               ;GO BACK FOR NEXT EVENT ENTRY

021754  012237  010304   RPTDER:  MOV     (R2)+,EVTTMP      ;GET ADDRESS OF DEVICE INFO MESSAGE
021760  012237  010334            MOV     (R2)+,DEV1        ;STORE DEVICE REG CONTENTS FOR PRINTING
021764  012237  010336            MOV     (R2)+,DEV2
021770                            PRINTS  #EVTF3,EVTTMP     ;PRINT DEVICE REG CONTENTS.
021770  013746  010304                                                                  MOV     EVTTMP,-(SP)
021774  012746  015250                                                                  MOV     #EVTF3,-(SP)
022000  012746  000002                                                                  MOV     #2,-(SP)
022004  010600                                                                          MOV     SP,R0
022006  104416                                                                          TRAP    C$PNTS
022010  062706  000006                                                                  ADD     #6,SP
022014                            PRINTS  #EVTF3C,DEV1,DEV2
022014  013746  010336                                                                  MOV     DEV2,-(SP)
022020  013746  010334                                                                  MOV     DEV1,-(SP)
022024  012746  015262                                                                  MOV     #EVTF3C,-(SP)
022030  012746  000003                                                                  MOV     #3,-(SP)
022034  010600                                                                          MOV     SP,R0
022036  104416                                                                          TRAP    C$PNTS
022040  062706  000010                                                                  ADD     #10,SP
022044  000137  021530            JMP     RPT               ;GO BACK FOR NEXT EVENT ENTRY

022050  005037  010334   RPTDVI:  CLR     DEV1
022054  005037  010336            CLR     DEV2              ;CLEAR UPPER BYTES OF DEV1 & DEV2 BEFORE USE
022060  112237  010334            MOVB    (R2)+,DEV1        ;STORE SETUP OPERATION PARAMETERS FOR PRINTING
022064  112237  010336            MOVB    (R2)+,DEV2
022070  012237  010340            MOV     (R2)+,DEV3
022074  012237  010342            MOV     (R2)+,DEV4
022100  010246                    MOV     R2,-(SP)          ;SAVE R2 ON THE STACK
022102  004737  023022            JSR     PC,SHWOP          ;GO PRINT MODE, MAINT-LOOP TYPE, PARAMTERS.
```

```
022106  012602                  MOV     (SP)+,R2                ;RESTORE R2
022110  000137  021530          JMP     RPT                     ;GO BACK FOR NEXT EVENT ENTRY
022114  012237  010300  RPTEOP: MOV     (R2)+,EVTADD
022120  012237  010302          MOV     (R2)+,EVTBCT
022124  012237  010304          MOV     (R2)+,EVTTMP
022130                          PRINTS  #EVTF4B,EVTADD,EVTBCT,EVTTMP    ;PRINT ADDR,RXBYTES,CMPBYTES.
022130  013746  010304                                                          MOV     EVTTMP,-(SP)
022134  013746  010302                                                          MOV     EVTBCT,-(SP)
022140  013746  010300                                                          MOV     EVTADD,-(SP)
022144  012746  015521                                                          MOV     #EVTF4B,-(SP)
022150  012746  000004                                                          MOV     #4,-(SP)
022154  010600                                                                  MOV     SP,R0
022156  104416                                                                  TRAP    C$PNTS
022160  062706  000012                                                          ADD     #12,SP

022164  000137  021530          JMP     RPT                     ;THEN GO GET NEXT EVENT ENTRY


022170  012237  010300  RPTDDE: MOV     (R2)+,EVTADD            ;STORE MESSAGE ADDRESS FOR PRINTING
022174  012237  010302          MOV     (R2)+,EVTBCT            ;STORE BYTE COUNT FOR PRINTING
022200  012237  010304          MOV     (R2)+,EVTTMP            ;STORE TOTAL # OF CMP ERRORS
022204                          PRINTS  #EVTF4,EVTADD,EVTBCT,EVTTMP     ;PRINT ADDR, BYTE CNT, # CMP ERRS
022204  013746  010304                                                          MOV     EVTTMP,-(SP)
022210  013746  010302                                                          MOV     EVTBCT,-(SP)
022214  013746  010300                                                          MOV     EVTADD,-(SP)
022220  012746  015321                                                          MOV     #EVTF4,-(SP)
022224  012746  000004                                                          MOV     #4,-(SP)
022230  010600                                                                  MOV     SP,R0
022232  104416                                                                  TRAP    C$PNTS
022234  062706  000012                                                          ADD     #12,SP
022240  000137  021530          JMP     RPT                     ;THEN GO GET NEXT EVENT ENTRY

022244                  RPTDLE:
022244  012237  010300  RPTDCK: MOV     (R2)+,EVTADD            ;STORE MSG ADDR FOR PRINT
022250  012237  010302          MOV     (R2)+,EVTBCT            ;STORE BYTE COUNT
022254  012237  010304          MOV     (R2)+,EVTTMP            ;STORE BYTE COUNT COMP
022260                          PRINTS  #EVTF4A,EVTADD,EVTBCT,EVTTMP    ;PRINT ADDR,RXBYTES,CMPBYTES.
022260  013746  010304                                                          MOV     EVTTMP,-(SP)
022264  013746  010302                                                          MOV     EVTBCT,-(SP)
022270  013746  010300                                                          MOV     EVTADD,-(SP)
022274  012746  015423                                                          MOV     #EVTF4A,-(SP)
022300  012746  000004                                                          MOV     #4,-(SP)
022304  010600                                                                  MOV     SP,R0
022306  104416                                                                  TRAP    C$PNTS
022310  062706  000012                                                          ADD     #12,SP

022314  000137  021530          JMP     RPT                     ;THEN GO GET NEXT EVENT ENTRY

022320  012604          ENDEVT: MOV     (SP)+,R4                ;RESTORE R4,R3,R2
022322  012603                  MOV     (SP)+,R3
022324  012602                  MOV     (SP)+,R2
022326  000207                  RTS     PC                      ;RETURN TO CALLING ROUTINE



                        ;REPORT MODEM STATUS SUBROUTINE
                        ;       PART OF STATISICAL REPORTING (DUMPING EVENT LOG)
```

```

022330                           RPTMSB: PRINTS  #EVMOHD                  ;PRINT MODEM STATUS HEADER
022330  012746  015707                                                                            MOV     #EVMOHD,-(SP)
022334  012746  000001                                                                            MOV     #1,-(SP)
022340  010600                                                                                    MOV     SP,R0
022342  104416                                                                                    TRAP    C$PNTS
022344  062706  000004                                                                            ADD     #4,SP
022350  012704  010210                   MOV     #MOBITS,R4      ;MAKE R4 A POINTER TO MODEM SIG. BIT DEF. TABLE
022354  012705  010226                   MOV     #MOMSGS,R5      ;MAKE R5 A POINTER TO MODEM MSG. POSITION TABLE
022360  005714                   6$:     TST     (R4)            ;SEE IF BIT AVAIABLE FROM DEVICE
022362  001004                           BNE     7$              ;BR IF THAT MODEM SIG. AVAIABLE
022364  112735  000130                   MOVB    #'X,@(R5)+      ;ELSE PUT 'X' IN REPORT IF SIGNAL NOT AVAILABLE
022370  005724                           TST     (R4)+           ;BUMP R4 TO POINT TO NEXT BIT DEFINITION
022372  000407                           BR      9$              ;GO SEE IF CHECKED ALL MODEM SIGNALS
022374  032403                   7$:     BIT     (R4)+,R3        ;IF THERE, SEE IF THAT BIT IN DEVICE'S ENTRY=1
022376  001403                           BEQ     8$              ;BR IF BIT (SIGNAL) VALUE =0
022400  112735  000061                   MOVB    #'1,@(R5)+      ;IF=1, PUT '1' IN REPORT MESSAGE
022404  000402                           BR      9$              ;GO SEE IF ALL MODEM SIGNALS CHECKED
022406  112735  000060           8$:     MOVB    #'0,@(R5)+      ;IF BIT(SIGNAL)=0, PUT '0' IN REPORT MESSAGE
022412  020427  010226           9$:     CMP     R4,#MOBITE      ;SEE IF ALL BITS(SIGNALS) CHECKED
022416  002760                           BLT     6$              ;LOOP UNTIL ALL SIGNALS(BITS) CHECKED
022420                                   PRINTS  #EVMOST         ;THEN PRINT MODEM SIGNAL VALUE MESSAGE
022420  012746  015767                                                                            MOV     #EVMOST,-(SP)
022424  012746  000001                                                                            MOV     #1,-(SP)
022430  010600                                                                                    MOV     SP,R0
022432  104416                                                                                    TRAP    C$PNTS
022434  062706  000004                                                                            ADD     #4,SP
022440  000207                           RTS     PC              ;RETURN TO EVENT DECODING


```

```
                                        .SBTTL          DUMP BYTES OR WORDS


                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       DUMPSR - DUMP BYTES OR WORDS SUBROUTINE
                                        ;
                                        ;       THIS SUBROUTINE PRINTS THE CONTENTS OF THE LOCATIONS BETWEEN
                                        ;       A STARTING AND END ADDRESS IN LOCS. ''STADD'' AND ''ENADD''.
                                        ;       THE WORD OR BYTE CONTENTS ARE PRINTED 8 TO A LINE WITH THE
                                        ;       ADDRESS OF THE FIRST BYTE AS THE FIRST 6 OCTAL CHARS. FOLLOWED
                                        ;       BY A SEMICOLON.
                                        ;
                                        ; INPUTS:
                                        ;       STADD=  STARTING ADDRESS (FIRST LOC. TO PRINT)
                                        ;       ENADD=  END ADDRESS (LAST LOCATION TO DUMP)
                                        ;       BYTBIT= 1 IF SUPPOSED TO PRINT ''BYTES''
                                        ;               0 IF SUPPOSED TO PRINT ''WORDS''
                                        ;
                                        ; OUTPUTS:
                                        ;       CONTENTS OF A RANGE OF LOC.S PRINTED ON THE OPERATORS CONSOLE.
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;       JSR PC,DUMPSR           ;CALL DUMP BYTES SUBROUTINE
                                        ;
                                        ;--

022442  013702  007146                  DUMPSR: MOV     STADD,R2        ;SET R2 UP TO STARTING ADDR.
022446  005003                          DUM4:   CLR     R3              ;CLEAR R3
022450                                          PRINTF  #BASM1,R2       ;PRINT ADDRESS
022450  010246                                                                                  MOV     R2,-(SP)
022452  012746  015010                                                                          MOV     #BASM1,-(SP)
022456  012746  000002                                                                          MOV     #2,-(SP)
022462  010600                                                                                  MOV     SP,R0
022464  104417                                                                                  TRAP    C$PNTF
022466  062706  000006                                                                          ADD     #6,SP
022472  005737  007152                  DUM3:   TST     BYTBIT          ;IS THIS BYTE OR WORD
022476  001416                                  BEQ     DUM1            ;BR IF WORD
022500  112237  007172                          MOVB    (R2)+,TEMP      ;MOV BYTE TO TEMP
022504                                          PRINTF  #BASM3,<B,TEMP> ;PRINT BYTE
022504  005046                                                                                  CLR     -(SP)
022506  153716  007172                                                                          BISB    TEMP,(SP)
022512  012746  014772                                                                          MOV     #BASM3,-(SP)
022516  012746  000002                                                                          MOV     #2,-(SP)
022522  010600                                                                                  MOV     SP,R0
022524  104417                                                                                  TRAP    C$PNTF
022526  062706  000006                                                                          ADD     #6,SP
022532  000411                                  BR      DUM2
022534                                  DUM1:   PRINTF  #BASM2,(R2)+    ;PRINT WORD
022534  012246                                                                                  MOV     (R2)+,-(SP)
022536  012746  015001                                                                          MOV     #BASM2,-(SP)
022542  012746  000002                                                                          MOV     #2,-(SP)
022546  010600                                                                                  MOV     SP,R0
022550  104417                                                                                  TRAP    C$PNTF
022552  062706  000006                                                                          ADD     #6,SP
022556  020237  007150                  DUM2:   CMP     R2,ENADD        ;COMPARE FOR LAST ADD
```

```
022562  003005                          BGT     DUMEX           ;IF DONE EXIT
022564  005203                          INC     R3              ;ELSE BUMP R3
022566  022703  000010                  CMP     #8.,R3          ;HAVE WE PRINTED 8 ACCROSS
022572  001725                          BEQ     DUM4            ;IF SO GO BACK TO 4
022574  000736                          BR      DUM3            ;ELSE GO BACK AND PRINT ANOTHER
                                                                ;BYTE OR WORD
022576  000207                  DUMEX:  RTS     PC              ;RETURN TO CALLER
```

```
                                   .SBTTL          UPDATE TOTAL CHAR. COUNT SUBROUTINE

                                   ;++
                                   ; FUNCTIONAL DESCRIPTION:
                                   ;               UPDATES TOTAL CHAR. COUNT TOTCC BASED ON CURCC.
                                   ;               LAST MESSAGE IS TRUNCATED TO FIT INTO THE
                                   ;               BUFFER IF TOTAL CHAR. COUNT EXCEEDS 'BUFLIM'  A MESSAGE
                                   ;               IS PRINTED TELLING THE OPERATOR THE TRUNCATION OCCURED.
                                   ;
                                   ; INPUTS:
                                   ;               CURCC=  CHAR. COUNT OF MESSAGE BEING ADDED
                                   ;               TOTCC=  TOTAL CHAR COUNT OF BUFFER ITS BEING ADDED TO
                                   ;
                                   ; OUTPUTS:
                                   ;               MESSAGE TO OPERATOR IF MESSAGE TRUNCATED TO FIT
                                   ;
                                   ; FUNCTIONAL SIDE EFFECTS:
                                   ;               LOCATION ''TEMP'' USED FOR CALCULATIONS
                                   ; CALLING SEQUENCE:
                                   ;               JSR     PC,ADCC         ;UPDATED TOTAL CHAR. COUNT
                                   ;--


022600  063737  007156  007166     ADDCC:  ADD     CURCC,TOTCC     ;ADD CURRENT TO TOTAL
022606  022737  001000  007166             CMP     #BUFLIM,TOTCC   ; COMPARE TO 'BUFLIM'
022614  103027                             BHIS    ADDC1           ;IF NOT MORE THEN 'BUFLIM'' EXIT

                                           ; PRINT MESSAGE AND TRUNCATE COUNT

022616                                     PRINTF  #MSGTRU
022616  012746  014102                                                                           MOV     #MSGTRU,-(SP)
022622  012746  000001                                                                           MOV     #1,-(SP)
022626  010600                                                                                   MOV     SP,R0
022630  104417                                                                                   TRAP    C$PNTF
022632  062706  000004                                                                           ADD     #4,SP
022636  163737  007156  007166             SUB     CURCC,TOTCC     ;SUB CURRENT FROM TOTAL
022644  012737  001000  007172             MOV     #BUFLIM,TEMP    ;MOV 'BUFLIM'' TO TEMP
022652  163737  007166  007172             SUB     TOTCC,TEMP      ;SUB TOTAL FROM 'BUFLIM''
022660  013737  007172  007156             MOV     TEMP,CURCC      ;AND ESTABLISH NEW CURRENT
022666  063737  007156  007166             ADD     CURCC,TOTCC     ;ADD ''ADJUSTED CURRENT'' TO TOTAL CHAR. CNT.
022674  000207                     ADDC1:  RTS     PC              ;RETURN TO CALLER
```

```
                                .SBTTL          BUILD MESSAGE BUFFERS SUBROUTINE

                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;       BLDBUF--  BUILD POINTER TABLE AND BUFFERS
                                ;
                                ;       THIS SUBROUTINE ADDS A MESSAGE TO THE TRANSMIT OR EXPECT LIST
                                ;       USING THE POINTER, BYTE COUNT, AND ADDRESS PASSED TO IT.
                                ;
                                ; INPUTS:
                                ;       CURCC=  CHAR. COUNT OF MESSAGE TO BE ADDED
                                ;       CURADD= ADDRESS OF MESSAGE TO BE ADDED
                                ;       CPTR=   ADDRESS OF POINTER TABLE WORD WHERE MESSAGE POINTERS ARE
                                ;                         TO BE BUILT
                                ;       MSGTYP= VALUE TO USE AS AN INDEX TO FIND SOURCE OF MESSAGE DATA
                                ;                         INDEX INTO DMSGCT() AND DMSGAD().
                                ; OUTPUTS:
                                ;       A MESSAGE ADDED TO EITHER TXBUF OR CMPBUF
                                ;       APPROPRIATE POINTERS IN PTRTAB POINTER TABLE
                                ;
                                ; CALLING SEQUENCE:
                                ;       JSR PC,BLDBUF             ;BUILD MESSAGE IN BUFFER AND ADD PTRS.
                                ;--

022676                          BLDBUF:
022676  010246                          MOV     R2,-(SP)          ;SAVE R2 AND R3 ON THE STACK
022700  010346                          MOV     R3,-(SP)
022702  013702  007162                  MOV     CPTR,R2

022706  013722  007164          BLDB1:  MOV     CURADD,(R2)+      ;PUT CURRENT ADD ON POINTER TAB
022712  013722  007156                  MOV     CURCC,(R2)+       ;PUT CURRENT CC ON POINTER TAB
022716  010237  007162                  MOV     R2,CPTR           ;PUT UPDATED R2 BACK TO CURRENT POINT
022722  013702  007154                  MOV     MSGTYP,R2         ;GET MESSAGE TYPE TO USE AS INDEX
022726  006302                          ASL     R2                ;DOUBLE FOR WORD INDEX
022730  013737  007164  007172          MOV     CURADD,TEMP       ;MOVE CURRENT ADD TO TEMP
022736  063737  007156  007172          ADD     CURCC,TEMP        ;ADD CHAR COUNT TO IT TO GET END
022744  013703  007164                  MOV     CURADD,R3         ;SET R3 TO CURRENT START ADD
022750  016237  002150  007176  BLDB2:  MOV     DMSGCT(R2),TEMP2        ;GET BYTE COUNT
022756  016204  002176                  MOV     DMSGAD(R2),R4     ;PUT STARTING FROM ADD IN R4
022762  060437  007176                  ADD     R4,TEMP2          ;ADD IT TO TEMP2 TO GET END OF FROM
022766  112423                  BLDB3:  MOVB    (R4)+,(R3)+       ;MOV BYTE FROM PATTERN TO BUFFER
022770  020337  007172                  CMP     R3,TEMP           ;ALL DONE?
022774  001404                          BEQ     BLDBEX            ;IF SO EXIT
022776  020437  007176                  CMP     R4,TEMP2          ;IS PATTERN COUNT EXPIRED
023002  001762                          BEQ     BLDB2             ;IF SO GO START AGAIN
023004  000770                          BR      BLDB3             ;IF NOT GET ANOTHER BYTE
023006  063737  007156  007164  BLDBEX: ADD     CURCC,CURADD      ;BUMP CURADD
023014  012603                          MOV     (SP)+,R3          ;RESTORE R3 AND R2
023016  012602                          MOV     (SP)+,R2
023020  000207                          RTS     PC                ;RETURN TO CALLER
```

```
                                        .SBTTL          SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS

                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       SHWOP - SHOW MODE OF OPERATION, LOOP, QULAIFIERS
                                        ;               PRINTED ON THE OPERATOR'S CONSOLE.
                                        ;
                                        ; INPUTS:
                                        ;       DEV1=   MODE TYPE (MODTYP)
                                        ;       DEV2=   MAINT LOOP TYPE (MLTYP)
                                        ;       DEV3=   'RUN PASS'' COUNT (RPASS) - COUNT DOWN
                                        ;       DEV4=   PARAMTERS WORD (PARAM)
                                        ;
                                        ; IMPLICIT INPUTS:
                                        ;       MODES= TABLE OF ADDRESSES OF MODE NAME STRINGS
                                        ;       LOOPS= TABLE OF ADDRESSES OF LOOP TYPE NAMES
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;       JSR PC,SHWOP
                                        ;--

023022  013702  010334                  SHWOP:  MOV     DEV1,R2                 ;GET THE MODE TYPE IN R2
023026  006302                                  ASL     R2                      ;MAKE IT A WORD TABLE OFFSET
023030  016237  003260  007172                  MOV     MODES(R2),TEMP          ;GET ADDRESS OF MODE-IN-ASCII
023036  013702  010336                          MOV     DEV2,R2                 ;GET MAINTENANCE LOOP TYPE
023042  006302                                  ASL     R2
023044  012737  013457  007200                  MOV     #LP00,TEMP3             ;LOAD TEMP3 TO POINT TO ''/LOOP=''
023052  005702                                  TST     R2                      ;SEE IF /LOOP=XXXXX OR NONE
023054  001003                                  BNE     10$                     ; BR IF /LOOP= OF SOME KIND
023056  012737  013456  007200                  MOV     #LP0,TEMP3              ;IF NO LOOP THEN DON'T PRINT ''/LOOP=''
023064  016237  003276  007174          10$:    MOV     LOOPS(R2),TEMP1         ;GET ADDRESS OF LOOP-IN-ASCII
023072  013737  010340  007176                  MOV     DEV3,TEMP2              ;GET NUMBER OF PASSES
023100                                          PRINTS  #SHF0,TEMP,TEMP3,TEMP1,TEMP2
023100  013746  007176                                                                                  MOV     TEMP2,-(SP)
023104  013746  007174                                                                                  MOV     TEMP1,-(SP)
023110  013746  007200                                                                                  MOV     TEMP3,-(SP)
023114  013746  007172                                                                                  MOV     TEMP,-(SP)
023120  012746  014165                                                                                  MOV     #SHF0,-(SP)
023124  012746  000005                                                                                  MOV     #5,-(SP)
023130  010600                                                                                          MOV     SP,R0
023132  104416                                                                                          TRAP    C$PNTS
023134  062706  000014                                                                                  ADD     #14,SP

023140  005002                                  CLR     R2                      ;NOW SET UP FOR QUALIFIERS IN ASCII
023142  012737  013536  007172                  MOV     #PST,TEMP
023150  032737  000001  010342                  BIT     #STATB,DEV4             ;SEE IF /STATUS OR /NOSTATUS
023156  001003                                  BNE     1$                      ;BR IF /STATUS
023160  012737  013534  007172                  MOV     #PNST,TEMP
023166  012737  013547  007174          1$:     MOV     #PCK,TEMP1
023174  032737  000002  010342                  BIT     #DATCKB,DEV4            ;SEE IF /CHECK OR /NOCHECK
023202  001003                                  BNE     2$                      ;BR IF /CHECK
023204  012737  013545  007174                  MOV     #PNCK,TEMP1
023212  012737  013557  007176          2$:     MOV     #PEC,TEMP2
023220  032737  000004  010342                  BIT     #ECHOB,DEV4             ;SEE IF /ECHO OR /NOECHO
023226  001003                                  BNE     5$                      ;BR IF /ECHO
023230  012737  013555  007176                  MOV     #PNEC,TEMP2
```

```
023236                          5$:     PRINTS  #SHF1,TEMP,TEMP1,TEMP2          ;,TEMP3,TEMP4 **;SEE NOTE ABOVE
023236  013746  007176                                                                  MOV     TEMP2,-(SP)
023242  013746  007174                                                                  MOV     TEMP1,-(SP)
023246  013746  007172                                                                  MOV     TEMP,-(SP)
023252  012746  014223                                                                  MOV     #SHF1,-(SP)
023256  012746  000004                                                                  MOV     #4,-(SP)
023262  010600                                                                          MOV     SP,R0
023264  104416                                                                          TRAP    C$PNTS
023266  062706  000012                                                                  ADD     #12,SP
023272  000207                                  RTS     PC              ;RETURN
```

```
                                 .SBTTL          TRAVERSE COMMAND LINE SUBROUTINES

                                 ;++
                                 ;        P$TRV SUBROUITNE
                                 ;
                                 ;PARSE THE COMMAND LINE SUBROUTINE
                                 ;TAKE ACTIONS (VIA ACTION TREE) AS PARSING LINE
                                 ;PARSING DIRECTIONS FROM ''CLI PARSING NODES''
                                 ;    REGS USED:
                                 ;
                                 ;        R1,R5=SCRATCH                               P$NUM=NUMERIC CODE FROM DATA
                                 ;        R2=ACTION CODE PARAMETER FROM TREE
                                 ;        R3=PARSE TREE POINTER
                                 ;        R4=INPUT STRING POINTER
                                 ; CALLING SEQUENCE:
                                 ;        JSR     PC,P$TRV
                                 ;--

023274                           P$TRV:
023274  013704  003310                   MOV     P$BUFA,R4
023300  013703  003312                   MOV     P$TREE,R3
023304  105714                   P$TR5:  TSTB    (R4)                 ;SEE IF ANY CHARS LEFT IN INPUT STRING
023306  001441                           BEQ     P$EXIT               ;BR IF NO
023310  121327  000013                   CMPB    (R3),#11.            ;SEE IF SPECIAL CLI CHAR CODE OR ASCII
023314  003023                           BGT     20$                  ;BR IF REGULAR ASCII CHAR.
023316  111305                           MOVB    (R3),R5              ;GET SPECIAL CHAR CODE INTO R5
023320  006305                           ASL     R5
023322  016505  023336                   MOV     10$(R5),R5           ;BUILD TRAVERSE ROUTINE ADDRESS
023326  062705  023336                   ADD     #10$,R5
023332  004715                           JSR     PC,(R5)              ;JSR TO SPECIAL CLI TRAVERSE ROUTINE
023334  000763                           BR      P$TR5                ;GO SEE IF MORE OF STRING LEFT


023336  000114                   10$:    .WORD   TRVERR-10$           ;TRAVERSE TABLE FOR ''CLI FUNTIONS''
023340  000134                           .WORD   TRVEXI-10$           ;1
023342  000152                           .WORD   TRVBR-10$            ;2
023344  000162                           .WORD   TRVBIF-10$           ;3
023346  000204                           .WORD   TRVSPA-10$           ;4
023350  000270                           .WORD   TRVNUM-10$           ;5
023352  000604                           .WORD   TRVALP-10$           ;6
023354  000650                           .WORD   TRVALN-10$           ;7
023356  000270                           .WORD   TRVOCT-10$           ;8
023360  000256                           .WORD   TRVDEC-10$           ;9
023362  000736                           .WORD   TRVSTR-10$           ;10

                                 ;NOT A SPECIAL CODE

023364  121314                   20$:    CMPB    (R3),(R4)            ;SEE IF FIRST CHAR OF STRING IS A MATCH
023366  001403                           BEQ     22$                  ;BR IF A MATCH
023370  004737  023434                   JSR     PC,TRVBRC            ;IF NOT A MATCH, GO TAKE MISS BRANCH
023374  000743                           BR      P$TR5                ; THEN GO BACK PT'G TO MISS NODE
023376  004737  023414           22$:    JSR     PC,TRVACT            ;IF A MATCH, GO DO ACTION DEFINED BY
023402  062703  000004                   ADD     #4,R3                ; ACTION CODE IN CLI NODE, THEN
                                                                      ; ADJUST PTR TO NEXT CLI NODE
023406  005204                           INC     R4                   ;ADJUST BUF PTR TO NEXT CHAR IF MATCH
023410  000735                           BR      P$TR5
```

```

023412  000207                          P$EXIT: RTS     PC              ;RETURN FROM PARSER

                                        ;------------------------------------------------------------

                                        ;GOTO USER ACTION ROUTINE
023414  116302  000001                  TRVACT: MOVB    1(R3),R2        ;GET ACTION CODE FROM CLI NODE
023420  042702  177400                          BIC     #177400,R2      ;CLEAR ANY SIGN EXTENSION
023424  013705  003314                          MOV     P$ACT,R5        ;GET ADDRESS OF CLI ACTION ROUTINE
023430  004715                                  JSR     PC,(R5)         ;GO DO ACTION DEFINED BY CODE
023432  000207                                  RTS     PC              ;RETURN TO CALLING CODE

                                        ;TAKE BRANCH IN TREE
023434  016305  000002                  TRVBRC: MOV     2(R3),R5        ;GET BRANCH DISPLACEMENT FROM TREE
023440  060503                                  ADD     R5,R3           ; AND POINT R3 TO THE ''MISS'' NODE
023442  000207                                  RTS     PC              ; RETURN TO P$TRV

                                        ;NO BRANCH TAKEN
023444  062703  000004                  TRVNOB: ADD     #4,R3           ;THINGS OK, UPDATE R3 TO POINT TO NEXT
023450  000207                                  RTS     PC              ; NODE AND RETURN TO P$TRV

                                        ;------------------------------------------------------------
023452  004737  023414                  TRVERR: JSR     PC,TRVACT       ;TAKE ERROR ACTION
023456  112737  177777  003325                  MOVB    #-1,P$GDBD      ;SET ERROR RETURN FLAG
023464  005726                                  TST     (SP)+           ;GET RID OF ''JSR PUSH TO TRVERR''
023466  000137  023412                          JMP     P$EXIT          ;RETURN DIRECT TO EXIT OF P$TRV ROUTINE

023472  004737  023414                  TRVEXI: JSR     PC,TRVACT       ;TAKE EXIT ACTION
023476  105037  003325                          CLRB    P$GDBD          ;SET GOOD/BAD FLAG TO ''SUCCESS (0)''
023502  005726                                  TST     (SP)+           ;GET RID OF ''JSR PUSH TO TRVEXI''
023504  000137  023412                          JMP     P$EXIT          ;RETURN DIRECT TO EXIT OF P$TRV ROUTINE

023510  004737  023414                  TRVBR:  JSR     PC,TRVACT       ;GO TAKE BRANCH ACTION
023514  000137  023434                          JMP     TRVBRC

023520  004737  023414                  TRVBIF: JSR     PC,TRVACT
023524  105737  003325                          TSTB    P$GDBD          ;SEE IF P$GDBD SET OR CLEARED BY ACTION
023530  001402                                  BEQ     1$              ;IF CLEAR FALL THRU TO NEXT NODE
023532  000137  023434                          JMP     TRVBRC          ;ELSE TAKE THE ''MISS'' BRANCH
023536  000137  023444                  1$:     JMP     TRVNOB          ;JUST UPDATE TO NEXT NODE IF THINGS OK

023542  005005                          TRVSPA: CLR     R5              ;CLEAR ''SPACE OR TAB FOUND'' FLAG
023544  121427  000011                  1$:     CMPB    (R4),#11        ;SEE IF CHAR. IN CMD LINE= TAB
023550  001003                                  BNE     2$              ;BR IF NO, NOT A TAB
023552  005204                                  INC     R4              ;INC INPUT STRING POINTER
023554  005205                                  INC     R5              ;INDICATE A TAB FOUND
023556  000772                                  BR      1$              ;GO CHECK NEXT CHAR

023560  121427  000040                  2$:     CMPB    (R4),#40        ;SEE IF CHAR. IN CMD LINE= SPACE
023564  001003                                  BNE     10$             ;BR IF NO, NON-SPACE OR NON-TAB CHAR.
023566  005204                                  INC     R4              ;INC INPUT STRING POINTER
023570  005205                                  INC     R5              ;INDICATE A SPACE FOUND
023572  000764                                  BR      1$              ;GO CHECK NEXT CHAR
023574  005705                          10$:    TST     R5              ;SEE IF ANY SPACES OR TABS FOUND
023576  001404                                  BEQ     15$             ;BR IF NO, TAKE NO ACTION
023600  004737  023414                          JSR     PC,TRVACT       ;GO TAKE ACTION IF ANY FOUND
```

```
023604  000137  023444                          JMP     TRVNOB                  ;JUST GO UPDATE R3 TO NEXT NODE IF OK
023610  000137  023434                  15$:    JMP     TRVBRC                  ;TAKE BRANCH (MISS) IF NONE FOUND


023614  012737  000012  003322          TRVDEC: MOV     #10.,P$RADX             ;USE DECIMAL AS RADIX AND ASSUME +
023622  000137  023634                          JMP     TRVNMA
023626                                  TRVOCT: ;(SAME AS TRVNUM SINCE DEFAULT RADIX IS OCTAL)
023626  012737  000010  003322          TRVNUM: MOV     #8.,P$RADX              ;USE OCTAL AS RADIX AND ASSUME +
023634  005005                          TRVNMA: CLR     R5                      ;CLEAR DIGIT COUNTER
023636  121427  000053                          CMPB    (R4),#'+                ;SEE IF THERE'S A + SIGN THERE
023642  001001                                  BNE     10$                     ; BR IF NO
023644  000406                                  BR      11$                     ; ELSE P$RADX ALREADY SAYS +, JUST BR
023646  121427  000055                  10$:    CMPB    (R4),#'-                ;SEE IF THERE'S A - SIGN THERE
023652  001004                                  BNE     1$                      ; BR IF NO
023654  112737  177777  003323                  MOVB    #-1,P$RADX+1            ;SET 'MINUS FLAG'' (HI BYTE OF P$RADX)
023662  005204                          11$:    INC     R4                      ;BUMP R4 TO POINT TO FIRST CHAR

023664  121427  000060                  1$:     CMPB    (R4),#60                ;SEE IF CHAR. LESS THAN A '0''
023670  002434                                  BLT     2$                      ;BR IF YES (NOT NUMERIC)
023672  121427  000067                          CMPB    (R4),#67                ;SEE IF CHAR. GREATER THAN A '7''
023676  003426                                  BLE     13$                     ; BR IF YES
023700  123727  003322  000012                  CMPB    P$RADX,#10.             ;SEE IF IN DECIMAL MODE
023706  001417                                  BEQ     12$                     ; BR IF YES (CAN USE HIGHER LIMIT)
023710  121427  000071                          CMPB    (R4),#71                ;SEE IF DIGIT WAS A 8 OR 9
023714  003022                                  BGT     2$                      ;BR IF NON-NUMERIC
023716                                          PRINTF  #CLIBRX                 ;ELSE WAS A 8 OR 9 WHEN IN OCTAL RADIX
023716  012746  012217                                                                  MOV     #CLIBRX,-(SP)
023722  012746  000001                                                                  MOV     #1,-(SP)
023726  010600                                                                          MOV     SP,R0
023730  104417                                                                          TRAP    C$PNTF
023732  062706  000004                                                                  ADD     #4,SP
023736  112737  177777  003325                  MOVB    #-1,P$GDBD              ;SET ERROR RETURN FLAG
023744  000474                                  BR      5$                      ; PRINT ERROR AND TAKE MISS

023746  121427  000071                  12$:    CMPB    (R4),#71                ;SEE IF CHAR. GREATER THAN A '9''
023752  003003                                  BGT     2$                      ;BR IF YES (NOT NUMERIC)
023754  005204                          13$:    INC     R4                      ;UPDATE CMD LINE PTR TO NEXT CHAR.
023756  005205                                  INC     R5                      ;INDICATE A NUMERIC FOUND
023760  000741                                  BR      1$                      ;GO LOOK AT NEXT CHAR.

023762  005705                          2$:     TST     R5                      ;SEE IF FOUND ANY NUMERICS
023764  001464                                  BEQ     5$                      ;BR IF NO, TAKE 'MISS'' BRANCH
023766  010401                                  MOV     R4,R1                   ;GET POINTER TO START OF NUMERIC STRING
023770  160501                                  SUB     R5,R1
023772  005037  003320                          CLR     P$NUM                   ;CLEAR LOC. WHERE VALUE WILL BE STORED
023776  112102                          3$:     MOVB    (R1)+,R2                ;GET ASCII CHAR AND CONVERT IT TO A #
024000  162702  000060                          SUB     #60,R2
024004  006337  003320                          ASL     P$NUM                   ;SHIFT CURRENT VALUE TO MAKE ROOM
024010  103437                                  BCS     7$                      ;ERROR IF NUMBER TOO BIG
024012  013737  003320  003316                  MOV     P$NUM,P$CNT             ;SAVE FOR LATER IN CASE DECIMAL RADIX
024020  006337  003320                          ASL     P$NUM
024024  103431                                  BCS     7$                      ;ERROR IF NUMBER TOO BIG
024026  006337  003320                          ASL     P$NUM
024032  103426                                  BCS     7$                      ;ERROR IF NUMBER TOO BIG
024034  123727  003322  000012                  CMPB    P$RADX,#10.             ;SEE IF DECIMAL RADIX
024042  001004                                  BNE     4$                      ;BR IF NOT EQUAL
```

```
024044  063737  003316  003320            ADD     P$CNT,P$NUM
024052  103416                            BCS     7$                      ;ERROR IF NUMBER TOO BIG
024054  060237  003320            4$:     ADD     R2,P$NUM
024060  103413                            BCS     7$                      ;ERROR IF NUMBER TOO BIG
024062  005305                            DEC     R5
024064  001344                            BNE     3$
024066  105737  003323                    TSTB    P$RADX+1                ;SEE IF NUM WAS PRECEDED BY A - SIGN
024072  001402                            BEQ     15$                     ; BR IF NO
024074  005437  003320                    NEG     P$NUM                   ; ELSE NEGATE THE NUMBER BEFORE LEAVING
024100  004737  023414            15$:    JSR     PC,TRVACT               ;SINCE NUMERIC FOUND, GO TAKE ACTION
024104  000137  023444                    JMP     TRVNOB                  ;GO POINT R3 TO NEXT NODE

024110                            7$:     PRINTF  #CLINBG                 ;PRINT NUMBER TOO BIG ERROR
024110  012746  012175                                                                    MOV     #CLINBG,-(SP)
024114  012746  000001                                                                    MOV     #1,-(SP)
024120  010600                                                                            MOV     SP,R0
024122  104417                                                                            TRAP    C$PNTF
024124  062706  000004                                                                    ADD     #4,SP
024130  112737  177777  003325            MOVB    #-1,P$GDBD              ;SET ERROR RETURN FLAG
024136  000137  023434            5$:     JMP     TRVBRC                  ;TAKE 'MISS'' BRANCH


024142  005005                    TRVALP: CLR     R5                      ;CLEAR ALPHA FOUND FLAG
024144  121427  000101            1$:     CMPB    (R4),#101               ;SEE IF CHAR. LESS THAN A ''A''
024150  002406                            BLT     2$                      ;BR IF YES (NOT ALPHA)
024152  121427  000132                    CMPB    (R4),#132               ;SEE IF CHAR. GREATER THAN A 'Z''
024156  003003                            BGT     2$                      ;BR IF YES (NOT ALPHA)
024160  005204                            INC     R4                      ;UPDATE CMD LINE PTR TO NEXT CHAR
024162  005205                            INC     R5                      ;INDICATE AN ALPHA WAS FOUND
024164  000767                            BR      1$                      ;GO LOOK AT NEXT CHAR.
024166  005705                    2$:     TST     R5                      ;SEE IF ANY ALPHA'S WERE FOUND
024170  001404                            BEQ     3$                      ;BR IF NO
024172  004737  023414                    JSR     PC,TRVACT               ;IF ANY FOUND TAKE ACTION
024176  000137  023444                    JMP     TRVNOB                  ;THEN UPDATE R3 TO NEXT NODE -NO BRANCH
024202  000137  023434            3$:     JMP     TRVBRC                  ;NONE FOUND, TAKE MISS BRANCH

024206  005005                    TRVALN: CLR     R5                      ;CLEAR ALPHANUM FOUND FLAG
024210  121427  000060            10$:    CMPB    (R4),#60                ;SEE IF CHAR. LESS THAN A '0''
024214  002417                            BLT     2$                      ;BR IF YES (NOT NUMERIC OR ALPHA)
024216  121427  000072                    CMPB    (R4),#72                ;SEE IF CHAR. GREATER THAN A '9''
024222  003003                            BGT     1$                      ;BR IF YES (NOT NUMERIC)
024224  005204                            INC     R4                      ;UPDATE CMD LINE PTR TO NEXT CHAR.
024226  005205                            INC     R5                      ;INDICATE A NUMERIC FOUND
024230  000767                            BR      10$                     ;GO LOOK AT NEXT CHAR.
024232  121427  000101            1$:     CMPB    (R4),#101               ;SEE IF CHAR. LESS THAN A ''A''
024236  002406                            BLT     2$                      ;BR IF YES (NOT ALPHA)
024240  121427  000132                    CMPB    (R4),#132               ;SEE IF CHAR. GREATER THAN A 'Z''
024244  003003                            BGT     2$                      ;BR IF YES (NOT ALPHA)
024246  005204                            INC     R4                      ;UPDATE CMD LINE PTR TO NEXT CHAR
024250  005205                            INC     R5                      ;INDICATE AN ALPHA FOUND
024252  000756                            BR      10$                     ;GO LOOK AT NEXT CHAR.
024254  005705                    2$:     TST     R5                      ;SEE IF ANY ALPHANUM'S WERE FOUND
024256  001404                            BEQ     3$                      ;BR IF NO
024260  004737  023414                    JSR     PC,TRVACT               ;IF ANY FOUND TAKE ACTION
024264  000137  023444                    JMP     TRVNOB                  ;THEN UPDATE R3 TO NEXT NODE -NO BRANCH
024270  000137  023434            3$:     JMP     TRVBRC                  ;NONE FOUND, TAKE MISS BRANCH
```

```



024274  010401          TRVSTR: MOV     R4,R1                   ;POINT R1 TO CMD STRING
024276  010305                  MOV     R3,R5
024300  062705  000006          ADD     #6,R5                   ;POUNT R5 TO MATCH STRING FROM CLI NODE
024304  005037  003316          CLR     P$CNT                   ;CLEAR CHAR MATCH COUNT
024310  105715          2$:     TSTB    (R5)                    ;SEE IF END OF MATCH STRING YET
024312  001411                  BEQ     10$                     ;BR IF YES
024314  105711                  TSTB    (R1)                    ;SEE IF END OF CMD LINE YET
024316  001407                  BEQ     10$                     ;BR IF YES
024320  121115                  CMPB    (R1),(R5)               ;SEE IF CHARACTERS MATCH
024322  001005                  BNE     10$                     ;BR IF NO
024324  005237  003316          INC     P$CNT                   ;MATCH -INCREMENT MATCH COUNT
024330  005201                  INC     R1                      ;UPDATE STRING POINTERS
024332  005205                  INC     R5
024334  000765                  BR      2$                      ;BR TO CONTINUE CHECKING CHARS.

024336  005737  003316  10$:    TST     P$CNT                   ;WHEN DONE SEE IF ANY MATCHES FOUND
024342  001406                  BEQ     15$                     ;BR IF NO, GO TAKE THE MISS BRANCH
024344  010104                  MOV     R1,R4                   ;POINT CMD POINTER TO END OF STRING &
024346  004737  023414          JSR     PC,TRVACT               ;IF A MATCH FOUND, GO DO MATCH ACTION
024352  066303  000004          ADD     4(R3),R3                ;UPDATE R3 TO NEXT NODE (NO BRANCH)
024356  000207                  RTS     PC                      ; (NO RETURN THRU TRVNOB SINCE DIFFERNT
                                                                ;  DISPLACEMENT DUE TO MATCH STRING)
024360  000137  023434  15$:    JMP     TRVBRC                  ; GO TAKE BRANCH

                                                                ; (PARSED OK), -1 IF ILL CMD.....
                        ;-----------------------------------------------------------------------------

```

```
                                   .SBTTL  REPORT CODING SECTION



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

024364                                     BGNRPT
024364                                                                      L$RPT::


024364  004737  021406                     JSR     PC,REPORT           ;CALL SUBROUTINE TO DUMP EVENT LOG
                                                                       ; AND BASE TABLE




024370                                     ENDRPT
024370                                                                      L10011:
024370  104425                                                                    TRAP    C$RPT
```

```
 4134                                .SBTTL  PROTECTION TABLE
 4135
 4136                                ;++
 4137                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
 4138                                ; TO PROTECT THE LOAD MEDIA.
 4139                                ;--
 4140
 4141  024372                               BGNPROT
 4142  024372                                                                  L$PROT::
 4143
 4144  024372  177777                       -1              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
 4145  024374  177777                       -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
 4146  024376  177777                       -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER
 4147
 4148  024400                               ENDPROT
 4149
```

```
                                .SBTTL  INITIALIZE SECTION

                                ;++
                                ; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
                                ; AT THE BEGINNING OF EACH PASS.
                                ;--

024400                                  BGNINIT
024400                                                                                  L$INIT::


024400  012737  177777  007216          MOV     #-1,RESFLG              ;SET RESTART FLAG
024406                                  READEF  #EF.START               ;IF HERE CAUSE OF START,DO SOME INIT
024406  012700  000040                                                                         MOV     #EF.START,R0
024412  104447                                                                                 TRAP    C$REFG
024414                                  BCOMPLETE       START
024414  103417                                                                                 BCS     START
024416                                  READEF  #EF.RESTART             ;IF HERE CAUSE OF RESTART, DO SOME INIT
024416  012700  000037                                                                         MOV     #EF.RESTART,R0
024422  104447                                                                                 TRAP    C$REFG
024424                                  BCOMPLETE       RESTRT
024424  103515                                                                                 BCS     RESTRT
024426                                  READEF  #EF.CONTINUE            ;SEE IF WE'RE HERE CAUSE OF A CONTINUE
024426  012700  000036                                                                         MOV     #EF.CONTINUE,R0
024432  104447                                                                                 TRAP    C$REFG
024434                                  BNCOMPLETE      S1              ;BR IF NOT HERE CAUSE OF CONITNUE
024434  103002                                                                                 BCC     S1
024436  000137  025146                  JMP     ENDIT                   ;JMP IF HERE CAUSE OF A CONTINUE
024442                          S1:     READEF  #EF.NEW                 ;SEE IF THIS IS A 'NEW PASS''
024442  012700  000035                                                                         MOV     #EF.NEW,R0
024446  104447                                                                                 TRAP    C$REFG
024450                                  BCOMPLETE       NEW             ;IF YES, BR AROUND LOGUNIT # SETUP
024450  103523                                                                                 BCS     NEW
024452  000525                          BR      GETPRM

024454  005037  007216          START:  CLR     RESFLG                  ;CLEAR RESTART FLAG SINCE HERE ON START
024460  005037  007256                  CLR     CLKVEC                  ;CLEAR CLK VECTOR PTR. AS A FLAG IN
                                                                        ; NO CLOCK IS FOUND.
024464  012702  007252                  MOV     #CLKCSR,R2              ;SETUP R2 AS A PTR. TO CLOCK INFO BLOCK
024470                                  CLOCK   L,R1                    ;LOOK FOR A LINE CLOCK
024470  012700  000114                                                                         MOV     #'L,R0
024474  104462                                                                                 TRAP    C$CLCK
024476  010001                                                                                 MOV     R0,R1
024500                                  BNCOMPLETE      S2              ; IF NONE THERE GO LOOK FOR A P-CLOCK
024500  103006                                                                                 BCC     S2
024502  004737  020544                  JSR     PC,CLKSET               ; GO SET UP CLOCK INFO TABLE & CLK VEC.
024506  012737  000100  007262          MOV     #LCLKEN,CLKEN           ;SETUP THE ENABLE LINE CLOCK DATA
024514  000461                          BR      RESTRT

024516                          S2:     CLOCK   P,R1                    ;LOOK FOR A P-CLOCK SINCE NO LINE CLOCK
024516  012700  000120                                                                         MOV     #'P,R0
024522  104462                                                                                 TRAP    C$CLCK
024524  010001                                                                                 MOV     R0,R1
024526                                  BNCOMPLETE      S3              ; IF NONE THERE GO SEE IF THIS IS LSI
024526  103017                                                                                 BCC     S3
024530  004737  020544                  JSR     PC,CLKSET               ; ELSE GO SET UP CLOCK INFO & VECTOR
```

```
024534 062737 000002 007252           ADD     #2,CLKCSR               ;POINT CLKCSR TO P-CLK COUNT SET REG.
024542 012777 001600 162502           MOV     #PCLKCT,@CLKCSR         ;LOAD CLK SET REG. WITH COUNT VALUE
024550 162737 000002 007252           SUB     #2,CLKCSR               ;POINT CLKCSR BAC TO P-CLK CSR
024556 012737 000111 007262           MOV     #PCLKEN,CLKEN           ;SETUP THE ENABLE THE P-CLK DATA
024564 000435                         BR      RESTRT

024566                        S3:     READBUS                         ;READ BUS TYPE TO SEE IF ON AN LSI
024566 104407                                                                   TRAP    C$RDBU
024570                                BNCOMPLETE      S4              ;BR IF NOT, NO CHANCE OF A CLOCK
024570 103021                                                                   BCC     S4
024572 012737 000100 007256           MOV     #100,CLKVEC             ;LOAD 100 AS CLK VECTOR
024600 005037 007254                  CLR     CLKBR                   ;LOAD 0 AS CLK INT. LEVEL
024604 012737 007262 007252           MOV     #CLKEN,CLKCSR           ;KLUDGE UP THE CSR & ENABLE DATA LOCS
024612                                GMANID  L5060,CLKHZ,D,377,50.,60.,YES
024612 104443                                                                   TRAP    C$GMAN
024614 000406                                                                   BR      10000$
024616 007260                                                                   .WORD   CLKHZ
024620 000052                                                                   .WORD   T$CODE
024622 013602                                                                   .WORD   L5060
024624 000377                                                                   .WORD   377
024626 000062                                                                   .WORD   T$LOLIM
024630 000074                                                                   .WORD   T$HILIM
024632                                                                  10000$:
024632 000412                         BR      RESTRT

024634                        S4:     PRINTF  #NOCLK                  ;INFORM OPR. NO CLOCK, & EXIT INIT
024634 012746 013712                                                            MOV     #NOCLK,-(SP)
024640 012746 000001                                                            MOV     #1,-(SP)
024644 010600                                                                   MOV     SP,R0
024646 104417                                                                   TRAP    C$PNTF
024650 062706 000004                                                            ADD     #4,SP
024654                                EXIT    INIT
024654 104432                                                                   TRAP    C$EXIT
024656 000404                                                                   .WORD   L10013-.

024660 005037 007264          RESTRT: CLR     TIMMIN                  ;CLEAR TIME SINCE START LOCATIONS
024664 005037 007266                  CLR     TIMSEC
024670 013737 007260 007270           MOV     CLKHZ,TIMTCK            ;LOAD TICKS/SEC
024676 012702 007302                  MOV     #EVTLOG,R2              ;INIT EVENT TABLE TO ALL 1'S AFTER EACH
024702 010237 007300                  MOV     R2,EVTPTR               ; START OR RES AND INIT TABLE POINTER
024706 012722 177777          1$:     MOV     #-1,(R2)+
024712 020227 010204                  CMP     R2,#EVTEND              ;SEE IF REACHED END OF TABLE
024716 001373                         BNE     1$                      ;LOOP UNTIL DONE

024720 012737 177777 007212   NEW:    MOV     #-1,LOGUNT              ;INITIALIZE LOGICAL UNIT #

024726 005237 007212          GETPRM: INC     LOGUNT                  ;POINT TO NEXT LOGICAL UNIT
024732 023737 007212 002012           CMP     LOGUNT,L$UNIT           ;SEE IF PAST MAX. LOG. UNIT #
024740 002367                         BGE     NEW                     ;BR IF YES, AND START OVER

024742                                GPHARD  LOGUNT,R1               ;GET THE P-TABLE FOR THIS LOG. UNIT
024742 013700 007212                                                            MOV     LOGUNT,R0
024746 104442                                                                   TRAP    C$GPHRD
024750 010001                                                                   MOV     R0,R1
024752                                BNCOMPLETE      GETPRM          ;IF NO P-TABLE AVAIL., GO GET NEXT ONE
024752 103365                                                                   BCC     GETPRM
```

```

024754  011137  007224                   MOV     (R1),FHDPLX                ;PUT FULL OR HALF DUPLEX ANSWER IN LOC.



                                         ;DEVICE DEPENDENT PART OF GETTING INFO FROM P-TABLE

024760  016137  000002  011776           MOV     2(R1),SEL0                 ;STORE AWAY CSR ADDRESSES
024766  016137  000002  012000           MOV     2(R1),BSEL1
024774  005237  012000                   INC     BSEL1
025000  016137  000002  012002           MOV     2(R1),SEL2
025006  062737  000002  012002           ADD     #2,SEL2
025014  016137  000002  012004           MOV     2(R1),BSEL3
025022  062737  000003  012004           ADD     #3,BSEL3
025030  016137  000002  012006           MOV     2(R1),SEL4
025036  062737  000004  012006           ADD     #4,SEL4
025044  016137  000002  012010           MOV     2(R1),BSEL5
025052  062737  000005  012010           ADD     #5,BSEL5
025060  016137  000002  012012           MOV     2(R1),SEL6
025066  062737  000006  012012           ADD     #6,SEL6
025074  016137  000002  012014           MOV     2(R1),BSEL7
025102  062737  000007  012014           ADD     #7,BSEL7

025110  016137  000004  012016           MOV     4(R1),INVEC                ;STORE AWAY INPUT INTERRUPT VECTOR
025116  016137  000004  012020           MOV     4(R1),OUTVEC
025124  062737  000004  012020           ADD     #4,OUTVEC                  ;BUILD OUTPUT INTERRUPT VECTOR
025132  016137  000006  012022           MOV     6(R1),INTPRI               ;STORE AWAY INTERRUPT PRIORITY
025140  016137  000012  012024           MOV     12(R1),OPTYP               ;STORE AWAY DEVICE OPTION TYPE

025146                           ENDIT:
025146                                   SETVEC  CLKVEC,#CLKINT,#340        ;SETUP CLOCK VECTOR
025146  012746  000340                                                                              MOV     #340,-(SP)
025152  012746  020570                                                                              MOV     #CLKINT,-(SP)
025156  013746  007256                                                                              MOV     CLKVEC,-(SP)
025162  012746  000003                                                                              MOV     #3,-(SP)
025166  104437                                                                                      TRAP    C$SVEC
025170  062706  000010                                                                              ADD     #10,SP

                                         ;DEVICE DEPENDENT VECTOR SETUP

025174                                   SETVEC  INVEC,#DVINS,INTPRI        ;SETUP INPUT INTERRUPT VECTOR
025174  013746  012022                                                                              MOV     INTPRI,-(SP)
025200  012746  034700                                                                              MOV     #DVINS,-(SP)
025204  013746  012016                                                                              MOV     INVEC,-(SP)
025210  012746  000003                                                                              MOV     #3,-(SP)
025214  104437                                                                                      TRAP    C$SVEC
025216  062706  000010                                                                              ADD     #10,SP
025222                                   SETVEC  OUTVEC,#DVOUTS,INTPRI      ;SETUP OUTPUT INTERRUPT VECTOR
025222  013746  012022                                                                              MOV     INTPRI,-(SP)
025226  012746  034710                                                                              MOV     #DVOUTS,-(SP)
025232  013746  012020                                                                              MOV     OUTVEC,-(SP)
025236  012746  000003                                                                              MOV     #3,-(SP)
025242  104437                                                                                      TRAP    C$SVEC
025244  062706  000010                                                                              ADD     #10,SP

025250                                   SETPRI  #PRI00                     ;SET THE 'RUN' PRIORITY TO 0
025250  012700  000000                                                                              MOV     #PRI00,R0
```

```
 4318  025254  104441                                                      TRAP    C$SPRI
 4319  025256                        EXIT    INIT
 4320  025256  104432                                                      TRAP    C$EXIT
 4321  025260  000002                                                      .WORD   L10013-.
 4322
 4323
 4324                                .EVEN
 4325
 4326  025262                        ENDINIT
 4327  025262                                                    L10013:
 4328  025262  104411                                                      TRAP    C$INIT
```

```
                                    .SBTTL  AUTODROP SECTION

                                    ;++
                                    ; THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
                                    ; THE ''ADR'' FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE CHECKED TO
                                    ; SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                                    ; DROPPED FROM TESTING.
                                    ;--

025264                                      BGNAUTO
025264                                                                      L$AUTO::


025264                                      ENDAUTO
025264                                                                      L10014:
025264  104461                                                                      TRAP    C$AUTO
```

```
                                  .SBTTL  CLEANUP CODING SECTION

                                  ;++
                                  ; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
                                  ; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
                                  ;--

025266                                    BGNCLN
025266                                                                        L$CLEAN::

025266  005077  161760                    CLR     @CLKCSR          ;DISABLE CLOCK
025272                                    SETPRI  #PRI07           ;SET PROCESSOR PRIORITY BACK TO 7
025272  012700  000340                                                                MOV     #PRI07,R0
025276  104441                                                                        TRAP    C$SPRI
025300                                    BRESET                   ;CLEAR ALL BEFORE END
025300  104433                                                                        TRAP    C$RESET

025302                                    EXIT    CLN
025302  104432                                                                        TRAP    C$EXIT
025304  000002                                                                        .WORD   L10015-.


                                          .EVEN

025306                                    ENDCLN
025306                                                                        L10015:
025306  104412                                                                        TRAP    C$CLEAN
```

```
                                        .SBTTL  DROP UNIT SECTION

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

025310                                          BGNDU
025310                                                                          L$DU::


025310                                          EXIT    DU
025310  000167                                                                          .WORD   J$JMP
025312  000000                                                                          .WORD   L10016-2-.


                                                .EVEN

025314                                          ENDDU
025314                                                                          L10016:
025314  104453                                                                          TRAP    C$DU
```

```
                                .SBTTL  ADD UNIT SECTION

                                ;++
                                ; THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
                                ;  TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
                                ;  TO THE TEST CYCLE.
                                ;--

025316                                  BGNAU
025316                                                                  L$AU::


025316                                  EXIT    AU
025316  000167                                                                  .WORD   J$JMP
025320  000000                                                                  .WORD   L10017-2-.


                                        .EVEN

025322                                  ENDAU
025322                                                                  L10017:
025322  104452                                                                  TRAP    C$AU


```

```
                                .SBTTL  TEST 1:  SETUP AND MODES OF OPERATION


                                ;++
                                ; TEST TO DETECT FAULTS IN THE DATA COMMUNICATION LINK.  THIS TEST WILL
                                ; THE PROVIDE COVERAGE NECESSARY TO ISOLATE FAILURES TO THE COMPUTER
                                ; EQUIPMENT, THE COMMUNICATION LINK, OR THE MODEM.
                                ;--



025324                                  BGNTST
025324                                                                                  T1::


                                .SBTTL          PROGRAM SETUP SECTION

025324  013777  007262  161720          MOV     CLKEN,@CLKCSR   ;ENABLE THE CLOCK

025332                          GTXRXB:
025332  005001                  GTRA2:  CLR     R1
025334  012737  000001  007272          MOV     #1,TIMER1       ;SET TIMER TO COUNT 1 TICK
025342  005737  007272          1$:     TST     TIMER1          ;CHECK FOR IT TO BE COUNTED OFF
025346  001412                          BEQ     GTRA3           ;BRANCH IF CLOCK EXISTS (COUNTED A TICK)
025350  005301                          DEC     R1
025352  001373                          BNE     1$              ;KEEP CHECKING UNTIL R1 DOES FULL COUNTDWN
025354                                  PRINTF  #NOCLK          ;PRINT BAD CLK MSG AND WARN OF HANG IF TIMEOUT
025354  012746  013712                                                                          MOV     #NOCLK,-(SP)
025360  012746  000001                                                                          MOV     #1,-(SP)
025364  010600                                                                                  MOV     SP,R0
025366  104417                                                                                  TRAP    C$PNTF
025370  062706  000004                                                                          ADD     #4,SP

025374  005737  007216          GTRA3:  TST     RESFLG          ;SEE IF HERE AFTER A RESTART.
025400  001117                          BNE     GTRA5           ;BR IF HERE CAUSE OF A RESTART

                                        ; CLEAR COUNTS AND SET UP DEFAULTS

025402  005037  007166          GTRA4:  CLR     TOTCC           ;CLEAR TOTAL CHAR. COUNT TEMP. LOC.
025406  005037  007122                  CLR     TTOTCC          ; CLEAR TOTAL CHAR. COUNT FOR TX BUFF
025412  005037  007106                  CLR     CTOTCC          ; CLEAR TOTAL CHAR. COUNT FOR CMP BUFF
025416  012701  006326                  MOV     #PTRTAB,R1      ;INIT TRANSMIT MESSAGE POINTER
025422  010137  007100                  MOV     R1,TXPTR
025426  005037  007076                  CLR     RXPTR           ; ZERO RX POINTER
025432  012702  000017                  MOV     #MSGLIM,R2
025436  006302                          ASL     R2
025440  006302                          ASL     R2
025442  010137  007102                  MOV     R1,CMPPTR
025446  060237  007102                  ADD     R2,CMPPTR       ;INIT COMPARE MESSAGE POINTER

025452  012737  000005  007154          MOV     #5,MSGTYP       ;SET UP DEFAULT MSG TYPE (QUICK FOX - ITEP MSG)
025460  013737  002162  007156          MOV     MSG5C,CURCC     ;SET UP DEFAULT CHAR COUNT
025466  012737  003326  007124          MOV     #TXBUF,TCURAD   ;SET UP CURRENT ADD TO START OF TX BUFFER
025474  012737  005326  007110          MOV     #CMPBUF,CCURAD  ;SET UP CURRENT ADD TO START OF CMP BUFFER

025502  013737  007124  007164          MOV     TCURAD,CURADD   ;SETUP CURRENT ADDR TO START OF TXBUF
```

```
025510 013737 007100 007162          MOV     TXPTR,CPTR      ;SETUP CURRENT POINTER TABLE POINTER FOR TXBUF
025516 004737 022676                 JSR     PC,BLDBUF       ; GO BUILD POINTER TABLE AND BUFFER
025522 012737 000001 007120          MOV     #1,TXMTOT       ;BUMP TOTAL MESSAGE COUNT

025530 013737 007102 007162          MOV     CMPPTR,CPTR     ;SET UP START OF COMPARE POINTER TABLE
025536 013737 007110 007164          MOV     CCURAD,CURADD   ;SET UP CURRENT ADDR. TO START OF CMPBUF
025544 012737 000005 007154          MOV     #5,MSGTYP
025552 013737 002162 007156          MOV     MSG5C,CURCC
025560 004737 022676                 JSR     PC,BLDBUF       ;PUT DEFAULT MESSAGE INTO CMPBUF
025564 012737 000001 007104          MOV     #1,CMPTOT       ;BUMP THE COMP MESG COUNT
025572 012737 000003 007220          MOV     #ACT,MODTYP     ;SET DEFAULT MODE= ACTIVE
025600 005037 007222                 CLR     MLTYP           ;SET DEFAULT MAINTENANCE LOOP MODE =NONE
025604 012737 000001 007230          MOV     #1,RPASS        ;SET UP DEFAULT 'RUN PASS'' COUNT TO 1
025612 012737 000002 007226          MOV     #2,PARAM        ;SET UP PROG. PARAMETERS - DATACHECKING ENABLED
                                                             ;                 OPERATOR STATUS MSGS. PRINT OFF
025620                               PRINTF  #HLPO
025620 012746 012442                                                                 MOV     #HLPO,-(SP)
025624 012746 000001                                                                 MOV     #1,-(SP)
025630 010600                                                                        MOV     SP,R0
025632 104417                                                                        TRAP    C$PNTF
025634 062706 000004                                                                 ADD     #4,SP
025640 013737 007220 010334   GTRA5: MOV     MODTYP,DEV1
025646 013737 007222 010336          MOV     MLTYP,DEV2
025654 013737 007230 010340          MOV     RPASS,DEV3
025662 013737 007226 010342          MOV     PARAM,DEV4
025670 004737 023022                 JSR     PC,SHWOP        ;PRINT TO OPERATOR THE CURRENT MODE......

025674                               MANUAL                  ;SEE IF MANUAL INTERVENTION ALLOWED
025674 104450                                                                        TRAP    C$MANI
025676                               BCOMPLETE       GETCL   ;  BR IF YES (UAM=0 AND NOT CHAINED)
025676 103412                                                                        BCS     GETCL
025700 005737 007230                 TST     RPASS           ;SEE IF THIS IS FIRST 'DCLT PASS''
025704 001002                        BNE     1$              ; BR IF NOT COMPLETED 1 PASS
025706                               EXIT    TST             ; IF DONE 1 PASS IN UNATTENDED MODE - EXIT
025706 104432                                                                        TRAP    C$EXIT
025710 010050                                                                        .WORD   L10020-.
025712 012737 000001 007222   1$:    MOV     #TTL,MLTYP      ;SET UP DEFAULT FOR UNATTENDED MODE
025720 000137 030634                 JMP     GTR9            ;  'R M=ACT/LO=I/PAS=1/NOST/CH'' AND RUN

                              .SBTTL         COMMAND LINE FETCH & INTERPRETATION SECTION

025724 105037 003325          GETCL: CLRB    P$GDBD          ;CLEAR CMD LINE PARSING ERROR FLAGS
025730 105037 003324                 CLRB    P$NNUF
025734                               GMANID  CLI$PM,CMDBUF,A,0,1,72.,NO      ;GET A COMMAND LINE FROM OPR.
025734 104443                                                                        TRAP    C$GMAN
025736 000406                                                                        BR      10000$
025740 003062                                                                        .WORD   CMDBUF
025742 000142                                                                        .WORD   T$CODE
025744 012114                                                                        .WORD   CLI$PM
025746 000000                                                                        .WORD   0
025750 000001                                                                        .WORD   T$LOLIM
025752 000110                                                                        .WORD   T$HILIM
025754                                                                       10000$:
025754 012737 003062 003310          MOV     #CMDBUF,P$BUFA
025762 012737 010344 003312          MOV     #CLITRE,P$TREE
025770 012737 026664 003314          MOV     #CLIACT,P$ACT
```

```
026002  004737  023274                  JSR     PC,P$TRV                ;GO PARSE COMMAND LINE
025776  005037  003206                  CLR     QUALFG                  ;CLEAR QUALIFIER FLAG LOCATION
```

```
025776  005037  003206                  CLR     QUALFG                  ;CLEAR QUALIFIER FLAG LOCATION
026002  004737  023274                  JSR     PC,P$TRV                ;GO PARSE COMMAND LINE
026006  105737  003325                  TSTB    P$GDBD                  ;SEE IF PARSED OK OR AN ERROR
026012  001412                          BEQ     1$
026014                                  PRINTF  #CLIERM
026014  012746  012122                                                                          MOV     #CLIERM,-(SP)
026020  012746  000001                                                                          MOV     #1,-(SP)
026024  010600                                                                                  MOV     SP,R0
026026  104417                                                                                  TRAP    C$PNTF
026030  062706  000004                                                                          ADD     #4,SP
026034  000137  025724                  JMP     GETCL
026040  105737  003324          1$:     TSTB    P$NNUF                  ;SEE IF INCOMPLETE COMMAND TYPED
026044  001412                          BEQ     10$
026046                                  PRINTF  #CLINUF
026046  012746  012152                                                                          MOV     #CLINUF,-(SP)
026052  012746  000001                                                                          MOV     #1,-(SP)
026056  010600                                                                                  MOV     SP,R0
026060  104417                                                                                  TRAP    C$PNTF
026062  062706  000004                                                                          ADD     #4,SP
026066  000137  025724                  JMP     GETCL

026072  023727  003204  000005  10$:    CMP     KEYWD1,#HLP             ;SEE IF HELP WAS TYPED
026100  001711                          BEQ     GETCL                   ;GO GET CMD AGAIN IF YES
026102  023727  003204  000055          CMP     KEYWD1,#PRNT            ;SEE IF PRINT WAS TYPED
026110  001705                          BEQ     GETCL                   ; GO GET CMD AGAIN IF YES
026112  023727  003204  000004          CMP     KEYWD1,#RUN             ;SEE IF RUN WAS TYPED
026120  001002                          BNE     11$                     ; BR IF NO
026122  000137  030634                  JMP     GTR9                    ; START EXEC. IF YES
026126  023727  003204  000052  11$:    CMP     KEYWD1,#DMPS            ;SEE IF DUMP WAS TYPED
026134  001004                          BNE     12$                     ; BR IF NO
026136  004737  022442                  JSR     PC,DUMPSR               ; ELSE, DUMP PART OF MEMORY
026142  000137  025724                  JMP     GETCL                   ; THEN RETURN TO GET ANOTHER CMD.
026146  023727  003204  000001  12$:    CMP     KEYWD1,#CLEAR           ;SEE IF CLEAR WAS TYPED
026154  001663                          BEQ     GETCL                   ; IF YES, BACK TO GET ANOTHER CMD.
026156  023727  003204  000002          CMP     KEYWD1,#SHOW            ;SEE IF SHOW WAS TYPED
026164  001657                          BEQ     GETCL                   ; IF YES, BACK TO GET ANOTHER CMD.
026166  023727  003204  000010  4$:     CMP     KEYWD1,#SETEXP          ;SEE IF SET EXPECTED
026174  001512                          BEQ     2$                      ; BR IF YES (A SETEXP WAS TYPED)
026176  013737  007122  007166  5$:     MOV     TTOTCC,TOTCC
026204  023727  007166  001000          CMP     TOTCC,#BUFLIM           ;SEE IF BUFFER ALREADY FULL
026212  002414                          BLT     15$                     ; BR IF NOT FULL (BUFLIM # OF CHARS.)
026214                                  PRINTF  #MSGTRN,#BUFEX          ; ELSE TELL OPR. AND DON'T BUILD MSG.
026214  012746  014033                                                                          MOV     #BUFEX,-(SP)
026220  012746  014051                                                                          MOV     #MSGTRN,-(SP)
026224  012746  000002                                                                          MOV     #2,-(SP)
026230  010600                                                                                  MOV     SP,R0
026232  104417                                                                                  TRAP    C$PNTF
026234  062706  000006                                                                          ADD     #6,SP
026240  000137  025724                  JMP     GETCL                   ; THEN GO GET A NEW COMMAND
026244  005737  007122          15$:    TST     TTOTCC                  ;IF FIRST 'SET'' THEN GET RID OF DEFAULT
026250  001002                          BNE     6$
026252  005037  007120                  CLR     TXMTOT
026256  012737  006326  007100  6$:     MOV     #PTRTAB,TXPTR           ;GET POSITION OF END OF TX LIST
026264  013701  007120                  MOV     TXMTOT,R1
026270  020127  000017                  CMP     R1,#MSGLIM              ;SEE IF MSG COUNT EXCEEDED.
026274  002414                          BLT     17$                     ; BR IF NO
```

```
026276                                PRINTF  #MSGTRN,#TABEX         ; ELSE TELL OPR. AND DON'T BUILD MSG.
026276  012746  013773                                                                       MOV     #TABEX,-(SP)
026302  012746  014051                                                                       MOV     #MSGTRN,-(SP)
026306  012746  000002                                                                       MOV     #2,-(SP)
026312  010600                                                                               MOV     SP,R0
026314  104417                                                                               TRAP    C$PNTF
026316  062706  000006                                                                       ADD     #6,SP
026322  000137  025724                JMP     GETCL                  ; THEN GO GET A NEW COMMAND.
026326  006301                17$:    ASL     R1                     ;# OF MSGS *4 = NEXT FREE PTR BLOCK
026330  006301                        ASL     R1
026332  060137  007100                ADD     R1,TXPTR
026336  013737  007100  007162        MOV     TXPTR,CPTR             ;SETUP CHAR. COUNT, CURRENT ADDR, & PTR
026344  013737  007124  007164        MOV     TCURAD,CURADD
026352  004737  022600                JSR     PC,ADDCC               ;ADD IN CHAR. COUNT AND CHECK TOTAL
026356  004737  022676                JSR     PC,BLDBUF              ;GO BUILD MESSAGE IN BUFFER AND PTRS.
026362  013737  007162  007100        MOV     CPTR,TXPTR
026370  013737  007166  007122        MOV     TOTCC,TTOTCC           ;UPDATE CHAR. COUNT, CURR ADDR, & PTR
026376  013737  007164  007124        MOV     CURADD,TCURAD
026404  005237  007120                INC     TXMTOT
026410  005337  003210                DEC     QUALVL                 ;DEC THE COPY COUNT
026414  001270                        BNE     5$
026416  000137  025724                JMP     GETCL

026422  013737  007106  007166  2$:   MOV     CTOTCC,TOTCC           ;SETUP CHAR. COUNT, CURR. ADDR. & PTR
026430  023727  007166  001000        CMP     TOTCC,#BUFLIM          ;SEE IF BUFFER ALREADY FULL
026436  002414                        BLT     16$                    ; BR IF NOT FULL (BUFLIM # OF CHARS.)
026440                                PRINTF  #MSGTRN,#BUFEX         ; ELSE TELL OPR. AND DON'T BUILD MSG.
026440  012746  014033                                                                       MOV     #BUFEX,-(SP)
026444  012746  014051                                                                       MOV     #MSGTRN,-(SP)
026450  012746  000002                                                                       MOV     #2,-(SP)
026454  010600                                                                               MOV     SP,R0
026456  104417                                                                               TRAP    C$PNTF
026460  062706  000006                                                                       ADD     #6,SP
026464  000137  025724                JMP     GETCL                  ; THEN GO GET A NEW COMMAND
026470  005737  007106        16$:    TST     CTOTCC                 ;IF FIRST ''SET'' THEN GET RID OF DEFAULT
026474  001002                        BNE     7$
026476  005037  007104                CLR     CMPTOT
026502  012701  006326        7$:     MOV     #PTRTAB,R1
026506  012702  000017                MOV     #MSGLIM,R2
026512  006302                        ASL     R2
026514  006302                        ASL     R2
026516  010137  007102                MOV     R1,CMPPTR
026522  060237  007102                ADD     R2,CMPPTR              ;INIT COMPARE MESSAGE POINTER
026526  013701  007104                MOV     CMPTOT,R1
026532  020127  000017                CMP     R1,#MSGLIM             ;SEE IF MSG COUNT EXCEEDED.
026536  002414                        BLT     18$                    ; BR IF NO
026540                                PRINTF  #MSGTRN,#TABEX         ; ELSE TELL OPR. AND DON'T BUILD MSG.
026540  012746  013773                                                                       MOV     #TABEX,-(SP)
026544  012746  014051                                                                       MOV     #MSGTRN,-(SP)
026550  012746  000002                                                                       MOV     #2,-(SP)
026554  010600                                                                               MOV     SP,R0
026556  104417                                                                               TRAP    C$PNTF
026560  062706  000006                                                                       ADD     #6,SP
026564  000137  025724                JMP     GETCL                  ; THEN GO GET A NEW COMMAND.
026570  006301                18$:    ASL     R1                     ;# OF MSGS *4 = NEXT FREE PTR BLOCK
026572  006301                        ASL     R1
```

```
 4641  026574  060137  007102                    ADD     R1,CMPPTR
 4642  026600  013737  007102  007162            MOV     CMPPTR,CPTR
 4643  026606  013737  007110  007164            MOV     CCURAD,CURADD
 4644  026614  004737  022600                    JSR     PC,ADDCC                  ;ADD IN XHAR. COUNT AND CHECK TOTAL
 4645  026620  004737  022676                    JSR     PC,BLDBUF
 4646  026624  013737  007162  007102            MOV     CPTR,CMPPTR
 4647  026632  005237  007104                    INC     CMPTOT
 4648  026636  013737  007164  007110            MOV     CURADD,CCURAD             ;UPDATE CHAR. COUNT, CURR ADDRR. & PTR
 4649  026644  013737  007166  007106            MOV     TOTCC,CTOTCC
 4650  026652  005337  003210                    DEC     QUALVL                    ;IF COPY WAS GIVEN, PUT MSG IN BUFF
 4651  026656  001261                            BNE     2$                        ; AGAIN
 4652  026660  000137  025724                    JMP     GETCL                     ;GO BACK UNTIL GET A 'RUN''
 4653
 4654
 4655
 4656
 4657
```

```

                                   .SBTTL          ACTION TABLE AND ROUTINES
                                   ;       USER MUST CLEAR/SET P$GDBD IF USE ''CLIBIF'' IN CONNECTION WITH ACTION
                                   ;       R2 WILL HOLD ACTION CODE FROM PARSING (CLI) NODE
 026664                            CLIACT:
 026664  006302                            ASL     R2
 026666  016202  026702                    MOV     10$(R2),R2          ;FORM ADDRESS OF ACTION ROUTINE
 026672  062702  026702                    ADD     #10$,R2
 026676  004712                            JSR     PC,(R2)
 026700  000207                            RTS     PC

 026702  000142                    10$:    .WORD   ACTNUL-10$
 026704  000144                            .WORD   ACTCLR-10$
 026706  000154                            .WORD   ACTSHO-10$
 026710  001530                            .WORD   ACTCHK-10$
 026712  000244                            .WORD   ACTRUN-10$
 026714  000164                            .WORD   ACTHLP-10$
 026716  000270                            .WORD   ACTCSE-10$
 026720  000416                            .WORD   ACTCST-10$
 026722  000740                            .WORD   ACTSTE-10$
 026724  000750                            .WORD   ACTSTT-10$
 026726  000766                            .WORD   ACTSZE-10$
 026730  000776                            .WORD   ACTCOP-10$
 026732  001006                            .WORD   ACTNUM-10$
 026734  001100                            .WORD   ACTOPM-10$
 026736  001536                            .WORD   ACTSTS-10$
 026740  001120                            .WORD   ACTEQO-10$
 026742  001200                            .WORD   ACTMS0-10$
 026744  001206                            .WORD   ACTMS1-10$
 026746  001216                            .WORD   ACTMS2-10$
 026750  001226                            .WORD   ACTMS3-10$
 026752  001236                            .WORD   ACTMS4-10$
 026754  001246                            .WORD   ACTMS5-10$
 026756  001264                            .WORD   ACTMS6-10$
 026760  001314                            .WORD   ACTATV-10$
 026762  001324                            .WORD   ACTPAS-10$
 026764  001344                            .WORD   ACTREC-10$
 026766  001352                            .WORD   ACTLIS-10$
 026770  001362                            .WORD   ACTDLL-10$
 026772  001372                            .WORD   ACTTRA-10$
 026774  001402                            .WORD   ACTTAL-10$
 026776  001430                            .WORD   ACTNO-10$
 027000  001440                            .WORD   ACTECH-10$
 027002  001544                            .WORD   ACTCRC-10$
 027004  001552                            .WORD   ACTPRO-10$
 027006  001604                            .WORD   ACTRPS-10$
 027010  001614                            .WORD   ACTMOP-10$
 027012  001624                            .WORD   ACTTLP-10$
 027014  001634                            .WORD   ACTCLP-10$
 027016  001644                            .WORD   ACTLLP-10$
 027020  001654                            .WORD   ACTRLP-10$
 027022  000134                            .WORD   ACTNUF-10$
 027024  001156                            .WORD   ACTBCR-10$
 027026  000674                            .WORD   ACTDMS-10$
 027030  000724                            .WORD   ACTDME-10$
 027032  000716                            .WORD   ACTDMQ-10$
```

```
4714  027034  000230                              .WORD    ACTPRT-10$
4715
4716
```

```
 4717
 4718  027036  112737  177777  003324  ACTNUF: MOVB    #-1,P$NNUF          ;SET FLAG TO SAY NEED MORE OF COMMAND
 4719  027044  000207                  ACTNUL: RTS     PC                  ;RETURN TO PARSER
 4720
 4721  027046  012737  000001  003204  ACTCLR: MOV     #CLEAR,KEYWD1       ;SET LOC TO SAY A CLEAR WAS TYPED
 4722  027054  000207                          RTS     PC
 4723
 4724  027056  012737  000002  003204  ACTSHO: MOV     #SHOW,KEYWD1        ;SET LOC. TO SAY A SHOW WAS TYPED
 4725  027064  000207                          RTS     PC
 4726
 4727  027066  012702  003212          ACTHLP: MOV     #HLPTAB,R2          ;SETUP R2 AS A POINTER TO HELP MSG TABLE
 4728  027072                          1$:     PRINTF  #HLPF,(R2)+         ;PRINT HELP INFORMATION MESSAGES
 4729  027072  012246                                                                          MOV     (R2)+,-(SP)
 4730  027074  012746  012520                                                                  MOV     #HLPF,-(SP)
 4731  027100  012746  000002                                                                  MOV     #2,-(SP)
 4732  027104  010600                                                                          MOV     SP,R0
 4733  027106  104417                                                                          TRAP    C$PNTF
 4734  027110  062706  000006                                                                  ADD     #6,SP
 4735  027114  020227  003230                  CMP     R2,#HLPEND          ;SEE IF ALL INFO PRINTED YET
 4736  027120  001364                          BNE     1$                  ;IF NO KEEP PRINTING
 4737  027122  012737  000005  003204          MOV     #HLP,KEYWD1         ;SET LOC. TO SAY A HELP WAS TYPED
 4738  027130  000207                          RTS     PC
 4739
 4740  027132  012737  000055  003204  ACTPRT: MOV     #PRNT,KEYWD1        ;SET LOC. TO SAY A HELP WAS TYPED
 4741  027140  004737  021406                  JSR     PC,REPORT           ;CALL ROUTINE TO PRINT EVENT LOG AND BASE TABLE
 4742  027144  000207                          RTS     PC
 4743
 4744  027146  012737  000004  003204  ACTRUN: MOV     #RUN,KEYWD1         ;SET RUN FLAG
 4745  027154  112737  177777  003324          MOVB    #-1,P$NNUF          ;SET FLAG TO SAY NEED MORE OF COMMAND
 4746  027162  012737  000001  007230          MOV     #1,RPASS            ;SET DEFAULT RUN 'PASS'' TO 1
 4747  027170  000207                          RTS     PC
 4748
 4749  027172  012701  006326          ACTCSE: MOV     #PTRTAB,R1
 4750  027176  012702  000017                  MOV     #MSGLIM,R2
 4751  027202  006302                          ASL     R2
 4752  027204  006302                          ASL     R2
 4753  027206  010137  007102                  MOV     R1,CMPPTR
 4754  027212  060237  007102                  ADD     R2,CMPPTR           ;INIT COMPARE MESSAGE POINTER
 4755  027216  013701  007102                  MOV     CMPPTR,R1
 4756
 4757  027222  013702  007104                  MOV     CMPTOT,R2
 4758  027226  105037  003324                  CLRB    P$NNUF              ;FLAG THAT HAVE VALID COMMAND AT THIS PT.
 4759  027232  023727  003204  000002          CMP     KEYWD1,#SHOW        ;SEE IF A CLEAR OR SHOW WAS TYPED
 4760  027240  001500                          BEQ     ACTSHW              ;BR IF A SHOW WAS TYPED
 4761  027242  012737  000001  007104          MOV     #1,CMPTOT           ;CLEAR COMPARE MESSAGE COUNT, CHAR. COUNT
 4762  027250  005037  007106                  CLR     CTOTCC              ; AND RESET POINTER
 4763
 4764  027254  012701  006326                  MOV     #PTRTAB,R1
 4765  027260  012702  000017                  MOV     #MSGLIM,R2
 4766  027264  006302                          ASL     R2
 4767  027266  006302                          ASL     R2
 4768  027270  010137  007102                  MOV     R1,CMPPTR
 4769  027274  060237  007102                  ADD     R2,CMPPTR           ;INIT COMPARE MESSAGE POINTER
 4770  027300  013737  007102  007162          MOV     CMPPTR,CPTR         ;SET UP TO FILL IN DEFAULT MESSAGE
 4771  027306  012701  005326                  MOV     #CMPBUF,R1
 4772  027312  010137  007110                  MOV     R1,CCURAD
```

```
027316 000431                          BR      ACTCLB

027320 012701 006326          ACTCST:  MOV     #PTRTAB,R1
027324 013702 007120                   MOV     TXMTOT,R2
027330 105037 003324                   CLRB    P$NNUF             ;FLAG THAT HAVE VALID COMMAND AT THIS PT.
027334 023727 003204 000002            CMP     KEYWD1,#SHOW       ;SEE IF A CLEAR OR SHOW WAS TYPED
027342 001437                          BEQ     ACTSHW             ;BR IF A SHOW WAS TYPED
027344 012737 000001 007120            MOV     #1,TXMTOT          ;CLEAR TRANSMIT MESSAGE COUNT, CHAR. COUNT
027352 005037 007122                   CLR     TTOTCC             ; AND RESET POINTER
027356 012737 006326 007100            MOV     #PTRTAB,TXPTR
027364 013737 007100 007162            MOV     TXPTR,CPTR
027372 012701 003326                   MOV     #TXBUF,R1
027376 010137 007124                   MOV     R1,TCURAD

027402 012702 001000          ACTCLB:  MOV     #BUFLIM,R2
027406 010137 007164                   MOV     R1,CURADD          ;SET UP TO PUT DEFAULT MSG IN LIST AFTER 033'S
027412 012737 000005 007154            MOV     #5,MSGTYP
027420 013737 002162 007156            MOV     MSG5C,CURCC
027426 105021                 1$:      CLRB    (R1)+              ;FILL EXPT OR TRAN BUFFER WITH 0'S IF A CLEAR
027430 005302                          DEC     R2                 ;DO 'BUFLIM' NUMBER OF BYTE LOCATIONS
027432 001375                          BNE     1$
027434 004737 022676                   JSR     PC,BLDBUF          ;''CLEAR'' REALLY MEANS TO PUT DEFAULT MSG IN
027440 000207                          RTS     PC                 ;WHEN DONE, RETURN TO PARSER


027442 012705 003250          ACTSHW:  MOV     #SHTAB,R5
027446 122571 000000          5$:      CMPB    (R5)+,@(R1)        ;LOOK AT FIRST BYTE OF MSG TO DECIPHER TYPE
027452 001404                          BEQ     6$
027454 020527 003257                   CMP     R5,#SHTEND         ;SEE IF LOOKED AT ALL OF DEFAULTS YET
027460 001372                          BNE     5$
027462 005205                          INC     R5                 ;MUST BE OPR. SPEC'D THEN
027464 162705 003251          6$:      SUB     #SHTAB+1,R5
027470 006305                          ASL     R5
027472 016137 000002 007172            MOV     2(R1),TEMP
027500                                 PRINTF  #SHMSG,SHTYTB(R5),TEMP  ;PRINT MSG SIZE & TYPE
027500 013746 007172                                                                MOV     TEMP,-(SP)
027504 016546 003230                                                                MOV     SHTYTB(R5),-(SP)
027510 012746 013251                                                                MOV     #SHMSG,-(SP)
027514 012746 000003                                                                MOV     #3,-(SP)
027520 010600                                                                       MOV     SP,R0
027522 104417                                                                       TRAP    C$PNTF
027524 062706 000010                                                                ADD     #10,SP
027530 062701 000004                   ADD     #4,R1              ;BUMP R1 TO NEXT SET OF POINTERS
027534 005302                          DEC     R2
027536 001341                          BNE     ACTSHW
027540 013737 007220 010334            MOV     MODTYP,DEV1
027546 013737 007222 010336            MOV     MLTYP,DEV2
027554 013737 007230 010340            MOV     RPASS,DEV3
027562 013737 007226 010342            MOV     PARAM,DEV4
027570 004737 023022                   JSR     PC,SHWOP           ;SHOW THE OPERATOR THE CURRENT MODE..... ALSO
027574 000207                          RTS     PC

027576 013737 003320 007146   ACTDMS:  MOV     P$NUM,STADD        ;SETUP STARTING ADDRESS FOR DUMP
027604 005037 007152                   CLR     BYTBIT             ;SET DEFAULT OF WORD DUMP
027610 012737 000052 003204            MOV     #DMPS,KEYWD1       ;FLAG THAT A DUMP WAS TYPED
027616 000403                          BR      ACTDME
```

```

027620  012737  177777  007152  ACTDMQ: MOV     #-1,BYTBIT          ;SET DUMP FLAG TO 'DUMP-WORD''
027626  013737  003320  007150  ACTDME: MOV     P$NUM,ENADD         ;SETUP END ADDRESS FOR DUMP (=START IF NO 'EEE''
027634  105037  003324          ACTDMX: CLRB    P$NNUF              ;CLEAR NOT-ENOUGH FLAG, 'DUMP N-N/B'' IS VALID
027640  000207                          RTS     PC

```

```


027642  012737  000010  003204  ACTSTE: MOV     #SETEXP,KEYWD1
027650  000403                          BR      ACTSTX

027652  012737  000011  003204  ACTSTT: MOV     #SETTRN,KEYWD1
027660  012737  000001  003210  ACTSTX: MOV     #1,QUALVL               ;SET UP DEFAULT COPY TO 1 (/COPY=0)
027666  000207                          RTS     PC

027670  012737  000012  003206  ACTSZE: MOV     #SIZE,QUALFG
027676  000207                          RTS     PC

027700  012737  000013  003206  ACTCOP: MOV     #QCOPY,QUALFG
027706  000207                          RTS     PC

027710  023727  003206  000012  ACTNUM: CMP     QUALFG,#SIZE            ;SEE IF A SIZE OR COPY TYPED
027716  001023                          BNE     1$                      ;BR IF IT WAS A COPY
027720  005737  003320                  TST     P$NUM                   ;CHECK TO BE SURE DIDN'T TRY SIZE=0
027724  001014                          BNE     3$                      ; BR IF NO
027726                                  PRINTF  #CLISE0
027726  012746  012411                                                                                  MOV     #CLISE0,-(SP)
027732  012746  000001                                                                                  MOV     #1,-(SP)
027736  010600                                                                                          MOV     SP,R0
027740  104417                                                                                          TRAP    C$PNTF
027742  062706  000004                                                                                  ADD     #4,SP
027746  112737  177777  003325          MOVB    #-1,P$GDBD              ;SEE ERROR-IN-CMD FLAG
027754  000411                          BR      2$
027756  013737  003320  007156  3$:     MOV     P$NUM,CURCC             ;IF A SIZE LOAD CURCC WITH BYTE COUNT
027764  000405                          BR      2$
027766  013737  003320  003210  1$:     MOV     P$NUM,QUALVL            ;IF A COPY, LOAD COPY COUNT
027774  005237  003210                  INC     QUALVL                  ;INCREMENT SO FIRST DEC MAKES IT REAL #
030000  000503                  2$:     BR      ACTMEX

030002  012737  000007  007154  ACTOPM: MOV     #7,MSGTYP
030010  010437  007172                  MOV     R4,TEMP                 ;KEEP TRACK OF START OF QUOTED TEXT
030014  005237  007172                  INC     TEMP                    ; SO CAN CALC OPCNT AT END OF QUOTES
030020  000207                          RTS     PC

030022  010402                  ACTEQO: MOV     R4,R2
030024  163702  007172                  SUB     TEMP,R2
030030  010237  007156                  MOV     R2,CURCC                ;CALC BYTE COUNT FOR QUOTED TEXT
030034  010237  002166                  MOV     R2,OPCNT
030040  013701  007172                  MOV     TEMP,R1
030044  012705  002524                  MOV     #OPBUF,R5
030050  112125                  1$:     MOVB    (R1)+,(R5)+             ;COPY QUOTED TEXT TO OPBUF
030052  005302                          DEC     R2
030054  001375                          BNE     1$
030056  000454                          BR      ACTMEX

030060                          ACTBCR: PRINTF  #CLIBCR                 ;BAD CHAR. IN OPR. QUOTED STRING
030060  012746  012344                                                                                  MOV     #CLIBCR,-(SP)
030064  012746  000001                                                                                  MOV     #1,-(SP)
030070  010600                                                                                          MOV     SP,R0
030072  104417                                                                                          TRAP    C$PNTF
030074  062706  000004                                                                                  ADD     #4,SP
030100  000207                          RTS     PC
```

```

030102  005037  007154          ACTMS0: CLR     MSGTYP
030106  000435                          BR      ACTME1
030110  012737  000001  007154  ACTMS1: MOV     #1,MSGTYP
030116  000431                          BR      ACTME1
030120  012737  000002  007154  ACTMS2: MOV     #2,MSGTYP
030126  000425                          BR      ACTME1
030130  012737  000003  007154  ACTMS3: MOV     #3,MSGTYP
030136  000421                          BR      ACTME1
030140  012737  000004  007154  ACTMS4: MOV     #4,MSGTYP
030146  000415                          BR      ACTME1
030150  012737  000005  007154  ACTMS5: MOV     #5,MSGTYP
030156  013737  002162  007156          MOV     MSG5C,CURCC             ;SETUP DEFAULT SIZE FOR THIS TYPE
030164  000411                          BR      ACTMEX
030166  012737  000006  007154  ACTMS6: MOV     #6,MSGTYP
030174  013737  002164  007156          MOV     MSG6C,CURCC             ;SETUP DEFAULT SIZE FOR THIS TYPE

030202  012737  000100  007156  ACTME1: MOV     #64.,CURCC              ;SETUP DEFAULT SIZE FOR MSG0-4
030210  105037  003324          ACTMEX: CLRB    P$NNUF                  ;CLEAR NOT-ENOUGH FLAG
030214  000207                          RTS     PC

```

```
030216  012737  000003  007220  ACTATV: MOV     #ACT,MODTYP
030224  000432                          BR      ACTM2X

030226  012737  000002  007220  ACTPAS: MOV     #PAS,MODTYP
030234  105037  003324                  CLRB    P$NNUF                  ;CLEAR NOT-ENOUGH FLAG
030240  005037  007222                  CLR     MLTYP                   ;CLEAR MAINT LOOP TYPE
030244  000207                          RTS     PC

030246  005037  007220          ACTREC: CLR     MODTYP
030252  000417                          BR      ACTM2X

030254  012737  000006  007220  ACTLIS: MOV     #LIS,MODTYP
030262  000413                          BR      ACTM2X

030264  012737  000004  007220  ACTDLL: MOV     #DOW,MODTYP
030272  000407                          BR      ACTM2X

030274  012737  000001  007220  ACTTRA: MOV     #TRA,MODTYP
030302  000403                          BR      ACTM2X

030304  012737  000005  007220  ACTTAL: MOV     #TAL,MODTYP

030312  042737  000004  007226  ACTM2X: BIC     #ECHOB,PARAM            ;DISABLE /ECHO (ALL BUT PASSIVE MODE)
030320  105037  003324                  CLRB    P$NNUF                  ;CLEAR NOT-ENOUGH FLAG
030324  005037  007222                  CLR     MLTYP                   ;CLEAR MAINT LOOP TYPE
030330  000207                          RTS     PC

```

```
030332  012737  000036  003206  ACTNO:  MOV     #NO,QUALFG
030340  000207                          RTS     PC

030342  022737  000036  003206  ACTECH: CMP     #NO,QUALFG
030350  001422                          BEQ     1$
030352  052737  000004  007226          BIS     #ECHOB,PARAM
030360  022737  000002  007220          CMP     #PAS,MODTYP                     ;BE SURE IN PASSIVE MODE IF
030366  001416                          BEQ     2$                              ;IF TRYING TO SET /ECHO
030370                                  PRINTF  #CLINPS
030370  012746  012301                                                                          MOV     #CLINPS,-(SP)
030374  012746  000001                                                                          MOV     #1,-(SP)
030400  010600                                                                                  MOV     SP,R0
030402  104417                                                                                  TRAP    C$PNTF
030404  062706  000004                                                                          ADD     #4,SP
030410  112737  177777  003325          MOVB    #-1,P$GDBD
030416  042737  000004  007226  1$:     BIC     #ECHOB,PARAM
030424  005037  003206          2$:     CLR     QUALFG                          ;CLEAR 'NO'' OUT OF QUALIFIER FLAG
030430  000476                          BR      ACTLXX

030432  012701  000002          ACTCHK: MOV     #DATCKB,R1                      ;SET DATA CHECK BIT
030436  000410                          BR      ACTQFG

030440  012701  000001          ACTSTS: MOV     #STATB,R1               ;SET THE STATUS BIT
030444  000405                          BR      ACTQFG

030446  012701  000020          ACTCRC: MOV     #CRCB,R1                ;SET THE CRC BIT
030452  000402                          BR      ACTQFG

030454  012701  000040          ACTPRO: MOV     #PROTOB,R1              ;SET THE PROTOCOL BIT

030460  050137  007226          ACTQFG: BIS     R1,PARAM
030464  022737  000036  003206          CMP     #NO,QUALFG
030472  001002                          BNE     1$
030474  040137  007226                  BIC     R1,PARAM
030500  005037  003206          1$:     CLR     QUALFG                          ;CLEAR 'NO'' OUT OF QUALIFIER FLAG
030504  000450                          BR      ACTLXX

030506  013737  003320  007230  ACTRPS: MOV     P$NUM,RPASS                     ;GET NUMBER OF 'RUN PASSES''
030514  000444                          BR      ACTLXX

030516  012737  000005  007222  ACTMOP: MOV     #5,MLTYP
030524  000417                          BR      ACTLPX
030526  012737  000001  007222  ACTTLP: MOV     #1,MLTYP
030534  000413                          BR      ACTLPX
030536  012737  000002  007222  ACTCLP: MOV     #2,MLTYP
030544  000407                          BR      ACTLPX
030546  012737  000003  007222  ACTLLP: MOV     #3,MLTYP
030554  000403                          BR      ACTLPX
030556  012737  000004  007222  ACTRLP: MOV     #4,MLTYP

030564  022737  000003  007220  ACTLPX: CMP     #ACT,MODTYP             ;BE SURE IN ACTIVE IF TRYING TO SET LOOP
030572  001415                          BEQ     ACTLXX                  ; BR IF IN ACTIVE
030574  112737  177777  003325          MOVB    #-1,P$GDBD
030602  005037  007222                  CLR     MLTYP                   ;CLEAR ANY LOOP TYPE THAT MAY HAVE GOT SET
030606                                  PRINTF  #CLIBDL
030606  012746  012237                                                                          MOV     #CLIBDL,-(SP)
```

```
4995 030612 012746 000001                                                         MOV     #1,-(SP)
4996 030616 010600                                                                MOV     SP,R0
4997 030620 104417                                                                TRAP    C$PNTF
4998 030622 062706 000004                                                         ADD     #4,SP
4999 030626 105037 003324          ACTLXX: CLRB    P$NNUF          ;CLEAR NOT-ENOUGH FLAG
5000 030632 000207                         RTS     PC
5001
```

```

                                                ; RX ALLOCATE CODE
030634  012701  006326          GTR9:   MOV     #PTRTAB,R1      ;INIT TRANSMIT MESSAGE POINTER
030640  010137  007100                  MOV     R1,TXPTR
030644  012702  000017                  MOV     #MSGLIM,R2
030650  006302                          ASL     R2
030652  006302                          ASL     R2
030654  010137  007102                  MOV     R1,CMPPTR
030660  060237  007102                  ADD     R2,CMPPTR       ;INIT COMPARE MESSAGE POINTER
030664  013701  007102                  MOV     CMPPTR,R1
030670  012702  000017                  MOV     #MSGLIM,R2
030674  006302                          ASL     R2
030676  006302                          ASL     R2
030700  010137  007076                  MOV     R1,RXPTR
030704  060237  007076                  ADD     R2,RXPTR        ;INIT RECEIVE MESSAGE POINTER

030710  013737  007104  007134          MOV     CMPTOT,RXMTOT   ;MAKE COMPARE AND RX MESSAGE COUNTS EQUAL


030716  005037  007232          GTREX:  CLR     FLAG            ;CLEAR FLAG
030722  005037  007140                  CLR     NOBUF           ;CLEAR NO BUFFER COUNTER
030726  005037  007142                  CLR     PSCNT           ;CLEAR PASS COUNT
030732  005037  007144                  CLR     ERRCNT          ;CLEAR ERROR COUNT

030736  004737  021012                  JSR     PC,LOGDVI       ;LOG ABOUT TO INIT DEVICE
030742  004737  033400                  JSR     PC,DVINIT       ;INIT DEVICE

030746  012737  001000  007156  GTRX2:  MOV     #BUFLIM,CURCC   ;SET CHAR COUNT TO 'BUFLIM' NO. OF BYTES
030754  012737  004326  007164          MOV     #RXBUF,CURADD   ;SET UP RX BUFFER AS CURRRENT ADD.
030762  013737  007076  007162          MOV     RXPTR,CPTR
030770  012737  000010  007154          MOV     #10,MSGTYP      ;SET UP FOR 33 TO FILL RX BUFFERS
030776  004737  022676                  JSR     PC,BLDBUF       ;CLEAR RX BUFFER
031002  013702  007220                  MOV     MODTYP,R2
031006  006302                          ASL     R2
031010  000172  007234                  JMP     @MODE(R2)       ;MODE DISPATCH

```

```
                                    .SBTTL          RECEIVE MODE SECTION
                                    ;++
                                    ; FUNCTIONAL DESCRIPTION:
                                    ;       RECEIVE-ONLY (OR ONE-WAY-IN) ROUTINE
                                    ;       IN THIS MODE OF TESTING THE DEVICE'S RECEIVER IS ENABLED IN EXPECTATION
                                    ;       OF RECEIVING A MESSAGE.  AFTER RECEIVING AN ''EXPECTED'' NUMBER OF
                                    ;       MESSAGES, THE DATA RECEIVED CAN BE COMPARED AGAINST A LIST OF ''EXPECT
                                    ;       TO RECEIVE'' MESSAGES IF DATA-CHECKING IS ENABLED.
                                    ;
                                    ; SUBORDINATE ROUTINES USED:
                                    ;               ''ALLTR''
                                    ;
                                    ; CALLING SEQUENCE:
                                    ;       JMP     @MODE(R2)       ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
                                    ;--

031014                              RXONLY:
031014  013737  007076  007160      RXON2:  MOV     RXPTR,CPTRR
031022  013737  007134  007132              MOV     RXMTOT,DVRCT    ;SET UP MESSAGE COUNT
031030  052737  000104  007232              BIS     #QRX+#ERX,FLAG  ;SET UP RX QUE
031036  005037  007162                      CLR     CPTR            ;CLEAR THE TX POINTER
031042  000137  031200                      JMP     ALLTR           ;GO RX.
```

```
                                        .SBTTL          TRANSMIT MODE SECTION

                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       TRANSMIT-ONLY (OR ONE-WAY-OUT) ROUTINE
                                        ;       IN THIS MODE OF TESTING A LIST OF MESSAGES IS TRANSMITTED WITHOUT
                                        ;       EXPECTING ANY DATA TO BE RECEIVED.  A REPETITION COUNT CAN BE
                                        ;       SPECIFIED TO REPETITIVELY TRANSMIT THE LIST.
                                        ;
                                        ; SUBORDINATE ROUTINES USED:
                                        ;               ''ALLTR''
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;       JMP     @MODE(R2)       ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
                                        ;--

031046  042737  000002  007226  TXONLY: BIC     #DATCKB,PARAM   ;SET NOCHECK
031054  013737  007100  007162  TXON2:  MOV     TXPTR,CPTR
031062  013737  007120  007116          MOV     TXMTOT,DVTCT    ;COPY COUNTER FOR THIS PASS
031070  052737  000210  007232          BIS     #QTX+#ETX,FLAG  ;SET THE QUE TX FLAG
031076  005037  007160                  CLR     CPTRR           ;CLEAR RX POINTER
031102  000137  031200                  JMP     ALLTR           ;GO TX.
```

```
                                .SBTTL          PASSIVE MODE SECTION

                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;       PASSIVE MODE SECTION
                                ;       IN THIS MODE OF TESTING, THE DEVICE'S RECEIVER IS ENABLED IN
                                ;       EXPECTATION OF RECEIVING A MESSAGE.  THEN EVERY TIME A MESSAGE IS
                                ;       RECEIVED, A MESSAGE IS TRANSMITTED.  DATA CHECKING CAN BE DONE ON THE
                                ;       RECEIVED DATA.
                                ;
                                ; SUBORDINATE ROUTINES USED:
                                ;
                                ;               ''ALLTR''
                                ;
                                ; CALLING SEQUENCE:
                                ;       JMP     @MODE(R2)       ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
                                ;--

031106                          PLCK:
031106  013737  007120  007116  PLCK2:  MOV     TXMTOT,DVTCT    ;SET UP  THE TRANSMIT COUNT
031114  013737  007100  007162          MOV     TXPTR,CPTR      ;SET UP CPTR TO TRANSMIT POINTER
031122  013737  007076  007160  PLCK3:  MOV     RXPTR,CPTRR     ;SET UP CPTRR TO REC POINTER
031130  052737  000104  007232          BIS     #QRX+#ERX,FLAG  ;SET UP Q AND EXPECT RX
031136  000137  031200                  JMP     ALLTR           ;AND GO RX FIRST MSG.
```

```
                                         .SBTTL          ACTIVE MODE SECTION

                                         ;++
                                         ; FUNCTIONAL DESCRIPTION:
                                         ;        ACTIVE MODE SECTION
                                         ;        IN THIS MODE OF TESTING A LIST OF MESSAGES IS TRANSMITTED AND
                                         ;        MESSAGES ARE EXPECTED TO BE RECEIVED.  RECEIVED DATA CAN BE COMPARED
                                         ;        AGAINST ''EXPECTED'' DATA IF DATA-CHECKING IS ENABLED.
                                         ;                NOTE: IF BOTH ENDS OF THE LINK ARE IN ACTIVE MODE, THEN THE
                                         ;                        LINK MUST BE A FULL DUPLEX LINK!
                                         ;
                                         ; SUBORDINATE ROUTINES USED:
                                         ;
                                         ;                ''ALLTR''
                                         ;
                                         ; CALLING SEQUENCE:
                                         ;        JMP     @MODE(R2)       ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
                                         ;--

 031142  013737  007120  007116  ALCK:   MOV     TXMTOT,DVTCT
 031150  013737  007100  007162          MOV     TXPTR,CPTR      ;SET UP TX COUNTS
 031156  013737  007134  007132          MOV     RXMTOT,DVRCT    ;SET UP COUNTS
 031164  013737  007076  007160          MOV     RXPTR,CPTRR
 031172  052737  000314  007232          BIS     #QRX+#QTX+#ETX+#ERX,FLAG
```

```
                                .SBTTL          TRANSMIT - RECEIVE FOR ALL STANDARD MODES

                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;       THIS CODE PERFORMS THE FOLLOWING FUNCTIONS
                                ;       1.) IF RX BUFFERS ARE TO BE QUED,TELL DEVICE
                                ;           CODE TO QUE THEM ,LOG RECEIVE QUED.
                                ;       2.) IF TX BUFFERS ARE TO BE QUED ,TELL DEVICE
                                ;           CODE TO QUE THEM, LOG TRANSMIT QUED.
                                ;       3.) WAIT FOR EITHER RECIVE BUFFER OR TRANSMIT BUFFER OR
                                ;           BOTH TO COMPLETE
                                ;       4.) IF RECEIVE COMPLETE LOG IT UPDATE RX TABLE IF DATA
                                ;           CHECKING.
                                ;       5.) IF TRANSMIT COMPLETE LOG IT.
                                ;       6.) WHEN BOTH TRANSMIT AND RECIEVE LISTS ARE DONE
                                ;           GO TO THE COMPARE BUFFER CODE
                                ;
                                ; SUBORDINATE ROUTINES USED:
                                ;               'DVRXQ'' -QUE RECEIVE BUFFER SPACE TO DEVICE
                                ;               'LOGRXQ''-LOG RECEIVE BUFFER SPACE TO EVENT LOG
                                ;               'LOGTXQ''-LOG TRANSMIT BUFFER QUED TO EVENT LOG
                                ;               'DVTXRX''-QUE TRANSMIT BUFFER AND WAIT FOR RX
                                ;                         OR TX TO COMPLETE
                                ;               'LOGRXC''-LOG RECEIVE BUFFER COMPLETED TO EVENT LOG
                                ;               'LOGTXC''-LOG TRANSMIT BUFFER COMPLETED TO EVENT LOG
                                ;
                                ; USE OF FLAG BITS:
                                ;               QRX - SET ON INPUT TO ALLTR IF REC IS TO BE QUED TO
                                ;                     DEVICE. CLEARED BY DVRXQ AND THEN SET BY DVTXRX
                                ;                     WHEN RX BUFFER IS COMPLETED.
                                ;               QTX - SET ON INPUT TO ALLTR IF TRANSMIT IS TO BE QUED TO
                                ;                     DEVICE. CLEARED ON ENTRY TO DVTXRX AND SET BY DVTXRX
                                ;                     WHEN TX BUFFER IS COMPLETED.
                                ;               ETX - USED BY DVTXRX TO DETERMINE IF TX BUFFER COMPLETED IS
                                ;                     EXPECTED.
                                ;               ERX - USED BY DVTXRX TO DETERMINE IF RX BUFFER COMPLETED IS
                                ;                     EXPECTED.
                                ;
                                ;
                                ;
                                ; CALLING SEQUENCE:
                                ;       JMP     ALLTR           ;GO TO TRANSMIT-RECEIVE FOR ALL STANDARD MODES
                                ;--


031200                          ALLTR:
031200  032737  000004  007232  ALCK5:  BIT     #QRX,FLAG
031206  001420                          BEQ     ALCK1           ;IF NOT RX GO TO TX'S
031210  013702  007160                  MOV     CPTRR,R2
031214  011237  007176                  MOV     (R2),TEMP2
031220  012237  007126                  MOV     (R2)+,DVRXA
031224  011237  007200                  MOV     (R2),TEMP3
031230  011237  007130                  MOV     (R2),DVRCC
031234  010237  007160                  MOV     R2,CPTRR
031240  004737  034120                  JSR     PC,DVRXQ        ;GO QUE DEVICE
031244  004737  020746                  JSR     PC,LOGRXQ       ;LOG REC QUED
031250  032737  000010  007232  ALCK1:  BIT     #QTX,FLAG
```

```
031256  001416                                  BEQ     ALCK2                   ;IF NO TX'S GO TO 2
031260  013702  007162                          MOV     CPTR,R2
031264  011237  007176                          MOV     (R2),TEMP2
031270  012237  007112                          MOV     (R2)+,DVTXA
031274  011237  007200                          MOV     (R2),TEMP3
031300  012237  007114                          MOV     (R2)+,DVTCC
031304  010237  007162                          MOV     R2,CPTR
031310  004737  020712                          JSR     PC,LOGTXQ

031314  004737  034200                  ALCK2:  JSR     PC,DVTXRX               ;GO TO TX AND RX SUB ROUT.

031320  032737  000004  007232                  BIT     #QRX,FLAG               ;CHECK FOR REC. MSG.
031326  001514                                  BEQ     ALCK3
031330  013737  007126  007176                  MOV     DVRXA,TEMP2
031336  013737  007130  007200                  MOV     DVRCC,TEMP3
031344  004737  020764                          JSR     PC,LOGRXC               ;LOG REC COMPLETE
031350  032737  000004  007226          UPTABL: BIT     #ECHOB,PARAM            ;IS THIS ECHO MODE(PASSIVE)
031356  001406                                  BEQ     UPTA4                   ;IF NOT GO TO 4
031360  013702  007162                          MOV     CPTR,R2                 ;ELSE SET R2 TO PRESENT TX TABL
031364  013722  007176                          MOV     TEMP2,(R2)+             ;STORE OFF RX ADD
031370  013712  007200                          MOV     TEMP3,(R2)              ;AND CC
031374  032737  000002  007226          UPTA4:  BIT     #DATCKB,PARAM           ;IS DATA CHECKING ASKED FOR
031402  001015                                  BNE     UPTA1                   ;IF SO GO TO 1
031404  012737  000001  007132                  MOV     #01,DVRCT               ;ELSE SET DVRCT TO A 1
031412  013737  007076  007160                  MOV     RXPTR,CPTRR             ;RESET POINTER
031420  022737  000003  007220                  CMP     #ACT,MODTYP             ;IS THIS ACTIVE
031426  001002                                  BNE     UPTA3
031430  005237  007132                          INC     DVRCT                   ;IF YES BUMP COUNT
031434  000424                          UPTA3:  BR      UPTEX
031436  013702  007160                  UPTA1:  MOV     CPTRR,R2
031442  011237  007172                          MOV     (R2),TEMP               ;LOAD TEMP WITH PREV. COUNT
031446  163737  007200  007172                  SUB     TEMP3,TEMP              ;LOAD TEMP WITH PREV.COUNT-CURRENT
031454  013722  007200                          MOV     TEMP3,(R2)+
031460  063737  007200  007176                  ADD     TEMP3,TEMP2
031466  013722  007176                          MOV     TEMP2,(R2)+             ;STORE OF NEW ADD
031472  013712  007172                          MOV     TEMP,(R2)               ;AND NEW CC
031476  162702  000002                          SUB     #2,R2                   ;PUT POINTER BACK TO ADDR.
031502  010237  007160                          MOV     R2,CPTRR                ;AND RESTORE IT.
031506                                  UPTEX:
031506  022737  000002  007220                  CMP     #PAS,MODTYP
031514  001007                                  BNE     ALCK2A                  ;IF NOT PASSIVE LOOP THEN GO TO 2A
031516  042737  000104  007232                  BIC     #QRX+#ERX,FLAG          ;CLEAR BOTH EXPECTED AND COMPLETED FLAGS
031524  052737  000210  007232                  BIS     #QTX+#ETX,FLAG          ;SET THE TX FLAGS
031532  000646                                  BR      ALCK1

031534  005337  007132                  ALCK2A: DEC     DVRCT                   ;DEC REC COUNT
031540  005737  007132                          TST     DVRCT                   ;IS IT ALL DONE
031544  001005                                  BNE     ALCK3                   ;NO. GO CHECK TX
031546  042737  000004  007232                  BIC     #QRX,FLAG               ;CLEAR THE RX FLAG
031554  005037  007160                          CLR     CPTRR                   ;YES. CLEAR POINTER
031560  032737  000010  007232          ALCK3:  BIT     #QTX,FLAG               ;IS IT TX
031566  001447                                  BEQ     ALCK4                   ;IF NOT TX THEN GO BACK
031570  013737  007112  007176                  MOV     DVTXA,TEMP2
031576  013737  007114  007200                  MOV     DVTCC,TEMP3             ;LOG TX COMPLETED
031604  004737  020730                          JSR     PC,LOGTXC
031610  005337  007116                          DEC     DVTCT                   ;DEC TX COUNT
```

```
031614  022737  000002  007220              CMP     #PAS,MODTYP
031622  001013                              BNE     ALCK3A              ;IF NOT PASSIVE MODE GO TO 3A
031624  042737  000210  007232              BIC     #QTX+ETX,FLAG       ;CLEAR THE TX FLAGS
031632  052737  000104  007232              BIS     #QRX+ERX,FLAG       ;AND SET THE RX FLAGS
031640  005737  007116                      TST     DVTCT
031644  001005                              BNE     ALCK3C              ;IF MORE RX'S DO IT
031646  000137  031726                      JMP     CMPSR               ; ELSE COMPARE
031652  005737  007116              ALCK3A: TST     DVTCT               ;IS IT ALL DONE
031656  001402                              BEQ     ALCK3B              ;IF NOT GO BACK TO 5
031660  000137  031200              ALCK3C: JMP     ALCK5
031664  005037  007162              ALCK3B: CLR     CPTR                ;IF SO CLEAR POINTER
031670  042737  000010  007232              BIC     #QTX,FLAG           ;CLEAR TX FLAG
031676  032737  000002  007226              BIT     #DATCKB,PARAM       ;IS IT DAT CK
031704  001403                              BEQ     ALCK4A              ;IF NOT THEN END WO CKING RX.
031706  005737  007160              ALCK4:  TST     CPTRR

031712  001362                              BNE     ALCK3C              ;IF SOME RX'S LEFT GO BACK
031714  005737  007162              ALCK4A: TST     CPTR
031720  001402                              BEQ     ALCK4B              ;BRANCH IF ANY TX'S LEFT
031722  000137  031314                      JMP     ALCK2
031726                              ALCK4B:
```

```
                                .SBTTL          DATA COMPARISON CODE


                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;
                                ;       CMPSR - COMPARE CODE
                                ;       THIS CODE COMPARES THE RECEIVED DATA AGAINST THE
                                ;       EXPECTED AND FILLS THE EVENT LOG WITH 1 OF 3 MSGS.
                                ;
                                ;       NOTE: IF NO DATA CHECKING SKIP THIS CODE
                                ;
                                ;       1.) A DATA COMPARISON ENTRY WHICH REPORTS THE NUMBER
                                ;           OF COMPARISON ERRORS FOUND.
                                ;       2.) A DATA COMPARISON ENTRY WHICH REPORTS DIFFERENCES
                                ;           IN REC LENGTH TO COMPARE LENGTH.
                                ;       3.) A DATA COMPARISON STARTED ENTRY WHICH REPORTS ADDRESS
                                ;           OF RECEIVE BUFFER AND BYTE COUNT.
                                ;       THIS CODE ALSO REPORTS SOFT ERRORS FOR DATA COMPARISON
                                ;       (THE FIRST 5 ONLY),LENGTH ERROR,AND TOTAL NUMBER OF ERRORS
                                ;
                                ; SUBORDINATE ROUTINES USED:
                                ;
                                ;               'LOGCMP'' - SEE ITEM 3 ABOVE
                                ;               'LOGCML'' - SEE ITEM 2 ABOVE
                                ;               'LOGCMD'' - SEE ITEM 1 ABOVE
                                ;
                                ; CALLING SEQUENCE:
                                ;       JMP     CMPSR           ;JUMP TO DATA COMPARISON CODE
                                ;--

031726  032737  000002  007226  CMPSR:  BIT     #DATCKB,PARAM   ;IS DATA CHECKING TO BE DONE
031734  001522                          BEQ     CMPSEX          ;IF NOT THEN EXIT
031736  013737  007076  007162          MOV     RXPTR,CPTR      ;PUT START OF RX POINTERS TO CPTR
031744  013737  007102  007160          MOV     CMPPTR,CPTRR    ; AND START OF COMPARE POINTS TO CPTRR
031752  013737  007134  007132          MOV     RXMTOT,DVRCT

031760                          CMPS3:
031760  013702  007162                  MOV     CPTR,R2         ;MOVE CURRET RX PT.TO R2
031764  011237  007176                  MOV     (R2),TEMP2      ;MOVE RX ADD TO EVENT LOG
031770  012201                          MOV     (R2)+,R1        ;SET R1 TO START ADD OF RX
031772  012237  007200                  MOV     (R2)+,TEMP3     ;SET CHAR  COUNT TO EVENT LOG
031776  010237  007162                  MOV     R2,CPTR         ;RESTORE RX POINT

032002  013702  007160                  MOV     CPTRR,R2        ;PUT R2 AT COMPARE TABLE
032006  012203                          MOV     (R2)+,R3        ;SET R3 TO COMPARE ADD
032010  012204                          MOV     (R2)+,R4        ;SET R4 TO COMP CC
032012  010237  007160                  MOV     R2,CPTRR        ;RESTORE POINTER
032016  010437  007202                  MOV     R4,TEMP4
032022  004737  021060                  JSR     PC,LOGCMP       ;LOG COMPARE START.

032026  020437  007200                  CMP     R4,TEMP3        ;IS COMPARE COUNT = TO RX COUNT
032032  001410                          BEQ     CMPS7           ;IF SO GO TO 7
032034  005237  007144                  INC     ERRCNT
032040                                  ERRSOFT 1,EDDLE,ERR10   ;PRINT ERROR
```

```
032040  104457                                                                                  TRAP    C$ERSOFT
032042  000001                                                                                  .WORD   1
032044  014651                                                                                  .WORD   EDDLE
032046  020260                                                                                  .WORD   ERR10
032050  004737  021076                    JSR       PC,LOGCML          ;LOG LENGTH ERROR

032054  005037  007202           CMPS7:   CLR       TEMP4              ;CLEAR BAD BYTE COUNTER
032060  012737  000001  007170            MOV       #1,OFSET           ;SET OFSET BYTE COUNT TO 1
032066  122123                   CMPS1:   CMPB      (R1)+,(R3)+        ;COMPARE RX WITH EXPETED
032070  001422                            BEQ       CMPS6              ;IF EQUAL THEN GO TO 6

032072  005237  007202           CMPS2:   INC       TEMP4              ;INC BAD COUNT
032076  023727  007202  000005            CMP       TEMP4,#5           ;IS IT MORE THEN 5
032104  101014                            BHI       CMPS6              ;IF SO GO  FOR MORE
032106  114337  007210                    MOVB      -(R3),GOOD         ;STORE GOOD BYTE FOR ERROR
032112  114137  007211                    MOVB      -(R1),BAD          ;STORE BAD BYTE FOR ERROR
032116  005237  007144                    INC       ERRCNT
032122                                    ERRSOFT 2,EDDDE,ERR1         ;REPORT COMPARISON FAILURE TO OPR.
032122  104457                                                                                  TRAP    C$ERSOFT
032124  000002                                                                                  .WORD   2
032126  014706                                                                                  .WORD   EDDDE
032130  020170                                                                                  .WORD   ERR1
032132  005201                            INC       R1
032134  005203                            INC       R3
032136  005237  007170           CMPS6:   INC       OFSET              ;INC OFFSET
032142  005304                            DEC       R4                 ;ELSE DEC CHAR COUNT AND SEE IF 0
032144  001350                            BNE       CMPS1              ;IF NOT GO BACK
032146  005737  007202                    TST       TEMP4              ;SEE IF ANY CMP ERRS FOR THIS MSG
032152  001410                            BEQ       CMPS5A             ;BR IF NONE
032154  005237  007144                    INC       ERRCNT
032160                                    ERRSOFT 3,EDDDE,ERR2         ;REPORT # OF MISMATCHES FOR MESSAGE
032160  104457                                                                                  TRAP    C$ERSOFT
032162  000003                                                                                  .WORD   3
032164  014706                                                                                  .WORD   EDDDE
032166  020232                                                                                  .WORD   ERR2
032170  004737  021114           CMPS5:   JSR       PC,LOGCMD          ;LOG DATA ERROR IN COMPARE
032174                           CMPS5A:
032174  005337  007132                    DEC       DVRCT
032200  001267                            BNE       CMPS3              ;IF NOT ALL DONE GO BACK

```

```
                                        .SBTTL          INTERNAL END OF PASS CODE


                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;               THIS CODE INCREMENTS THE PASS COUNT FOR THE
                                        ;               EVENT LOG, LOGS THE END OF PASS EVENT
                                        ;               IF 'RPASS'' IS A MINUS ONE RETURN TO MODE
                                        ;               DISPATCHER. IF NOT -1 THEN DECREMENT RPASS
                                        ;               AND IF 'RPASS'' IS THEN = TO 0 GO TO DCLT PROMT
                                        ;               IN NOT = TO 0 THEN GO BACK TO MODE DISPATCHER
                                        ;
                                        ; SUBORDINATE ROUTINES USED:
                                        ;
                                        ;                       'LOGEOP'' - LOG END OF PASS TO EVENT LOG
                                        ;
                                        ;------

 032202  005237  007142                 CMPSEX: INC     PSCNT                   ;BUMP PASS COUNT

 032206  013737  007140  007202                 MOV     NOBUF,TEMP4
 032214  013737  007142  007176                 MOV     PSCNT,TEMP2
 032222  013737  007144  007200                 MOV     ERRCNT,TEMP3
 032230  004737  021132                         JSR     PC,LOGEOP               ;LOG END OF PASS

 032234  022737  177777  007230                 CMP     #-1,RPASS               ;SEE IF RPASS=-1
 032242  001403                                 BEQ     1$                      ;IF IT IS DON'T DECRMNT, LOOP FOREVER
 032244  005337  007230                         DEC     RPASS                   ;DEC PASS COUNT
 032250  001402                                 BEQ     2$                      ;IF DONE EXIT TEST
 032252  000137  030746                 1$:     JMP     GTRX2                   ;ELSE GO BACK AND DISPATCH
 032256  000137  025640                 2$:     JMP     GTRA5                   ;WHEN RPASS=0 GO BACK TO 'DCLT>''
```

```
                                                .SBTTL          DOWN-LINE-LOAD SECTION

                                                ;++
                                                ; FUNCTIONAL DESCRIPTION:
                                                ;       DOWN-LINE-LOAD SECTION
                                                ;       IN THIS MODE OF TESTING THE 'HOST'' OR ORIGINATING STATION
                                                ;       REQUESTS THE 'SATELLITE'' OR BOOT STATION TO ENTER MOP MODE.
                                                ;       THE BOOT STATION THEN SENDS A  'REQUEST PROGRAM MESSAGE''.
                                                ;       THE 'HOST'' THEN SENDS A 'MEMORY LOAD WITH TRANSFER ADDRESS''
                                                ;       THAT CONTAINS IMAGE DATA TO BE LOADED BY THE BOOT STATION'S
                                                ;       M9312 STARTING AT LOC. 0.  THIS IMAGE DATA WILL CONTAIN A
                                                ;       PROGRAM THAT WILL PRINT A MSG THAT DOWN-LINE-LOAD WAS SUCESSFUL.
                                                ;
                                                ; SUBORDINATE ROUTINES USED:
                                                ;
                                                ;               'DLTXRX'' - SPECIAL TX RX ROUTINE FOR DLL
                                                ;               'DVRXQ''  - QUE RX BUFFER SPACE TO DEVICE
                                                ;               'LOGRXQ'' - LOG RX SPACE QUED TO EVENT LOG
                                                ;               'LOGTXQ'' - LOG TX BUFFER QUED TO EVENT LOG
                                                ;               'DVTXRX'' - QUE TX BUFFER AND WAIT FOR RX OR TX TO COMPLETE
                                                ;               'LOGTXC'' - LOG TX COMPLETED TO EVENT LOG
                                                ;               'LOGRXC'' - LOG RX COMPLETED TO EVENT LOG
                                                ;
                                                ; CALLING SEQUENCE:
                                                ;       JMP     @MODE(R2)       ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
                                                ;
                                                ;--

032262  052737  000100  007232  DLL:    BIS     #ERX,FLAG       ;SET EXPECTED TO RX
032270  042737  000002  007226          BIC     #DATCKB,PARAM   ;CLEAR NOCHECK
032276  012737  002647  007164          MOV     #DLLM1,CURADD   ;SET THE DOWN LINE LOAD MSG TO #1
032304  013737  002172  007156          MOV     DLLM1C,CURCC    ;SET THE CC
032312  004737  032404                  JSR     PC,DLTXRX       ;GO TO THE DOWN LINE TX RX ROUTINE

                                        ;RETURN WHEN TX AND RX ARE COMPLETED

032316  012737  002654  007164          MOV     #DLLM2,CURADD   ;SET THE DOWN LINE LOAD MSG TO #2
032324  013737  002174  007156          MOV     DLLM2C,CURCC    ;SET CC
032332  042737  000400  007232          BIC     #DLLGA,FLAG     ;CLEAR THE GO AHEAD FLAG
032340  004737  032404                  JSR     PC,DLTXRX       ;GO TO THE DOWN LINE TX RX ROUTINE

                                        ; RETURN WHEN TX AND RX ARE COMPLETED
032344                          DLLPRI:
032344                                  PRINTF  #DLLCM
032344  012746  013640                                                          MOV     #DLLCM,-(SP)
032350  012746  000001                                                          MOV     #1,-(SP)
032354  010600                                                                  MOV     SP,R0
032356  104417                                                                  TRAP    C$PNTF
032360  062706  000004                                                          ADD     #4,SP
032364  000137  025640                  JMP     GTRA5

032370                          DLLEA:
032370                                  ERRHRD  20,DLLAB,ERR14
032370  104456                                                                  TRAP    C$ERHRD
032372  000024                                                                  .WORD   20
032374  017130                                                                  .WORD   DLLAB
032376  020502                                                                  .WORD   ERR14
```

```

032400  000137  025640                   JMP     GTRA5            ;PRINT ABORT AND EXIT



032404                          DLTXRX:
032404  052737  000004  007232           BIS     #QRX,FLAG        ;SET THE QUE RX FLAG
032412  012737  004326  007126           MOV     #RXBUF,DVRXA     ;SET THE DEVICE RX BUFFER TO RXBUF
032420  012737  004326  007176           MOV     #RXBUF,TEMP2     ;SET UP FOR LOG
032426  012737  000400  007130           MOV     #256.,DVRCC      ;SET UP FOR CC OF 256
032434  012737  000400  007200           MOV     #256.,TEMP3      ;SET UP FOR LOG
032442  004737  034120                   JSR     PC,DVRXQ         ; GO QUE RX
032446  004737  020746                   JSR     PC,LOGRXQ        ;AND LOG IT...

032452  013737  007164  007112           MOV     CURADD,DVTXA     ;SET UP FOR TX
032460  013737  007164  007176           MOV     CURADD,TEMP2     ;AND LOG
032466  013737  007156  007114           MOV     CURCC,DVTCC      ;SE UP FOR TX COUNT
032474  013737  007156  007200           MOV     CURCC,TEMP3      ;AND LOG IT
032502  004737  020712                   JSR     PC,LOGTXQ        ;LOG THE TX QUEUED
032506  052737  000210  007232           BIS     #QTX+#ETX,FLAG   ;SET UP TO QUE AND EXPECTED
032514  004737  034200          DLLE2:   JSR     PC,DVTXRX        ;GO TO DEVICE ROUTINE
032520  032737  000400  007232           BIT     #DLLGA,FLAG      ;TEST FOR GO AHEAD BIT
032526  001047                           BNE     DLLE1            ;IF SET GO TO ONE
032530  032737  000010  007232           BIT     #QTX,FLAG        ;ELSE CHECK FOR TX DONE
032536  001020                           BNE     DLLE6            ;IF DONE THEN BRANCH
                                                                  ;ELSE ERROR
032540  012737  020062  007204           MOV     #TXNC,CONOTM
032546  013737  004326  007200  DLLE7:   MOV     RXBUF,TEMP3
032554  013737  003326  007202           MOV     TXBUF,TEMP4
032562  012737  017130  007176           MOV     #DLLAB,TEMP2
032570  004737  020774                   JSR     PC,LGDVE         ;LOG ERROR
032574  000137  032370                   JMP     DLLEA            ;ABORT TEST

032600  013737  007112  007176  DLLE6:   MOV     DVTXA,TEMP2
032606  013737  007114  007200           MOV     DVTCC,TEMP3
032614  004737  020730                   JSR     PC,LOGTXC        ;LOG TX DONE
032620  042737  000210  007232           BIC     #QTX+#ETX,FLAG   ;CLEAR QUE AND EXPECTED
032626  052737  000400  007232           BIS     #DLLGA,FLAG      ;SET THE GO AHEAD BIT
032634  023737  002174  007114           CMP     DLLM2C,DVTCC
032642  001441                           BEQ     DLLE5            ;EXIT IF SECOND MSG.
032644  000723                           BR      DLLE2            ;AND GO BACK TO 2
032646  032737  000004  007232  DLLE1:   BIT     #QRX,FLAG        ;IS THE A RX COMPLETED
032654  001004                           BNE     DLLE8            ;IF SO GO TO 8
032656  012737  020102  007204           MOV     #RXNC,CONOTM     ;ELSE SET UP ERROR AND ABORT.
032664  000730                           BR      DLLE7
032666  013737  007126  007176  DLLE8:   MOV     DVRXA,TEMP2
032674  013737  007130  007200           MOV     DVRCC,TEMP3
032702  004737  020764                   JSR     PC,LOGRXC        ;LOG RECEIVE COMPLETE
032706  022737  006010  004326           CMP     #6010,RXBUF      ;CHECK FOR FIRST WORD OF RX
                                                                  ;SEC BOOT MSG.
032714  001404                           BEQ     DLLE3
032716  012737  020122  007204  DLLE4:   MOV     #RXM1,CONOTM     ;SET UP MESG AND ABORT
032724  000710                           BR      DLLE7            ;ABORT TEST

032726  022737  000001  004330  DLLE3:   CMP     #1,RXBUF+2       ;IS SECOND WORD 1
032734  001404                           BEQ     DLLE5            ;IF OK RETURN
```

```
032736  012737  020145  007204            MOV     #RXM2,CONOTM
032744  000700                            BR      DLLE7              ;SET UP MESSAGE AND ABORT

032746  000207                  DLLE5:    RTS     PC                 ;RETURN TO CALLER



```

```
                                        .SBTTL          TALK MODE SECTION

                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;               TALK MODE SECTION
                                        ;               IN THIS MODE, THE ''TALK'' END OF THE LINK TRANSMITS OPERATOR
                                        ;               SPECIFIED MESSAGES UNTIL A 'EXIT'' MESSAGE IS TYPE.  AT THAT POINT,
                                        ;               THIS END OF THE LINK GOES INTO 'LISTEN'' MODE.
                                        ;
                                        ; SUBORDINATE ROUTINES USED:
                                        ;
                                        ;                       'LOGTXQ'' - LOG TX BUFFER QUED TO EVENT LOG
                                        ;                       'DVTXRX'' - QUE TX BUFFER TO DEVICE AND WAIT FOR COMPLETE
                                        ;                       'LOGTXC'' - LOG TX COMPLETE TO EVENT LOG
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;               JMP     @MODE(R2)               ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
                                        ;--

 032750                                 TALCK:
 032750  042737  000002  007226                 BIC     #DATCKB,PARAM           ;SET NOCHECK
 032756  012702  002524                         MOV     #OPBUF,R2
 032762  012722  177777                 1$:     MOV     #-1,(R2)+               ;CLEAR OUT OPBUFFER FIRST
 032766  022702  002646                         CMP     #OPEND,R2
 032772  001373                                 BNE     1$
 032774                                         GMANID  OPRMM,OPBUF,A,0,1,72.,NO            ;GET TALK MESSAGE
 032774  104443                                                                                     TRAP    C$GMAN
 032776  000406                                                                                     BR      10001$
 033000  002524                                                                                     .WORD   OPBUF
 033002  000142                                                                                     .WORD   T$CODE
 033004  013575                                                                                     .WORD   OPRMM
 033006  000000                                                                                     .WORD   0
 033010  000001                                                                                     .WORD   T$LOLIM
 033012  000110                                                                                     .WORD   T$HILIM
 033014                                                                                     10001$:
 033014  005002                                 CLR     R2                      ;NOW GET CHAR COUNT
 033016  122762  000377  002524         2$:     CMPB    #377,OPBUF(R2)
 033024  001402                                 BEQ     3$
 033026  005202                                 INC     R2
 033030  000772                                 BR      2$
 033032  010237  002166                 3$:     MOV     R2,OPCNT

 033036  012737  002524  007112                 MOV     #OPBUF,DVTXA            ;SET UP TX ADDR.
 033044  012737  002524  007176                 MOV     #OPBUF,TEMP2
 033052  013737  002166  007200                 MOV     OPCNT,TEMP3
 033060  013737  002166  007114                 MOV     OPCNT,DVTCC             ;SET UP TX CC
 033066  004737  020712                         JSR     PC,LOGTXQ
 033072  052737  000210  007232                 BIS     #QTX+#ETX,FLAG          ;SET UP FLAGS
 033100  005037  007160                         CLR     CPTRR                   ;CLEAR RX POINTER

 033104  004737  034200                         JSR     PC,DVTXRX

 033110  013737  007112  007176                 MOV     DVTXA,TEMP2
 033116  013737  007114  007200                 MOV     DVTCC,TEMP3
 033124  004737  020730                         JSR     PC,LOGTXC
 033130  022737  054105  002524                 CMP     #'EX,OPBUF              ;CHECK FOR EXIT
```

```
033136  001304                          BNE     TALCK
033140  022737  052111  002526          CMP     #''IT,OPBUF+2
033146  001300                          BNE     TALCK
033150  042737  000210  007232          BIC     #QTX+#ETX,FLAG  ;CLEAR THE TX BITS
033156  012737  000006  007220          MOV     #LIS,MODTYP     ;CHANGE TO LISTEN MODE
033164  000137  030746                  JMP     GTRX2           ;AND GO BACK TO DISPATCH
```

```
                                    .SBTTL          LISTEN MODE SECTION

                                    ;++
                                    ; FUNCTIONAL DESCRIPTION:
                                    ;       LISTEN MODE SECTION
                                    ;       IN THIS MODE, THE ''LISTEN'' END OF THE LINK PRINTS ALL OF THE MESSAGES
                                    ;       RECEIVED BY THE DEVICE ON THE OPERATOR'S CONSOLE.  IF THE MESSAGE
                                    ;       RECEIVED IS AN ''EXIT'' MESSAGE, THEN THE NODE ENTERS ''TALK'' MODE.
                                    ;
                                    ; SUBORDINATE ROUTINES USED:
                                    ;
                                    ;               ''DVRXQ'' - QUE RECEIVE BUFFER SPACE TO DEVICE
                                    ;               ''LOGRXQ'' - LOG RECEIVE BUFFER QUED TO EVENT LOG
                                    ;               ''DVTXRX'' - WAIT FOR RX TO COMPLETE
                                    ;               ''LOGRXC'' - LOG RX COMPLETE TO EVENT LOG
                                    ;
                                    ; CALLING SEQUENCE:
                                    ;       JMP     @MODE(R2)               ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
                                    ;--

033170  042737  000002  007226  LISCK:  BIC     #DATCKB,PARAM           ;CLEAR CHECK BIT
033176                                  PRINTF  #LISP                   ;PRINT PROMPT FOR OPR.
033176  012746  013564                                                                          MOV     #LISP,-(SP)
033202  012746  000001                                                                          MOV     #1,-(SP)
033206  010600                                                                                  MOV     SP,R0
033210  104417                                                                                  TRAP    C$PNTF
033212  062706  000004                                                                          ADD     #4,SP
033216  012737  002524  007126  LISCKA: MOV     #OPBUF,DVRXA            ;SET DEVICE UP TO REC AT OPBUF
033224  012737  002524  007176          MOV     #OPBUF,TEMP2
033232  012737  000122  007130          MOV     #82.,DVRCC              ;SET UP CHAR COUNT TO 82.
033240  012737  000122  007200          MOV     #82.,TEMP3
033246  052737  000104  007232          BIS     #QRX+#ERX,FLAG          ;SET UP FLAG
033254  005037  007162                  CLR     CPTR                    ;CLEAR THE TX.

033260  004737  034120                  JSR     PC,DVRXQ                ;QUE RX
033264  004737  020746                  JSR     PC,LOGRXQ

033270  004737  034200                  JSR     PC,DVTXRX               ;GO TO DEVICE RX. SUBROUTINE

033274  013737  007126  007176          MOV     DVRXA,TEMP2
033302  013737  007130  007200          MOV     DVRCC,TEMP3             ;SET UP ADDR.AND CC.
033310  004737  020764                  JSR     PC,LOGRXC               ;LOG COMPLETED
033314  063737  007126  007130          ADD     DVRXA,DVRCC
033322  105077  153602                  CLRB    @DVRCC
033326                                  PRINTF  #OPBFPT
033326  012746  002520                                                                          MOV     #OPBFPT,-(SP)
033332  012746  000001                                                                          MOV     #1,-(SP)
033336  010600                                                                                  MOV     SP,R0
033340  104417                                                                                  TRAP    C$PNTF
033342  062706  000004                                                                          ADD     #4,SP
033346  022737  054105  002524          CMP     #'EX,OPBUF              ;COMPARE FOR EX OF ''EXIT''
033354  001320                          BNE     LISCKA                  ;IF NOT EXIT THEN GO BACK
033356  022737  052111  002526          CMP     #'IT,OPBUF+2            ;IF FIRST HALF OK CHECK NEXT PART
033364  001314                          BNE     LISCKA                  ;IF NOT EXIT THE GO BACK
033366  012737  000005  007220          MOV     #TAL,MODTYP             ;CHANGE MODE TO TALK
033374  000137  030746                  JMP     GTRX2                   ;RETURN TO DISPATCHER
```

```
5636
5637
```

```
                                    .SBTTL          DEVICE FUNCTION SUBROUTINES




                                    .SBTTL                  DEVICE INIT SUBROUTINE


                                    ;++
                                    ; FUNCTIONAL DESCRIPTION:
                                    ;       DVINIT- DEVICE INIT ROUTINE
                                    ;       THIS ROUTINE IS DEVICE DEPENDENT CODE THAT INITS
                                    ;       THE DEVICE BEING TESTED. (I.E. FULL/HALF DUPLEX BAUD RATE, MAINT MODE.)
                                    ;
                                    ; INPUTS:       ''FHDPLX'' INDICATES IF MODE IS FULL OR HALF DUPLEX. (1=FULL)
                                    ;               ADDRESS POINTERS (SEL0,...) ALREADY POINT TO DEVICE'S REG.S
                                    ;
                                    ; SUBORDINATE ROUTINES USED:
                                    ;
                                    ;                ''LGDVE'' - LOG DEVICE ERROR TO EVENT LOG
                                    ;               ''TOORIO'' - TIME OUT OR INPUT INTERRUPT OR OUTPUT INTERRUPT
                                    ;                ''CLRAW'' - CLEAR RQI AND WAIT FOR RDI TO GO AWAY
                                    ;
                                    ;
                                    ; CALLING SEQUENCE:
                                    ;               JSR     PC,DVINIT
                                    ;--

033400                              DVINIT:
                                            ;MASTER CLEAR DEVICE

033400  012737  000100  007272              MOV     #100,TIMER1         ;SET UP TIMER 1 FOR 100(OCTAL) TICKS
033406  005077  156400                      CLR     @SEL6
033412  005077  156370                      CLR     @SEL4
033416  012777  040000  156352              MOV     #MCLR,@SEL0         ;DO A MASTER CLEAR

033424  022737  000004  012024              CMP     #DMRC6,OPTYP        ;IS THIS A 8206
033432  001003                              BNE     DVIN6               ;IF NOT GO TO 6
033434  112777  000200  156336              MOVB    #200,@BSEL1         ;SET RUN FOR 8206
033442  022737  000006  012024      DVIN6:  CMP     #DMR6,OPTYP         ;IS THIS AN 8206 DMR
033450  001003                              BNE     DVIN2               ;IF NOT GO TO 2
033452  112777  000200  156320              MOVB    #200,@BSEL1         ;SET RUN BIT FOR 8206

033460  005777  156312              DVIN2:  TST     @SEL0               ;IS RUN BIT SET
033464  100426                              BMI     DVIN1               ;IF YES GO TO 1 ELSE...
033466                                      BREAK
033466  104422                                                                                  TRAP    C$BRK
033470  005737  007272                      TST     TIMER1              ;SEE IF TIME HAS EXPIRED
033474  001371                              BNE     DVIN2               ;IF NOT GO BACK AND CHECK
                                                                        ;AGAIN ELSE...PRINT ERROR
033476  012737  016340  007176              MOV     #DVEM3,TEMP2
033504  017737  156266  007200              MOV     @SEL0,TEMP3
033512  017737  156264  007202              MOV     @SEL2,TEMP4         ;LOAD UP ERRM. AND REG OUTPUTS
033520  004737  020774                      JSR     PC,LGDVE            ;LOG TIME OUT WAITING FOR RUN
033524  005237  007144                      INC     ERRCNT
033530                                      ERRSOFT 11,DVEM3,ERR13
```

```
033530  104457                                                          TRAP    C$ERSOFT
033532  000013                                                          .WORD   11
033534  016340                                                          .WORD   DVEM3
033536  020450                                                          .WORD   ERR13

033540  000717                                  BR DVINIT                       ;GO BACK AND TRY MSTR CLR AGAIN IF ERROR

033542                          DVIN1:

                                        ; DO BASE IN COMMAND

033542  042737  000003  007232          BIC     #3,FLAG                 ;CLEAR INPUT AND OUTPUT INT FLAGS
033550  112777  000143  156220          MOVB    #143,@BSEL0             ;SET UP BASE IN INT EN
033556  004737  035026                  JSR     PC,TOORIO               ;GO WAIT FOR INTERRUPT OR TIME OUT
033562  012777  017370  156216          MOV     #BASE,@SEL4

033570  012777  000000  156214          MOV     #0,@SEL6                ;SET UP SEL 6
033576  023727  012024  000006          CMP     OPTYP,#6                ;IS THIS DMR MODE
033604  002403                          BLT     DVIN7                   ;IF NOT GO TO 7
033606  012777  000522  156176          MOV     #522,@SEL6              ;SET DMR MODE
033614  052777  000100  156160  DVIN7:  BIS     #IEO,@SEL2              ;SET IEO
033622  042777  004000  156146          BIC     #LULOOP,@SEL0           ;CLEAR LU LOOP
033630  022737  000001  007222          CMP     #TTL,MLTYP              ;IS TTL SELECTED
033636  001003                          BNE     DVIN3                   ; IF NOT GO TO 3
033640  052777  004000  156130          BIS     #LULOOP,@SEL0           ;ELSE SET LU LOOP
033646  004737  034720          DVIN3:  JSR     PC,CLRAW

                                        ; DO WRITE MODEM IF DMR MODE

033652  023727  012024  000006          CMP     OPTYP,#6                ;IS THIS DMR MODE
033660  002437                          BLT     DVIN8                   ;IF NOT GO TO 8
033662  112777  000145  156106          MOVB    #145,@BSEL0             ;SET UP WRITE MODEM
033670  004737  035026                  JSR     PC,TOORIO               ;GO TO  WAIT FOR INT
033674  042777  000014  156110          BIC     #BIT2+#BIT3,@SEL6         ;CLEAR BSEL6 AND 7
033702  022737  000004  007222          CMP     #MODREM,MLTYP           ;IS THIS REMOTE LOOP
033710  001003                          BNE     DVIN9                   ;IF NOT GO TO 9
033712  052777  000004  156072          BIS     #BIT2,@BSEL6            ;SET THE BIT
033720  022737  000003  007222  DVIN9:  CMP     #MODLOC,MLTYP           ;IS IT MODEM LOCAL
033726  001003                          BNE     DVIN10                  ;IF NOT EXIT
033730  052777  000010  156054          BIS     #BIT3,@BSEL6            ;SET MODEM LOCAL
033736  004737  034720          DVIN10: JSR     PC,CLRAW                ;CLEAR RDI AND WAIT

                                        ; ENABLE EXTENDED ERROR IF DMR MODE


033742  112777  000146  156026          MOVB    #146,@BSEL0             ;SET UP FOR ENABLE
033750  004737  035026                  JSR     PC,TOORIO
033754  004737  034720                  JSR     PC,CLRAW                ;CLEAR RDI AND WAIT

                                        ; DO CONTROL IN COMMAND

033760  112777  000141  156010  DVIN8:  MOVB    #141,@BSEL0             ;SET UP CONTROL IN
033766  004737  035026                  JSR     PC,TOORIO               ;WAIT FOR INT OR TIME OUT
033772  005077  156014                  CLR     @SEL6                   ;CLEAR HALF/DUP
033776  022737  000004  007220          CMP     #DOW,MODTYP             ;IS THIS DOWN LINE LOAD?
034004  001004                          BNE     DVIN5                   ; BR IF NOT
```

```
034006  052777  002400  155776            BIS     #MAINTB+HALFDB,@SEL6    ;IF SO SET MAINT MODE  BIT
034014  000406                            BR      DVIN4                   ; AND FORCE HALF DUPLEX

034016  005737  007224            DVIN5:  TST     FHDPLX                  ;IS THIS A HALF/DUP
034022  001003                            BNE     DVIN4                   ;IF NOT GO TO 4
034024  052777  002000  155760            BIS     #HALFDB,@SEL6           ;ELSE SET HALF/DUP

034032  017737  155754  007206    DVIN4:  MOV     @SEL6,CONTIN            ;SET UP CONTROL IN FOR MODS
034040  004737  034720                    JSR     PC,CLRAW                ;GO CLEAR RQI AND WAIT
                                                                          ;FOR RDI TO GO AWAY.
034044  023727  012024  000006            CMP     OPTYP,#6                ;IS THIS DMR
034052  002403                            BLT     DVINEX                  ;IF NOT EXIT
034054  052737  001000  007232            BIS     #DMRRUN,FLAG            ;SET RUN OUTPUT EXPECTED BIT

034062  000207                    DVINEX: RTS     PC                      ;RETURN TO CALLER
```

```
                                           .SBTTL                DEVICE GET MODEM STATUS SUBROUTINE



                                           ;++
                                           ; FUNCTIONAL DESCRIPTION:
                                           ;       'DVMODS''   GET MODEM STATUS
                                           ;
                                           ; IMPLICIT INPUTS:
                                           ;       THE BIT POSITION AND AVAIABLITY OF THE MODEM SIGNALS CTS,DSR,....RI,.
                                           ;       IN THE DEPENDENT PORITION OF THE GLOBAL EQUATES SECTION.
                                           ;
                                           ; OUTPUTS:
                                           ;       CURRENT MODEM SIGNAL VALUES IN 'MODS''
                                           ;
                                           ; SUBORDINATE ROUTINES USED:
                                           ;
                                           ;               'TOORIO'' - TIME OUT OR INPUT INTERRUPT OR OUTPUT INTERRUPT
                                           ;                ''CLRAW'' - CLEAR RQI AND WAIT FOR RDI TO CLEAR
                                           ;
                                           ;
                                           ; CALLING SEQUENCE:
                                           ;       JSR     PC,DVMODS
                                           ;--

034064  112777  000141  155704  DVMODS: MOVB    #141,@BSEL0       ;SET UP CONTORL IN
034072  004737  035026                  JSR     PC,TOORIO         ;GO TIME OUT CHECK
034076  017737  155704  010206          MOV     @SEL4,MODS        ;SET UP MODEM STATUS
034104  013777  007206  155700          MOV     CONTIN,@SEL6      ;SET UP OLD CONTORL IN
034112  004737  034720                  JSR     PC,CLRAW
034116  000207                          RTS     PC                ;RETURN TO CALLER


```

```
                                                .SBTTL          DEVICE QUEUE RECEIVE SPACE SUBROUTINE

                                                ;++
                                                ; FUNCTIONAL DESCRIPTION:
                                                ;       DVRXQ - THIS SUB ROUTINE QUES THE REC BUFFER SPACE TO THE
                                                ;               DEVICE, THEN CLEARS THE QRX BIT OF THE FLAG WORD.
                                                ;
                                                ; INPUTS:
                                                ;       DVRXA = ADDRESS OF RX BUFFER SPACE
                                                ;       DVRCC = BYTE CHAR COUNT OF RX BUFFER
                                                ;       QRX FLAG BIT = SET BY CALLING ROUTINE
                                                ;
                                                ; OUTPUTS:
                                                ;       QRX FLAG BIT = CLEARED BY ROUTINE
                                                ;
                                                ;
                                                ; SUBORDINATE ROUTINES USED:
                                                ;
                                                ;               "TOORIO" - TIME OUT OR OUTPUT INTERRUPT OR INTPUT INTERRUPT
                                                ;               "CLRAW"  - CLEAR RQI AND WAIT FOR RDI TO CLEAR
                                                ;
                                                ; CALLING SEQUENCE:
                                                ;       JSR     PC,DVRXQ
                                                ;--

034120                                          DVRXQ:
034120  032737  000004  007232                          BIT     #QRX,FLAG
034126  001423                                          BEQ     DVREX                   ;IF NOT RX THEN EXIT
                                                                                        ;ELSE QUE RX
034130  042737  000004  007232                          BIC     #QRX,FLAG               ;CLEAR FLAG FOR RX
034136  112777  000144  155632                          MOVB    #144,@BSEL0
034144  004737  035026                                  JSR     PC,TOORIO               ;GO CHECK FOR IN OR OUT
034150  017737  155632  010206                          MOV     @SEL4,MODS              ;SET UP NEW MOD STATUS
034156  013777  007126  155622                          MOV     DVRXA,@SEL4
034164  013777  007130  155620                          MOV     DVRCC,@SEL6             ;LOAD CC AND ADDR
034172  004737  034720                                  JSR     PC,CLRAW                ;CLEAR AND WAIT
034176  000207                                  DVREX:  RTS     PC                      ;RETURN TO CALLER
```

```
                                        .SBTTL                  DEVICE TRANSMIT AND RECEIVE SUBROUTINE


                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       DVTXRX-DEVICE TRANSMIT AND RECEIVE ROUTINE
                                        ;       THIS CODE QUES THE TRANSMIT BUFFER TO THE DEVICE
                                        ;       IF NEEDED. THE CODE THEN WAITS FOR A TX COMPLE,
                                        ;       RX COMPLETE OR BOTH. THE CODE REPORTS A TIME OUT
                                        ;       ERROR IF NO BACC OUTPUT INTERRUPT IS RECIEVED BEFORE
                                        ;       60 SECONDS. AFTER REPORTING ERROR TIMER IS RE STARTED
                                        ;       AND DEVICE WILL CONTINUE TO WAIT FOR INTERRUPT. CODE
                                        ;       ALSO REPORTS ERROR IF INPUT INTERRUPT OCCURS WHEN
                                        ;       EXPECTING OUTPUT INTERRUPT;WHEN RX BACC OCCURS WHEN
                                        ;       EXPECTING TX,AND WHEN TX INT. OCCURS WHEN EXPECTING
                                        ;       RECIEVE.
                                        ;
                                        ;INPUTS:
                                        ;       ''DVTXA'' = ADDRESS OF TRANSMIT MSG.
                                        ;       ''DVTCC'' = BYTE COUNT OF TRANSMIT MSG.
                                        ;    ''QTX'' BIT  = SET IF TRANSMIT REQUESTED
                                        ;    ''ETX'' BIT  = SET IF TRNASMIT EXPECTED
                                        ;    ''ERX'' BIT  = SET IF RECIEVE EXPECTED
                                        ;
                                        ;OUTPUTS:
                                        ;       ''DVTXA'' = ADDRESS OF TX MSG. COMPLETED
                                        ;       ''DVTCC'' = BYTE COUNT OF TX MSG. COMPLETED
                                        ;       ''QTX''   = SET IF TX COMPLETED
                                        ;       ''DVRXA'' = ADDRESS OF RX MSG. COMPLETED
                                        ;       ''DVRCC'' = BYTE COUNT OF RX MSG. COMPLETED
                                        ;       ''QRX''   = SET IF RX COMPLETED
                                        ;
                                        ;
                                        ; SUBORDINATE ROUTINES USED:
                                        ;
                                        ;               ''TOORIO'' - TIME OUT OR OUTPUT INTERRUPT OR INTPUT INTERRUPT
                                        ;               ''CLRAW''  - CLEAR RQI AND WAIT FOR RDI TO CLEAR
                                        ;               ''LGDVE''  - LOG DEVICE ERROR TO EVENT LOG
                                        ;               ''OUTHDL'' - OUTPUT INTERRUPT HANDLER CODE
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;       JSR     PC,DVTXRX
                                        ;--

034200  032737  000010  007232  DVTXRX: BIT     #QTX,FLAG               ;ANY TX TO QUE
034206  001423                          BEQ     DVTR3                   ;IF NOT GO WAIT FOR OUPUT
034210  042737  000010  007232          BIC     #QTX,FLAG               ;CLEAR FLAG
034216  112777  000140  155552          MOVB    #140,@BSEL0
034224  004737  035026                  JSR     PC,TOORIO               ;GO CHECK FOR IN OR OUT
034230  017737  155552  010206          MOV     @SEL4,MODS              ;PUT IN NEW MOD STAT
034236  013777  007112  155542          MOV     DVTXA,@SEL4
034244  013777  007114  155540          MOV     DVTCC,@SEL6
034252  004737  034720                  JSR     PC,CLRAW                ;CLEAR RQUI ANDWAIT
034256                          DVTR3:
034256  012737  000074  007276          MOV     #60.,TIMERS             ;SET TIMER FOR 60 SECS
034264  032737  000060  007232  TOINOT: BIT     #CRX+#CTX,FLAG          ;IS IT TX OR RX COMP ALREADY?
```

```
034272  001071                            BNE      DVTR4             ;IS SO EXIT

034274  005737  007276                    TST      TIMERS            ;IS TIMER EXPIRED
034300  001022                            BNE      TOIN1
034302  012737  016424  007176            MOV      #DVEM4,TEMP2
034310  017737  155462  007200            MOV      @SEL0,TEMP3
034316  017737  155460  007202            MOV      @SEL2,TEMP4
034324  004737  020774                    JSR      PC,LGDVE
034330  005237  007144                    INC      ERRCNT
034334                                    ERRSOFT  12,DVEM4,ERR13
034334  104457                                                                          TRAP    C$ERSOFT
034336  000014                                                                          .WORD   12
034340  016424                                                                          .WORD   DVEM4
034342  020450                                                                          .WORD   ERR13
034344  000744                            BR       DVTR3             ;RETURN TO CHECK TIMER


034346                           TOIN1:   BREAK
034346  104422                                                                          TRAP    C$BRK
034350  032737  000001  007232            BIT      #ININT,FLAG       ;IS IT INPUT INTERRUPT
034356  001425                            BEQ      TOIN2             ;IF SO LOG ERROR

034360  012737  016516  007176            MOV      #DVEM5,TEMP2
034366  017737  155404  007200            MOV      @SEL0,TEMP3
034374  017737  155402  007202            MOV      @SEL2,TEMP4
034402  004737  020774                    JSR      PC,LGDVE
034406  042737  000001  007232            BIC      #ININT,FLAG       ;CLEAR BIT
034414  005237  007144                    INC      ERRCNT
034420                                    ERRSOFT  13,DVEM5,ERR13
034420  104457                                                                          TRAP    C$ERSOFT
034422  000015                                                                          .WORD   13
034424  016516                                                                          .WORD   DVEM5
034426  020456                                                                          .WORD   ERR13
034430  000715                            BR       TOINOT

034432  032737  000002  007232   TOIN2:   BIT      #OTINT,FLAG
034440  001711                            BEQ      TOINOT            ;IF NOT OUTPUT GO BACK AND
                                                                     ;CHECK TIMER AGAIN
034442  004737  035150                    JSR      PC,OUTHDL         ;ELSE HANDLE OUTPUT AND RETURN
034446  032737  000060  007232            BIT      #CTX+#CRX,FLAG    ;IS IT TX OR RX
034454  001703                            BEQ      TOINOT            ;IF NOT GO BACK AND TRY AGAIN
034456  032737  000020  007232   DVTR4:   BIT      #CTX,FLAG         ;IS IT TX
034464  001440                            BEQ      DVTR5             ;IF NOT TRY RX
034466  032737  000200  007232            BIT      #ETX,FLAG         ;IF SO SHOULD IT BE
034474  001020                            BNE      DVTR4A            ;IF IT SHOULD GO TO 4A
034476  012737  017041  007176            MOV      #DVEM9,TEMP2      ;ELSE LOG ERROR
034504  013737  035746  007200            MOV      TSEL4,TEMP3
034512  013737  035750  007202            MOV      TSEL6,TEMP4
034520  004737  020774                    JSR      PC,LGDVE          ;
034524                                    ERRSOFT  14,DVEM9,ERR13    ;REPORT ERROR
034524  104457                                                                          TRAP    C$ERSOFT
034526  000016                                                                          .WORD   14
034530  017041                                                                          .WORD   DVEM9
034532  020450                                                                          .WORD   ERR13

034534  000411                            BR       DVTR4B            ;THEN CLEAR COMPL.FLAG
```

```
034536 013737 035746 007112 DVTR4A: MOV      TSEL4,DVTXA
034544 013737 035750 007114         MOV      TSEL6,DVTCC
034552 052737 000010 007232         BIS      #QTX,FLAG          ;AND SET TX COMPL FLAG
034560 042737 000020 007232 DVTR4B: BIC      #CTX,FLAG          ;ELSE CLEAR FLAG
034566 032737 000040 007232 DVTR5:  BIT      #CRX,FLAG          ;IS IT RX TOO?
034574 001440                       BEQ      DVTREX             ;IF NOT THEN EXIT.
034576 032737 000100 007232         BIT      #ERX,FLAG          ;TEST IS THIS SUPPOSED TO BE RX
034604 001020                       BNE      DVTR5A             ;IF YES PROCESS AS SUCH
034606 012737 016752 007176         MOV      #DVEM8,TEMP2
034614 013737 035752 007200         MOV      RSEL4,TEMP3
034622 013737 035754 007202         MOV      RSEL6,TEMP4        ;ELSE
034630 004737 020774                JSR      PC,LGDVE           ;LOG ERROR
034634                              ERRSOFT  15,DVEM8,ERR13
034634 104457                                                                       TRAP    C$ERSOFT
034636 000017                                                                       .WORD   15
034640 016752                                                                       .WORD   DVEM8
034642 020450                                                                       .WORD   ERR13

034644 000411                       BR       DVTRX1             ;AND EXIT
034646 013737 035752 007126 DVTR5A: MOV      RSEL4,DVRXA
034654 013737 035754 007130         MOV      RSEL6,DVRCC
034662 052737 000004 007232         BIS      #QRX,FLAG
034670 042737 000040 007232 DVTRX1: BIC      #CRX,FLAG          ;CLEAR FLAG FOR RX DONE
034676 000207               DVTREX: RTS      PC                 ;AND EXIT
```

```
                                    ; DEVICE DEPENDENT SUBROUTINES


                                    .SBTTL              DEVICE INTERRUPT SERVICE ROUTINES


034700                                       BGNSRV  DVINS
034700                                                                              DVINS::
034700  052737  000001  007232               BIS     #ININT,FLAG
034706                                       ENDSRV
034706                                                                              L10021:
034706  000002                                                                               RTI

034710                                       BGNSRV  DVOUTS
034710                                                                              DVOUTS::
034710  052737  000002  007232               BIS     #OTINT,FLAG
034716                                       ENDSRV
034716                                                                              L10022:
034716  000002                                                                               RTI


                                    ;++
                                    ; FUNCTIONAL DESCRIPTION:
                                    ;        CLRAW - CLEAR RQI AND WAIT FOR RDI TO GO AWAY
                                    ;        THIS CODE CLEARS THE INPUT REQEST BIT(RQI) SETS A
                                    ;        TIMER UP TO TIME 50(OCTAL) TICKS AND MAKES SURE
                                    ;        RDI CLEARS BEFORE TIMER EXPIRES. IF TIMER EXPIRERS
                                    ;        CODE REPORTS ERROR AND SETS UP TIMER AND WAITS AGAIN.
                                    ;
                                    ;
                                    ;
                                    ; SUBORDINATE ROUTINES USED:
                                    ;
                                    ;                "LGDVE" - LOG DEVICE ERROR (TIME OUT)
                                    ;
                                    ;
                                    ; CALLING SEQUENCE:
                                    ;        JSR     PC,CLRAW
                                    ;--


034720  011637  007214              CLRAW:   MOV     (SP),PCADD          ;SAVE PC OF CALLING ROUTINE
034724  042777  000040  155044               BIC     #RQI,@SEL0
034732  012737  000050  007272      CLRA3:   MOV     #50,TIMER1          ;SET UP TIMER FOR 50(OCTAL) TICKS
034740  005737  007272              CLRA1:   TST     TIMER1
034744  001406                               BEQ     CLRA2               ;IF TIMER EXPIRED ERROR
034746                                       BREAK
034746  104422                                                                               TRAP    C$BRK
034750  032777  000200  155020               BIT     #RDI,@SEL0          ;IS RDI CLEAR
034756  001370                               BNE     CLRA1               ;IF NOT GO CHECK TIMER
                                                                         ; ELSE
034760  000207                               RTS     PC                  ;RETURN TO CALLER
034762  012737  016166  007176      CLRA2:   MOV     #DVEM0,TEMP2
034770  017737  155002  007200               MOV     @SEL0,TEMP3
034776  017737  155000  007202               MOV     @SEL2,TEMP4
035004  004737  020774                       JSR     PC,LGDVE            ;LOG DEVEICE EVENT 0
035010  005237  007144                       INC     ERRCNT
```

```
6035 035014                          ERRSOFT 16,DVEMO,ERR9    ;WHILE WAITING FOR RDI
6036 035014  104457                                                                  TRAP    C$ERSOFT
6037 035016  000020                                                                  .WORD   16
6038 035020  016166                                                                  .WORD   DVEMO
6039 035022  020372                                                                  .WORD   ERR9
6040 035024  000742                  BR       CLRA3           ;RESET TIMER AND CONTINUE
```

```
                                        .SBTTL                  TIME OUT OR INPUT INT. OR OUTPUT INT.



                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       TOORIO - TIME OUT OR INPUT INTERRUPT OR OUTPUT INTERRUPT
                                        ;       THIS ROUTINE SETS UP A TIMER FOR 100 (OCTAL) TICKS
                                        ;       THEN CHECKS FOR TIME OUT,OR INPUT INTERRUPT,OR OUTPUT
                                        ;       INTERRUPT. IF TIME OUT OCCURS IT REPORTS ERROR AND
                                        ;       RESTARTS TIMER. IF INPUT INTERRUPT OCCURS RETURN TO CALLER
                                        ;       IF OUTPUT INTERRPUT OCCURS LOG IT AND CONTINUE WAITING FOR
                                        ;       INPUT INTERRUPT.
                                        ;
                                        ; USE OF FLAGS:
                                        ;               ''OTINT'' - SET BY OUTPUT INT ROUTINE
                                        ;               ''ININT'' - SET BY INPUT INT. ROUTINE
                                        ;                           CLEARED BY THIS ROUTINTE.
                                        ;
                                        ; SUBORDINATE ROUTINES USED:
                                        ;
                                        ;               ''OUTHDL'' - OUTPUT INTERRUPT HANDLER
                                        ;
                                        ; CALLING SEQUENCE:
                                        ;       JSR     PC,TOORIO
                                        ;--

 035026  011637  007214                 TOORIO: MOV     (SP),PCADD              ;SAVE ADDR. OF CALLING ROUTINE
 035032  012737  000100  007272                 MOV     #100,TIMER1             ;SET UP TIMER
 035040  005737  007272                 TOOR3:  TST     TIMER1                  ;IS TIME EXPIRED
 035044  001022                                 BNE     TOOR1                   ;IF NOT CONTINUE
                                                                                ;IF YES ERROR
 035046  012737  016254  007176                 MOV     #DVEM1,TEMP2
 035054  017737  154722  007202                 MOV     @SEL2,TEMP4
 035062  017737  154710  007200                 MOV     @SEL0,TEMP3
 035070  004737  020774                         JSR     PC,LGDVE
 035074  005237  007144                         INC     ERRCNT
 035100                                         ERRSOFT 17,DVEM1,ERR9
 035100  104457                                                                                         TRAP    C$ERSOFT
 035102  000021                                                                                         .WORD   17
 035104  016254                                                                                         .WORD   DVEM1
 035106  020372                                                                                         .WORD   ERR9
 035110  000746                                 BR      TOORIO

 035112                                 TOOR1:  BREAK
 035112  104422                                                                                         TRAP    C$BRK
 035114  032737  000002  007232                 BIT     #OTINT,FLAG             ;IS THERE AN OUTPUT
                                                                                ;PENDING
 035122  001402                                 BEQ     TOOR2                   ;IF NOT GO TO 2
                                                                                ;ELSE GO HANDL IT
 035124  004737  035150                         JSR     PC,OUTHDL               ;
 035130  032737  000001  007232         TOOR2:  BIT     #ININT,FLAG             ;IS THERE AN INPUT PENDING
 035136  001740                                 BEQ     TOOR3                   ;IF NOT GO BACK TO TIMER CK.
 035140  042737  000001  007232                 BIC     #ININT,FLAG             ;ELSE CLEAR THE INPUT PEND FLAG
 035146  000207                                 RTS     PC                      ;AND RETURN TO CALLER

```

```
                                        .SBTTL          OUTPUT INTERRUPT HANDLER

                                        ;++
                                        ; FUNCTIONAL DESCRIPTION:
                                        ;       OUTHDL - OUTPUT INTERRUPT HANDLER
                                        ;       THIS ROUTINE IS CALLED WHEN AN OUTPUT INTERRUPT HAS SET
                                        ;       THE ''OTINT'' BIT IN THE ''FLAG'' WORD. IT CHECKS FOR
                                        ;       AN RDO SIGNAL IF NO RDO THEN REPORT ILLEGAL INTERRUPT.
                                        ;       THEN IT CHECKS FOR BACC OUT IF NOT BACC OUT REPORT THE
                                        ;       TYPE OF OUTPUT ERROR. IF BACC OUT FIND IF RX OR TX
                                        ;       IF RX SET CRX BIT AND MOVE ADDR AND BYTE COUNT TO RSEL4
                                        ;       AND RSEL6. IF TX SET CTX BIT AND MOVE ADDR AND BYTE COUNT
                                        ;       TO TSEL4 AND TSEL6. CLEAR OTINT FLAG AND RETURN TO CALLER.
                                        ;
                                        ; USE OF FLAGS:
                                        ;               ''OTINT'' - SET BY OUPUT ROUTINE
                                        ;                           CLEARED BY THIS ROUTINE
                                        ;               ''DMRRUN'' - SET BY DVINIT ROUTINE IF THIS IS DMR
                                        ;                            CHECKED AND CLEARED BY THIS ROUTINE.
                                        ;               ''CTX''    - SET IF TRANSMIT COMLETED
                                        ;               ''CRX''    - SET IF RECIEVE COMPLETED
                                        ;
                                        ;
                                        ; SUBORDINATE ROUTINES USED:
                                        ;
                                        ;               ''LGDVE'' -LOG DEVICE ERRORS TO EVENT LOG
                                        ;
                                        ; CALLING SEQUENCE
                                        ;               JSR     PC,OUTHDL
                                        ;
                                        ;--

035150  011637  007214                  OUTHDL: MOV     (SP),PCADD      ;SAVE ADDR. OF CALLING ROUTINE
035154  042737  000002  007232                  BIC     #OTINT,FLAG
035162  032777  000200  154612                  BIT     #RDO,@SEL2      ;CLEAR PEND FLAG AND CHK FOR RDO
035170  001023                                  BNE     OUTH1           ;IF RDO OK ...ELSE LOG ERROR
035172  012737  016610  007176                  MOV     #DVEM6,TEMP2
035200  017737  154576  007200                  MOV     @SEL2,TEMP3
035206  017737  154600  007202                  MOV     @SEL6,TEMP4
035214  004737  020774                          JSR     PC,LGDVE        ;GO LOG ERROR
035220  005237  007144                          INC     ERRCNT
035224                                          ERRSOFT 18,DVEM6,ERR9
035224  104457                                                                                  TRAP    C$ERSOFT
035226  000022                                                                                  .WORD   18
035230  016610                                                                                  .WORD   DVEM6
035232  020372                                                                                  .WORD   ERR9

                                                ;EXIT TEST IF ERROR

035234                                          ESCAPE  TST
035234  104410                                                                                  TRAP    C$ESCAPE
035236  000522                                                                                  .WORD   L10020-.

035240  032777  000001  154534  OUTH1:  BIT     #BACC,@SEL2     ;IS THE OUTPUT BACC
035246  001002                                  BNE     1$              ; BR IF NO
035250  000137  035660                          JMP     OUTH2           ;IF SO GO TO 2
```

```
                                                                        ;ELSE LOG ERROR AND PRINT IT
035254  017737  154532  007202  1$:     MOV     @SEL6,TEMP4
                                        ; IF NO BUFFER OUTPUT JUST COUNT THEM

035262  032737  000004  007202          BIT     #BIT2,TEMP4
035270  001404                          BEQ     OUTH6                   ;IF NO BUFF INC COUNT AND EXIT
                                                                        ;ELSE GO TO 6
035272  005237  007140                  INC     NOBUF
035276  000137  035736                  JMP     OUTHEX

035302  023727  012024  000006  OUTH6:  CMP     OPTYP,#6
035310  002420                          BLT     51$                     ;IF NOT DMR MODE SKIP TO 51
035312  032737  000040  007202          BIT     #BIT5,TEMP4             ;IS IT RUN STATE
035320  001414                          BEQ     51$                     ;IF NOT BRANCH
035322  032737  001000  007232          BIT     #DMRRUN,FLAG            ;IS RUN EXPECTED
035330  001405                          BEQ     52$                     ;IF NOT BRANCH
035332  042737  001000  007232          BIC     #DMRRUN,FLAG            ;IF SO THEN CLEAR EXPECTED
035340  000137  035736                  JMP     OUTHEX                  ;AND EXIT
035344  012737  020000  007204  52$:    MOV     #RUNSBM,CONOTM
035352  012737  016665  007176  51$:    MOV     #DVEM7,TEMP2
035360  017737  154416  007200          MOV     @SEL2,TEMP3

035366  004737  020774                  JSR     PC,LGDVE
035372  012737  013456  007204          MOV     #LP0,CONOTM             ;LOAD 'NULL STRING' TO INIT CONOTM
035400  032737  000001  007202          BIT     #BIT0,TEMP4             ;IS THIS DATA CHECK
035406  001403                          BEQ     1$
035410  012737  017340  007204          MOV     #DATCKM,CONOTM
035416  032737  000002  007202  1$:     BIT     #BIT1,TEMP4             ;IS THIS TIMEOUT
035424  001403                          BEQ     2$
035426  012737  017327  007204          MOV     #TIMOM,CONOTM
035434  032737  000010  007202  2$:     BIT     #BIT3,TEMP4             ;IS THIS DDCMP MAINT RECVD
035442  001403                          BEQ     4$
035444  012737  017307  007204          MOV     #DDCMRM,CONOTM
035452  032737  000020  007202  4$:     BIT     #BIT4,TEMP4             ;IS THIS LOST DATA
035460  001403                          BEQ     5$
035462  012737  017275  007204          MOV     #LOSDAM,CONOTM
035470  032737  000100  007202  5$:     BIT     #BIT6,TEMP4             ;IS THIS DISCONNECT
035476  001403                          BEQ     6$
035500  012737  017262  007204          MOV     #DISCOM,CONOTM
035506  032737  000200  007202  6$:     BIT     #BIT7,TEMP4             ;IS THIS DDCMP START RECVD
035514  001403                          BEQ     7$
035516  012737  017242  007204          MOV     #DDCSRM,CONOTM
035524  032737  000400  007202  7$:     BIT     #BIT8,TEMP4             ;IS THIS NON-EXSISTENT MEMORY
035532  001403                          BEQ     8$
035534  012737  017224  007204          MOV     #NXMM,CONOTM
035542  032737  001000  007202  8$:     BIT     #BIT9,TEMP4             ;IS THIS PROCEDURE ERROR
035550  001403                          BEQ     9$
035552  012737  017204  007204          MOV     #PROEM,CONOTM
035560  023727  012024  000006  9$:     CMP     OPTYP,#6                ;IS THIS DMR MODE
035566  002416                          BLT     11$                     ;IF NOT BRANCH
035570  032737  002000  007202          BIT     #BIT10,TEMP4            ;IS THIS A RX IDLE
035576  001403                          BEQ     10$                     ;IF NOT BRANCH
035600  012737  020022  007204          MOV     #RXIDM,CONOTM           ;IF SO SET UP MESSAGE
035606  032737  004000  007202  10$:    BIT     #BIT11,TEMP4            ;IS THIS CTS FAILED
035614  001403                          BEQ     11$                     ;IF NOT BRANCH
035616  012737  020046  007204          MOV     #CTSFM,CONOTM           ;IF SO SET UP MESSAGE
```

```
035624  032737  010000  007202  11$:    BIT     #BIT12,TEMP4        ;IS THIS CD GLITCHED
035632  001403                          BEQ     12$                 ;BR IF NO
035634  012737  020032  007204          MOV     #CDGLM,CONOTM       ;IF SO SET UP MESSAGE

035642  005237  007144          12$:    INC     ERRCNT
035646                                  ERRSOFT 19,DVEM7,ERR8
035646  104457                                                                              TRAP    C$ERSOFT
035650  000023                                                                              .WORD   19
035652  016665                                                                              .WORD   DVEM7
035654  020310                                                                              .WORD   ERR8
035656  000427                          BR      OUTHEX              ;CLEAR RDO AND RETURN TO CALLER

035660                          OUTH2:
035660  032777  000004  154114          BIT     #RXBIT,@SEL2        ;IS THIS RX BACC OUT
035666  001012                          BNE     OUTH3               ;IF NOT THEN IT MUST BE TX.
035670  052737  000020  007232          BIS     #CTX,FLAG
035676  017737  154104  035746          MOV     @SEL4,TSEL4
035704  017737  154102  035750          MOV     @SEL6,TSEL6
035712  000411                          BR      OUTHEX

035714  052737  000040  007232  OUTH3:  BIS     #CRX,FLAG           ;SET RX COMPL
035722  017737  154060  035752  OUTH4:  MOV     @SEL4,RSEL4         ;THEN MOVE TO TEMP
035730  017737  154056  035754          MOV     @SEL6,RSEL6         ;AND SEL6 TO TEMP
035736  042777  000200  154036  OUTHEX: BIC     #RDO,@SEL2          ;CLEAR RDO
035744  000207                          RTS     PC                  ;RETURN TO CALLER
035746  000000                  TSEL4:  .WORD   0
035750  000000                  TSEL6:  .WORD   0
035752  000000                  RSEL4:  .WORD   0
035754  000000                  RSEL6:  .WORD   0

035756  000207                          RTS     PC
```

```
 6241
 6242                                   .EVEN
 6243
 6244  035760                           ENDTST
 6245  035760                                                                   L10020:
 6246  035760  104401                                                                   TRAP    C$ETST
 6247
```

```
6248
6249
```

```
                                .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.
                                ;--

035762                                  BGNHRD
035762  000025                                                                  .WORD L10023-L$HARD/2
035764                                                                 L$HARD::


                                .SBTTL          DEVICE INDEPENDENT SECTION

035764                                  GPRML   DPLX,0,1,YES
035764  000130                                                                  .WORD   T$CODE
035766  036036                                                                  .WORD   DPLX
035770  000001                                                                  .WORD   1




                                .SBTTL          DEVICE DEPENDENT SECTION

035772                                  GPRMA   CSRADR,2,0,160000,177776,YES
035772  001031                                                                  .WORD   T$CODE
035774  036067                                                                  .WORD   CSRADR
035776  160000                                                                  .WORD   T$LOLIM
036000  177776                                                                  .WORD   T$HILIM
036002                                  GPRMA   VECTOR,4,0,300,776,YES
036002  002031                                                                  .WORD   T$CODE
036004  036115                                                                  .WORD   VECTOR
036006  000300                                                                  .WORD   T$LOLIM
036010  000776                                                                  .WORD   T$HILIM
036012                                  GPRMD   PRIOR,6,0,340,4,7,YES
036012  003032                                                                  .WORD   T$CODE
036014  036150                                                                  .WORD   PRIOR
036016  000340                                                                  .WORD   340
036020  000004                                                                  .WORD   T$LOLIM
036022  000007                                                                  .WORD   T$HILIM
                                ;       GPRMD   DEVPRM,10,D,17,0,15.,YES

036024                                  GPRMD   OPTN,12,0,7,0,7,YES
036024  005032                                                                  .WORD   T$CODE
036026  036176                                                                  .WORD   OPTN
036030  000007                                                                  .WORD   7
036032  000000                                                                  .WORD   T$LOLIM
036034  000007                                                                  .WORD   T$HILIM
                                ;       GPRMD   BAUD,14,0,7,0,7,YES
                                ;       GPRMD   LININ,16,0,7,0,7,YES

```

```
     036036                                       ENDHRD

     036036                                                                          L10023: .EVEN

                                           .NLIST  BEX

                                           ;DEVICE INDEPENDENT QUESTIONS

     036036  052506  046114  042040  DPLX:    .ASCIZ  /FULL DUPLEX OPERATION : /

                                           ;DEVICE DEPENDENT QUESTION

     036067     104  053105  041511  CSRADR: .ASCIZ  /DEVICE CSR ADDRESS : /
     036115     111  052116  051105  VECTOR: .ASCIZ  /INTERRUPT VECTOR ADDRESS: /
     036150  047111  042524  051122  PRIOR:   .ASCIZ  /INTERRUPT PRIORITY : /
     036176  042504  044526  042503  OPTN:    .ASCIZ  /DEVICE OPTION TYPE : (0=DMC,5=DMR-DMC MODE ,7=DMR)/

                                     ;DEVPRM:         .ASCIZ  /DEVICE PARMETER WORD (ENABLE CRC,STRIP SYNC,..) : /
                                     ;BAUD:   .ASCII  /BAUD RATE TO USE ('0' FOR 2.4K, '1' FOR 4.8K;/
                                     ;        .ASCII  <15><12>/   '2' FOR 9.6K, '3' FOR 19.2K, '4' FOR 56K;/
                                     ;        .ASCIZ  <15><12>/   '5' FOR 250K, '6' FOR 500K, '7' FOR 1 MEG : /
                                     ;LININ:  .ASCIZ  /LINE INTERFACE (0=423,1=...)/

                                     .LIST    BEX
             036262                           .EVEN
```

```
                              ;.SBTTL SOFTWARE PARAMETER CODING SECTION

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

                              ;       BGNSFT



                              ;       ENDSFT



                              ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
                              ; TEMPORARY PATCH AREA - FOR DEBUG PURPOSES
                              ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

036262                        $PATCH:
036262  000030                        .BLKW   30


036342                                LASTAD
                                                                              .EVEN
036342  000000                                                                .WORD   0
036344  000000                                                                .WORD   0
036346                        L$LAST::
036346                                ENDMOD

        000001                .END
```

```
ACT    = 000003          1660#    4483    4912    4989    5217
ACTATV   030216          4692     4912#
ACTBCR   030060          4710     4884#
ACTCHK   030432          4672     4958#
ACTCLB   027402          4773     4787#
ACTCLP   030536          4706     4983#
ACTCLR   027046          4670     4721#
ACTCOP   027700          4680     4847#
ACTCRC   030446          4701     4964#
ACTCSE   027172          4675     4749#
ACTCST   027320          4676     4775#
ACTDLL   030264          4696     4926#
ACTDME   027626          4712     4828    4831#
ACTDMQ   027620          4713     4830#
ACTDMS   027576          4711     4825#
ACTDMX   027634          4832#
ACTECH   030342          4700     4942#
ACTEQO   030022          4684     4873#
ACTHLP   027066          4674     4727#
ACTLIS   030254          4695     4923#
ACTLLP   030546          4707     4985#
ACTLPX   030564          4980     4982    4984    4986    4989#
ACTLXX   030626          4956     4974    4977    4990    4999#
ACTMEX   030210          4866     4882    4904    4909#
ACTME1   030202          4893     4895    4897    4899    4901    4908#
ACTMOP   030516          4704     4979#
ACTMS0   030102          4685     4892#
ACTMS1   030110          4686     4894#
ACTMS2   030120          4687     4896#
ACTMS3   030130          4688     4898#
ACTMS4   030140          4689     4900#
ACTMS5   030150          4690     4902#
ACTMS6   030166          4691     4905#
ACTM2X   030312          4913     4921    4924    4927    4930    4934#
ACTNO    030332          4699     4939#
ACTNUF   027036          4709     4718#
ACTNUL   027044          4669     4719#
ACTNUM   027710          4681     4850#
ACTOPM   030002          4682     4868#
ACTPAS   030226          4693     4915#
ACTPRO   030454          4702     4967#
ACTPRT   027132          4714     4740#
ACTQFG   030460          4959     4962    4965    4969#
ACTREC   030246          4694     4920#
ACTRLP   030556          4708     4987#
ACTRPS   030506          4703     4976#
ACTRUN   027146          4673     4744#
ACTSHO   027056          4671     4724#
ACTSHW   027442          4760     4779    4798#   4817
ACTSTE   027642          4677     4837#
ACTSTS   030440          4683     4961#
ACTSTT   027652          4678     4840#
ACTSTX   027660          4838     4841#
ACTSZE   027670          4679     4844#
ACTTAL   030304          4698     4932#
ACTTLP   030526          4705     4981#
```

```
ACTTRA  030274                  4697    4929#
ADDCC   022600                  3718#   4598    4644
ADDC1   022674                  3720    3735#
ADR   = 000020 G                1633#
ALCK    031142                  2100    5127#
ALCK1   031250                  5182    5191#   5235
ALCK2   031314                  5192    5201#   5267
ALCK2A  031534                  5232    5237#
ALCK3   031560                  5204    5239    5242#
ALCK3A  031652                  5249    5255#
ALCK3B  031664                  5256    5258#
ALCK3C  031660                  5253    5257#   5264
ALCK4   031706                  5243    5262#
ALCK4A  031714                  5261    5265#
ALCK4B  031726                  5266    5268#
ALCK5   031200                  5181#   5257
ALLTR   031200                  5059    5082    5106    5180#
ASSEMB= 000010                  1403
ATVMOD= 000027                  1759#   2239
BACC  = 000001                  1803#   6149
BAD     007211                  2071#   3068    5343*
BADCHR= 000051                  1777#   2331
BASE    017370                  3018#   3451    3452    5708
BASM1   015010                  2746#   3662
BASM1A  014755                  2738#   3445
BASM2   015001                  2744#   3681
BASM3   014772                  2742#   3673
BASN1   021470                  3449    3455#
BIT0  = 000001 G                1606#   1680    1791    1803    6176
BIT00 = 000001 G                1595#   1606
BIT01 = 000002 G                1594#   1605
BIT02 = 000004 G                1593#   1604
BIT03 = 000010 G                1592#   1603
BIT04 = 000020 G                1591#   1602
BIT05 = 000040 G                1590#   1601
BIT06 = 000100 G                1589#   1600
BIT07 = 000200 G                1588#   1599
BIT08 = 000400 G                1587#   1598
BIT09 = 001000 G                1586#   1597
BIT1  = 000002 G                1605#   1681    6179
BIT10 = 002000 G                1585#   1807    6202
BIT11 = 004000 G                1584#   1805    6205
BIT12 = 010000 G                1583#   6208
BIT13 = 020000 G                1582#
BIT14 = 040000 G                1581#   1794    1804
BIT15 = 100000 G                1580#
BIT2  = 000004 G                1604#   1682    1789    1808    5727    5730    6156
BIT3  = 000010 G                1603#   1790    5727    5733    6182
BIT4  = 000020 G                1602#   1683    6185
BIT5  = 000040 G                1601#   1684    1792    1800    6164
BIT6  = 000100 G                1600#   1809    6188
BIT7  = 000200 G                1599#   1793    1801    1802    6191
BIT8  = 000400 G                1598#   1806    6194
BIT9  = 001000 G                1597#   1795    6197
BLDBEX  023006                  3779    3783#
BLDBUF  022676                  3761#   4474    4481    4599    4645    4794    5033
```

```
BLDB1    022706           3766#
BLDB2    022750           3774#    3781
BLDB3    022766           3777#    3782
BOE    = 000400 G         1637#
BSEL0    011776           2374#    5706*    5725*    5739*    5745*    5795*    5834*    5889*
BSEL1    012000           2375#    4269*    4270*    5676*    5679*
BSEL2    012002           2377#
BSEL3    012004           2378#    4273*    4274*
BSEL4    012006           2380#
BSEL5    012010           2381#    4277*    4278*
BSEL6    012012           2383#    5730*    5733*
BSEL7    012014           2384#    4281*    4282*
BUFEX    014033           2638#    4571     4612
BUFLIM=  001000           1648#    2020     2021     2022     3719     3731     4568     4609     4787     5029
BYTBIT   007152           2051#    3453*    3667     4826*    4830*
CABLE =  000002           1667#
CBLLOP=  000045           1773#    2352
CCURAD   007110           2031#    4470*    4478     4643     4648*    4772*
CDGLM    020032           3026#    6210
CHECK =  000003           1739#    2287
CLEAR =  000001           1737#    2225     4561     4721
CLIACT   026664           4528     4662#
CLIALN=  000007           1730#    2329
CLIALP=  000006           1729#    2273     2283     2334
CLIBCR   012344           2466#    4885
CLIBDL   012237           2454#    4994
CLIBIF=  000003           1726#
CLIBR =  000002           1725#    2222     2224     2226     2229     2231     2240     2242     2244     2246     2248     2251     2253
                          2268     2270     2280     2288     2295     2297     2301     2303     2310     2312     2314     2316     2318
                          2320     2322     2327     2330     2337     2339     2344     2351     2353     2355     2357     2362
CLIBRX   012217           2451#    3997
CLIDEC=  000011           1732#    2343     2361
CLIERM   012122           2439#    4534
CLIERR=  000000           1723#    2232     2331
CLIEXI=  000001           1724#    2216     2218     2220     2259     2261     2367
CLINBG   012175           2447#    4040
CLINPS   012301           2460#    4948
CLINUF   012152           2443#    4543
CLINUM=  000005           1728#
CLIOCT=  000010           1731#    2275     2277
CLISEO   012411           2473#    4855
CLISPA=  000004           1727#    2214     2236     2257     2266     2274     2328
CLISTR=  000012           1733#    2217     2219     2221     2223     2225     2228     2230     2237     2239     2241     2243     2245
                          2247     2250     2252     2258     2260     2267     2269     2285     2287     2294     2296     2300     2302
                          2309     2311     2313     2315     2317     2319     2321     2336     2338     2350     2352     2354     2356
CLITRE   010344           2210#    4527
CLISPM   012114           2438#    4521
CLKBR    007254           2108#    4217*
CLKCSR   007252           2107#    3267*    3288*    4188     4206*    4207*    4208*    4218*    4355*    4434*
CLKEN    007262           2111#    3288     4196*    4209*    4218     4434
CLKHZ    007260           2110#    3270     3285     3415     4222     4243
CLKINT   020570 G         3265#    4293
CLKSET   020544           3220#    4195     4205
CLKVEC   007256           2109#    4186*    4216*    4294
CLRAW    034720           5719     5734     5741     5758     5799     5839     5894     6019#
CLRA1    034740           6022#    6027
```

```
CLRA2    034762          6023     6030#
CLRA3    034732          6021#    6040
CMDBUF   003062          1967#    4519     4526
CMPBUF   005326          2022#    4470     4771
CMPPTR   007102          2028#    4464*    4465*    4477     4626*    4627*    4641*    4642     4646*    4753*    4754*    4755     4768*
                         4769*    4770     5009*    5010*    5011     5307
CMPSEX   032202          5305     5385#
CMPSR    031726          5254     5304#
CMPS1    032066          5336#    5354
CMPS2    032072          5339#
CMPS3    031760          5310#    5366
CMPS5    032170          5363#
CMPS5A   032174          5356     5364#
CMPS6    032136          5337     5341     5352#
CMPS7    032054          5325     5334#
CMPTOT   007104          2029#    4482*    4621*    4628     4647*    4757     4761*    5018
CMSG0  = 000020          1752#    2311
CMSG1  = 000021          1753#    2309
CMSG2  = 000022          1754#    2313
CMSG3  = 000023          1755#    2315
CMSG4  = 000024          1756#    2319
CMSG5  = 000025          1757#    2317
CMSG6  = 000026          1758#    2321
CONOTM   007204          2068#    3111     3168     5481*    5498*    5506*    5511*    6170*    6175*    6178*    6181*    6184*    6187*
                         6190*    6193*    6196*    6199*    6204*    6207*    6210*
CONTIN   007206          2069#    5757*    5798
CPTR     007162          2059#    3764     3768*    4473*    4477*    4596*    4600     4642*    4646     4770*    4783*    5031*    5058*
                         5078*    5103*    5128*    5193     5198*    5210     5258*    5265     5306*    5311     5315*    5612*
CPTRR    007160          2058#    5055*    5081*    5104*    5130*    5183     5188*    5216*    5221     5229*    5241*    5262     5307*
                         5317     5320*    5566*
CR       016042          2866#    3399
CRC    = 000040          1768#
CRCB   = 000020          1683#    4964
CRX    = 000040          1715#    5897     5937     5958     5976     6228
CSHEXP=  000006          1742#    2258
CSHTRN=  000007          1743#    2260
CSRADR   036067          6281     6310#
CTOTCC   007106          2030#    4457*    4608     4619     4649*    4762*
CTS    = 000004          1789#    2132
CTSFM    020046          3028#    6207
CTX    = 000020          1714#    5897     5937     5939     5957     6223
CURADD   007164          2060#    3766     3771     3773     3783*    4472*    4478*    4597*    4602     4643*    4648     4788*    5030*
                         5428*    5434*    5469     5470
CURCC    007156          2057#    3718     3730     3733*    3734     3767     3772     3783     4468*    4480*    4790*    4862*    4875*
                         4903*    4906*    4908*    5029*    5429*    5435*    5471     5472
C$AU   = 000052          1403#    4414
C$AUTO=  000061          1403#    4344
C$BRK  = 000022          1403#    5684     5916     6025     6085
C$BSEG=  000004          1403#
C$BSUB=  000002          1403#
C$CEFG=  000045          1403#
C$CLCK=  000062          1403#    4191     4201
C$CLEA=  000012          1403#    4371
C$CLOS=  000035          1403#
C$CLP1=  000006          1403#
C$CVEC=  000036          1403#
```

```
C$DCLN= 000044           1403#
C$DODU= 000051           1403#
C$DRPT= 000024           1403#
C$DU  = 000053           1403#   4392
C$EDIT= 000003           1403#   1465
C$ERDF= 000055           1403#
C$ERHR= 000056           1403#   5451
C$ERRO= 000060           1403#
C$ERSF= 000054           1403#
C$ERSO= 000057           1403#   5328    5346    5359    5694    5908    5927    5948    5967    6036    6078    6138    6214
C$ESCA= 000010           1403#   6146
C$ESEG= 000005           1403#
C$ESUB= 000003           1403#
C$ETST= 000001           1403#   6246
C$EXIT= 000032           1403#   4238    4320    4363    4507
C$GETB= 000026           1403#
C$GETW= 000027           1403#
C$GMAN= 000043           1403#   3441    4220    4517    5544
C$GPHR= 000042           1403#   4258
C$GPLO= 000030           1403#
C$GPRI= 000040           1403#
C$INIT= 000011           1403#   4328
C$INLP= 000020           1403#
C$MANI= 000050           1403#   4501
C$MEM = 000031           1403#
C$MSG = 000023           1403#   3079    3092    3106    3128    3149    3163    3178
C$OPEN= 000034           1403#
C$PNTB= 000014           1403#   3075    3088    3102    3117    3124    3138    3145    3159    3174
C$PNTF= 000017           1403#   3402    3410    3665    3676    3684    3728    4000    4043    4235    4447    4492    4537
                         4546    4575    4590    4616    4636    4733    4813    4858    4888    4951    4997    5445    5605
                         5628
C$PNTS= 000016           1403#   3462    3485    3499    3512    3525    3533    3557    3573    3588    3606    3626    3829
                         3854
C$PNTX= 000015           1403#
C$QIO = 000377           1403#
C$RDBU= 000007           1403#   4213
C$REFG= 000047           1403#   4164    4169    4174    4180
C$RESE= 000033           1403#   4360
C$REVI= 000003           1403#   1464
C$RFLA= 000021           1403#
C$RPT = 000025           1403#   4133
C$SEFG= 000046           1403#
C$SPRI= 000041           1403#   4318    4358
C$SVEC= 000037           1403#   4296    4306    4313
C$TPRI= 000013           1403#
DATCKB= 000002           1681#   3838    4958    5077    5213    5260    5304    5427    5538    5600
DATCKM  017340           3008#   6178
DCD   = 000001           1791#   2134
DCK   = 000014           1702#   3363
DDCMRM  017307           3003#   6184
DDCSRM  017242           2996#   6193
DDE   = 000022           1705#   3371
DER   = 000010           1700#   3349
DEV1    010334           2190#   3518*   3529    3537*   3539*   3810    4494*   4818*
DEV2    010336           2191#   3519*   3528    3538*   3540*   3813    4495*   4819*
DEV3    010340           2192#   3541*   3820    4496*   4820*
```

```
DEV4     010342              2193#    3542*    3834     3838     3842     4497*    4821*
DFPTBL   002130 G            1531#
DIAGMC=  000000              1403
DISCOM   017262              2999#    6190
DLE   =  000020              1704#    3367
DLL      032262              2101     5426#
DLLAB    017130              2981#    5453     5484
DLLCM    013640              2616#    5442
DLLEA    032370              5449#    5486
DLLE1    032646              5477     5496#
DLLE2    032514              5475#    5495
DLLE3    032726              5505     5509#
DLLE4    032716              5506#
DLLE5    032746              5494     5510     5514#
DLLE6    032600              5479     5488#
DLLE7    032546              5482#    5499     5507     5512
DLLE8    032666              5497     5500#
DLLGA =  000400              1718#    5436     5476     5492
DLLMOD=  000033              1763#    2247
DLLM1    002647              1831     1914#    5428
DLLM1C   002172              1831#    5429
DLLM1E   002654              1831     1919#
DLLM2    002654              1832     1920#    5434
DLLM2C   002174              1832#    5435     5493
DLLM2E   003062              1832     1961#
DLLPRI   032344              5440#
DLTXRX   032404              5430     5437     5460#
DMC   =  000000              1688#
DMPE  =  000053              1779#    2277
DMPQ  =  000054              1780#    2279
DMPS  =  000052              1778#    2275     4557     4827
DMRC6 =  000004              1689#    5674
DMRC7 =  000005              1690#
DMRRUN=  001000              1719#    5762     6166     6168
DMR6  =  000006              1691#    5677
DMR7  =  000007              1692#
DMSGAD   002176              1836#    3775
DMSGCT   002150              1821#    3774
DOW   =  000004              1661#    4926     5748
DPLX     036036              6271     6310#
DSR   =  000010              1790#    2133
DUMEX    022576              3687     3693#
DUMPSR   022442              3454     3658#    4559
DUM1     022534              3668     3679#
DUM2     022556              3678     3686#
DUM3     022472              3667#    3691
DUM4     022446              3659#    3690
DVEM0    016166              2890#    6030     6038
DVEM1    016254              2900#    6072     6080
DVEM3    016340              2910#    5688     5696
DVEM4    016424              2920#    5902     5910
DVEM5    016516              2931#    5920     5929
DVEM6    016610              2942#    6132     6140
DVEM7    016665              2950#    6171     6216
DVEM8    016752              2961#    5962     5969
DVEM9    017041              2971#    5943     5950
```

```
DVI    = 000012              1701#    3354
DVINEX   034062              5761     5764#
DVINIT   033400              5027     5666#    5699
DVINS    034700 G            4303     5986#
DVIN1    033542              5682     5701#
DVIN10   033736              5732     5734#
DVIN2    033460              5678     5681#    5686
DVIN3    033646              5717     5719#
DVIN4    034032              5751     5754     5757#
DVIN5    034016              5749     5753#
DVIN6    033442              5675     5677#
DVIN7    033614              5712     5714#
DVIN8    033760              5724     5745#
DVIN9    033720              5729     5731#
DVMODS   034064              3379     5795#
DVOUTS   034710 G            4310     5993#
DVRCC    007130              2041#    5187*    5206     5464*    5501     5609*    5620     5622*    5623*    5838     5974*
DVRCT    007132              2042#    5056*    5129*    5215*    5219*    5237*    5238     5308*    5365*
DVREX    034176              5831     5840#
DVRXA    007126              2040#    5185*    5205     5462*    5500     5607*    5619     5622     5837     5973*
DVRXQ    034120              5189     5466     5614     5829#
DVTCC    007114              2034#    5197*    5245     5471*    5489     5493     5563*    5571     5893     5955*
DVTCT    007116              2035#    5079*    5102*    5127*    5247*    5252     5255
DVTREX   034676              5959     5977#
DVTRX1   034670              5972     5976#
DVTR3    034256              5887     5895#    5912
DVTR4    034456              5898     5939#
DVTR4A   034536              5942     5954#
DVTR4B   034560              5953     5957#
DVTR5    034566              5940     5958#
DVTR5A   034646              5961     5973#
DVTXA    007112              2033#    5195*    5244     5469*    5488     5560*    5570     5892     5954*
DVTXRX   034200              5201     5475     5568     5617     5886#
ECHO   = 000037              1767#    2296
ECHOB  = 000004              1682#    3842     4934     4944     4954     5208
EDDCK    014573              2160     2711#
EDDDE    014706              2163     2725#    5348     5361
EDDER    014556              2158     2708#
EDDLE    014651              2162     2720#    5330
EDDVI    014623              2159     2716#
EDEOP    014741              2164     2730#
EDRXC    014530              2157     2704#
EDRXQ    014503              2156     2700#
EDTXC    014454              2155     2696#
EDTXQ    014430              2154     2692#
EFM11    014353              2680#    3099
EFM2     014256              2669#    3085
EF.CON=  000036 G            1613#    4173
EF.NEW=  000035 G            1614#    4179
EF.PWR=  000034 G            1615#
EF.RES=  000037 G            1612#    4168
EF.STA=  000040 G            1611#    4163
EMSG0    002221              1822     1848#
EMSG1    002222              1823     1850#
EMSG2    002223              1824     1852#
EMSG3    002224              1825     1854#
```

```
EMSG4   002324          1826    1868#
EMSG5   002416          1827    1882#
EMSG6   002517          1828    1896#
EMSG8   002647          1830    1910#
ENADD   007150          2050#   3452*   3686    4831*
ENDEVT  022320          3464    3474    3478    3593#
ENDIT   025146          4177    4290#
ENDQO = 000017          1751#   2326
EOP   = 000024          1706#   3375
ERRCNT  007144          2048#   3378    3381    5024*   5326*   5344*   5357*   5389    5692*   5906*   5925*   6034*   6076*
                        6136*   6212*
ERR1    020170 G        3065#   5349
ERR10   020260 G        3095#   5331
ERR13   020450 G        3152#   5697    5911    5930    5951    5970
ERR14   020502 G        3166#   5454
ERR2    020232 G        3082#   5362
ERR8    020310 G        3109#   6217
ERR9    020372 G        3131#   6039    6081    6141
ERX   = 000100          1716#   5057    5105    5131    5233    5251    5426    5611    5960
ETX   = 000200          1717#   5080    5131    5234    5250    5474    5491    5565    5577    5941
EVL   = 000004 G        1631#
EVMCTS  016004          2143    2848#
EVMDCD  016014          2145    2852#
EVMDSR  016010          2144    2850#
EVMOCG  015664          2828#
EVMOHD  015707          2836#   3603
EVMOST  015767          2845#   3623
EVMRI   016024          2147    2856#
EVMRTS  016020          2146    2854#
EVMSQD  016030          2148    2858#
EVMTM   016034          2149    2860#
EVTADD  010300          2171#   3503*   3508    3547*   3553    3563*   3569    3578*   3584
EVTBCT  010302          2172#   3504*   3507    3548*   3552    3564*   3568    3579*   3583
EVTEND  010204          2124#   3423    3471    4247
EVTF0   015051          2756#   3482
EVTF1   015147          2767#   3496
EVTF2   015176          2771#   3509
EVTF3   015250          2778#   3522
EVTF3C  015262          2780#   3135    3156    3530
EVTF3D  015277          2783#   3114    3171
EVTF4   015321          2787#   3570
EVTF4A  015423          2799#   3585
EVTF4B  015521          2810#   3554
EVTF5A  015600          2818#   3072
EVTLOG  007302          2122    2123#   3427    3456    3469    4244
EVTLST  010244          2154#   3492
EVTMIN  010274          2169#   3490*   3495
EVTPTR  007300          2122#   3413    3428*   3455    3476    4245*
EVTSEC  010272          2168#   3489*   3494
EVTTCK  010276          2170#   3488*   3493
EVTTMP  010304          2173#   3517*   3521    3549*   3551    3565*   3567    3580*   3582
E$END = 002100          1403#
E$LOAD= 000035          1403#   1488
FHDPLX  007224          2084#   4263*   5753
FLAG    007232          2094#   5021*   5057*   5080*   5105*   5131*   5181    5191    5203    5233*   5234*   5240*   5242
                        5250*   5251*   5259*   5426*   5436*   5461*   5474*   5476    5478    5491*   5492*   5496    5565*
```

```
                          5577*   5611*   5705*   5762*   5830    5833*   5886    5888*   5897    5917    5924*   5933    5937
                          5939    5941    5956*   5957*   5958    5960    5975*   5976*   5987*   5994*   6086    6091    6093*
                          6129*   6166    6168*   6223*   6228*
F$AU   = 000015           1403#   4402    4413
F$AUTO= 000020            1403#   4339    4343
F$BGN  = 000040           1403#   1405    3065    3082    3095    3109    3131    3152    3166    3183    3265    4122    4142
                          4158    4238    4320    4339    4353    4363    4380    4386    4402    4408    4429    4507    5986
                          5993    6146    6245    6263    6347
F$CLEA= 000007            1403#   4353    4370
F$DU   = 000016           1403#   4380    4391
F$END  = 000041           1403#   1405    3080    3093    3107    3129    3150    3164    3179    3181    3292    4134    4238
                          4320    4329    4345    4363    4372    4384    4393    4406    4415    4429    4507    5991    5998
                          6146    6245    6247    6309    6347
F$HARD= 000004            1403#   6263    6307
F$HW   = 000013           1403#   1529    1560
F$INIT= 000006            1403#   4158    4327
F$JMP  = 000050           1403#   3181    4238    4320    4363    4384    4406    4507
F$MOD  = 000000           1403#   1405    6347
F$MSG  = 000011           1403#   3065    3078    3082    3091    3095    3105    3109    3127    3131    3148    3152    3162
                          3166    3177
F$PROT= 000021            1403#   4142    4149
F$PWR  = 000017           1403#
F$RPT  = 000012           1403#   4122    4132
F$SEG  = 000003           1403#
F$SOFT= 000005            1403#
F$SRV  = 000010           1403#   3265    3290    5986    5989    5993    5996
F$SUB  = 000002           1403#
F$SW   = 000014           1403#
F$TEST= 000001            1403#   4430    6245
GETCL    025724           4503    4514#   4539    4548    4551    4553    4560    4562    4564    4577    4592    4606    4618
                          4638    4652
GETPRM   024726           4183    4252#   4261
GOOD     007210           2070#   3070    5342*
GTRA2    025332           4437#
GTRA3    025374           4440    4450#
GTRA4    025402           4455#
GTRA5    025640           4451    4494#   5397    5447    5456
GTREX    030716           5021#
GTRX2    030746           5029#   5396    5579    5635
GTR9     030634           4510    4556    5004#
GTXRXB   025332           4436#
G$CNTO= 000200            1403#
G$DELM= 000372            1403#
G$DISP= 000003            1403#
G$EXCP= 000400            1403#
G$HILI= 000002            1403#
G$LOLI= 000001            1403#
G$NO   = 000000           1403#   4520    5547
G$OFFS= 000400            1403#   3444    4223    4520    5547    6270    6280    6285    6290    6298
G$OFSI= 000376            1403#   3444    4223    4520    5547    6270    6280    6285    6290    6298
G$PRMA= 000001            1403#   6280    6285
G$PRMD= 000002            1403#   4223    4520    5547    6290    6298
G$PRML= 000000            1403#   3444    6270
G$RADA= 000140            1403#   4520    5547
G$RADB= 000000            1403#
G$RADD= 000040            1403#   4223
```

```
GSRADL= 000120          1403#   3444    6270
GSRADO= 000020          1403#   6280    6285    6290    6298
GSXFER= 000004          1403#
GSYES = 000010          1403#   3444    4223    6270    6280    6285    6290    6298
HALFDB= 002000          1807#   5750    5755
HELP  = 000000             1#   1403    1417    1507    1519    1533    1564    1568    1575    2394    2415    2432    3046
                        3048    3063    3192    4124    4128    4130    4150    4160    4323    4341    4355    4366    4382
                        4387    4404    4409    4426    4427    4431    6241    6248    6266    6313    6327    6332    6340
HELPDC= 000000             1#      3    1506    1542    1563    1783    1797    2368    2395    2416    2887    3108    4265
                        4301    5640    5645    5772    5804    5844    5981    5984    5992    6275    6310
HLP   = 000005          1741#   2215    2217    4550    4737
HLPEND  003230          1979#   4735
HLPF    012520          2486#   4730
HLPTAB  003212          1972#   4727
HLPO    012442          2478#   4489
HLP1    012525          1972    2487#
HLP2    012540          1973    2489#
HLP3    012646          1974    2502#
HLP4    012733          1975    2511#
HLP4A   013012          1976    2519#
HLP5    013070          1977    2527#
HLP6    013152          1978    2536#
HOE   = 100000 G        1644#
IBE   = 010000 G        1641#
IDU   = 000040 G        1634#
IEO   = 000100          1809#   5714
IER   = 020000 G        1642#
ININT = 000001          1710#   5917    5924    5987    6091    6093
INTPRI  012022          2389#   4287*   4302    4309
INVEC   012016          2387#   4284*   4304
ISR   = 000100 G        1635#
IXE   = 004000 G        1640#
I$AU  = 000041          1403#   4402#   4415#
I$AUTO= 000041          1403#   4339#   4345#
I$CLN = 000041          1403#   4353#   4363    4372#
I$DU  = 000041          1403#   4380#   4393#
I$HRD = 000041          6263#   6309#
I$INIT= 000041          1403#   4158#   4238    4320    4329#
I$MOD = 000041          1403#   1405#   6347#
I$MSG = 000041          1403#   3065#   3080#   3082#   3093#   3095#   3107#   3109#   3129#   3131#   3150#   3152#   3164#
                        3166#   3179#
I$PROT= 000040          1403#   4142#
I$PTAB= 000041          1403#
I$PWR = 000041          1403#
I$RPT = 000041          1403#   4122#   4134#
I$SEG = 000041          1403#   4429
I$SETU= 000041          1403#
I$SRV = 000041          1403#   3265#   3292#   5986#   5991#   5993#   5998#
I$SUB = 000041          1403#   4429
I$TST = 000041          1403#   4429#   4507    6146    6245#   6247#
J$JMP = 000167          1403#   3181    4384    4406
KEYWD1  003204          1968#   4550    4552    4554    4557    4561    4563    4565    4721*   4724*   4737*   4740*   4744*
                        4759    4778    4827*   4837*   4840*
LCLKEN= 000100          1674#   4196
LGDVE   020774          3347#   5485    5691    5905    5923    5946    5965    6033    6075    6135    6174
LIS   = 000006          1663#   4923    5578
```

```
LISCK   033170          2103    5600#
LISCKA  033216          5607#   5631    5633
LISMOD= 000032          1762#   2245
LISP    013564          2600#   5602
LMDLOP= 000046          1774#   2354
LNCNT   007136          2045#   3394    3396*   3405*
LOE   = 040000 G        1643#
LOGCMD  021114          3369#   5363
LOGCML  021076          3365#   5332
LOGCMP  021060          3361#   5322
LOGDVI  021012          3352#   5026
LOGEOP  021132          3373#   5390
LOGEX   021404          3383    3430#
LOGRXC  020764          3344#   5207    5502    5621
LOGRXQ  020746          3339#   5190    5467    5615
LOGS1   021150          3332    3337    3342    3346    3378#
LOGS2   021376          3424    3428#
LOGS3   021202          3350    3359    3364    3368    3372    3376    3387#
LOGS4   021256          3395    3404#
LOGS5   021302          3389    3391    3412#
LOGTXC  020730          3334#   5246    5490    5572
LOGTXQ  020712          3329#   5199    5473    5564
LOGUNT  007212          2075#   4250*   4252*   4253    4257
LOOPS   003276          2001#   3819
LOSDAM  017275          3001#   6187
LOT   = 000010 G        1632#
LP0     013456          2001    2161    2579#   3818    6175
LP00    013457          2580#   3815
LP1     013466          2002    2582#
LP2     013477          2003    2584#
LP3     013505          2004    2586#
LP4     013520          2005    2588#
LULOOP= 004000          1805#   5715    5718
L$ACP   002110 G        1495#
L$APT   002036 G        1453#
L$AU    025316 G        1480    4402#
L$AUT   002070 G        1479#
L$AUTO  025264 G        1496    4339#
L$CCP   002106 G        1493#
L$CLEA  025266 G        1494    4353#
L$CO    002032 G        1449#
L$DEPO  002011 G        1431#
L$DESC  012042 G        1486    2421#
L$DESP  002076 G        1485#
L$DEVP  002060 G        1471#
L$DISP  002124 G        1456    1516#
L$DLY   002116 G        1501#
L$DTP   002040 G        1455#
L$DTYP  002034 G        1451#
L$DU    025310 G        1482    4380#
L$DUT   002072 G        1481#
L$DVTY  012026 G        1472    2410#
L$EF    002052 G        1466#
L$ENVI  002044 G        1459#
L$ETP   002102 G        1489#
L$EXP1  002046 G        1461#
```

```
L$EXP4   002064 G          1475#
L$EXP5   002066 G          1477#
L$HARD   035764 G          1438     6263     6264#
L$HIME   002120 G          1503#
L$HPCP   002016 G          1437#
L$HPTP   002022 G          1441#
L$HW     002130 G          1442     1529     1530#
L$ICP    002104 G          1491#
L$INIT   024400 G          1492     4158#
L$LADP   002026 G          1445#
L$LAST   036346 G          1446     6345#
L$LOAD   002100 G          1487#
L$LUN    002074 G          1483#
L$MREV   002050 G          1463#
L$NAME   002000 G          1420#
L$PRIO   002042 G          1457#
L$PROT   024372 G          1498     4142#
L$PRT    002112 G          1497#
L$REPP   002062 G          1473#
L$REV    002010 G          1429#
L$RPT    024364 G          1474     4122#
L$SPC    002056 G          1469#
L$SPCP   002020 G          1439#
L$SPTP   002024 G          1443#
L$STA    002030 G          1447#
L$TEST   002114 G          1499#
L$TIML   002014 G          1435#
L$UNIT   002012 G          1433#    4253
L10000   002150            1529     1560#
L10001   020230            3078#
L10002   020256            3091#
L10003   020306            3105#
L10004   020370            3127#
L10005   020446            3148#
L10006   020500            3162#
L10007   020536            3177#    3182
L10010   020710            3290#
L10011   024370            4132#
L10013   025262            4239     4321     4327#
L10014   025264            4343#
L10015   025306            4364     4370#
L10016   025314            4385     4391#
L10017   025322            4407     4413#
L10020   035760            4508     6147     6245#
L10021   034706            5989#
L10022   034716            5996#
L10023   036036            6263     6308#
L5060    013602            2603#    4224
MAINTB=  000400            1806#    5750
MCLR   = 040000            1804#    5672
MLTYP    007222            2081#    3356     4484*    4495     4509*    4819     4917*    4936*    4979*    4981*    4983*    4985*    4987*
                           4992*    5716     5728     5731
MOBITE   010226            2139#    3620
MOBITS   010210            2132#    3608
MODE     007234            2097#    5036
MODES    003260            1993#    3812
```

```
MODLOC= 000003           1668#   5731
MODREM= 000004           1669#   5728
MODS    010206           2128#   3385    5797*   5836*   5891*
MODTYP  007220           2078#   3355    4483*   4494    4818    4912*   4915*   4920*   4923*   4926*   4929*   4932*   4945
                         4989    5034    5217    5231    5248    5578*   5634*   5748
MOMSGS  010226           2143#   3609
MOP   = 000043           1670#   1771#
MO0     013365           1993    2565#
MO1     013375           1994    2567#
MO2     013406           1995    2569#
MO3     013416           1996    2571#
MO4     013425           1997    2573#
MO5     013442           1998    2576#
MO6     013447           1999    2577#
MSG     002736           1934    1942#
MSGLIM= 000017           1650#   4461    4583    4623    4629    4750    4765    5006    5012
MSGTRN  014051           2641#   4572    4587    4613    4633
MSGTRU  014102           2646#   3725
MSGTYP  007154           2055#   3769    4467*   4479*   4789*   4868*   4892*   4894*   4896*   4898*   4900*   4902*   4905*
                         5032*
MSG0    002220           1822    1837    1847#
MSG0C   002150           1822#
MSG1    002221           1823    1838    1849#
MSG1C   002152           1823#
MSG2    002222           1824    1839    1851#
MSG2C   002154           1824#
MSG3    002223           1825    1840    1853#
MSG3C   002156           1825#
MSG4    002224           1826    1841    1855#
MSG4C   002160           1826#
MSG5    002324           1827    1842    1869#
MSG5C   002162           1827#   4468    4480    4790    4903
MSG6    002416           1828    1843    1883#
MSG6C   002164           1828#   4906
MSG8    002646           1830    1845    1909#
MSG8C   002170           1830#
NEW     024720           4182    4250#   4254
NO    = 000036           1766#   2285    4939    4942    4970
NOBUF   007140           2046#   5022*   5387    6159*
NOCLK   013712           2623#   4232    4444
NOD0    010344           2214#
NOD1    010350           2215#
NOD10   010424           2222#
NOD100  011262           2297#
NOD101  011266           2300#
NOD102  011302           2301#
NOD103  011306           2302#
NOD104  011322           2303#
NOD105  011326           2306#
NOD106  011332           2309#
NOD107  011346           2310#
NOD11   010430           2223#
NOD110  011352           2311#
NOD111  011370           2312#
NOD112  011374           2313#
NOD113  011410           2314#
```

```
NOD114  011414          2315#
NOD115  011430          2316#
NOD116  011434          2317#
NOD117  011450          2318#
NOD12   010444          2224#
NOD120  011454          2319#
NOD121  011470          2320#
NOD122  011474          2321#
NOD123  011510          2322#
NOD124  011514          2325#
NOD125  011520          2326#
NOD126  011524          2327#
NOD127  011530          2328#
NOD13   010450          2225#
NOD130  011534          2329#
NOD131  011540          2330#
NOD132  011544          2331#
NOD133  011546          2334#
NOD134  011552          2335#
NOD135  011556          2336#
NOD136  011572          2337#
NOD137  011576          2338#
NOD14   010464          2226#
NOD140  011612          2339#
NOD141  011616          2342#
NOD142  011622          2343#
NOD143  011626          2344#
NOD144  011632          2347#
NOD145  011636          2350#
NOD146  011660          2351#
NOD147  011664          2352#
NOD15   010470          2227#
NOD150  011700          2353#
NOD151  011704          2354#
NOD152  011726          2355#
NOD153  011732          2356#
NOD154  011754          2357#
NOD155  011760          2360#
NOD156  011764          2361#
NOD157  011770          2362#
NOD16   010474          2228#
NOD160  011774          2367#
NOD17   010506          2229#
NOD2    010354          2216#
NOD20   010512          2230#
NOD21   010524          2231#
NOD22   010530          2232#
NOD23   010532          2236#
NOD24   010536          2237#
NOD25   010552          2238#
NOD26   010556          2239#
NOD27   010574          2240#
NOD3    010356          2217#
NOD30   010600          2241#
NOD31   010616          2242#
NOD32   010622          2243#
```

```
NOD33     010640          2244#
NOD34     010644          2245#
NOD35     010662          2246#
NOD36     010666          2247#
NOD37     010712          2248#
NOD4      010372          2218#
NOD40     010716          2249#
NOD41     010722          2250#
NOD42     010740          2251#
NOD43     010744          2252#
NOD44     010756          2253#
NOD45     010762          2257#
NOD46     010766          2258#
NOD47     011010          2259#
NOD5      010374          2219#
NOD50     011012          2260#
NOD51     011036          2261#
NOD52     011040          2266#
NOD53     011044          2267#
NOD54     011064          2268#
NOD55     011070          2269#
NOD56     011112          2270#
NOD57     011116          2273#
NOD6      010410          2220#
NOD60     011122          2274#
NOD61     011126          2275#
NOD62     011132          2276#
NOD63     011136          2277#
NOD64     011142          2278#
NOD65     011146          2279#
NOD66     011152          2280#
NOD67     011156          2283#
NOD7      010412          2221#
NOD70     011162          2284#
NOD71     011166          2285#
NOD72     011200          2286#
NOD73     011204          2287#
NOD74     011220          2288#
NOD75     011224          2294#
NOD76     011242          2295#
NOD77     011246          2296#
NONE   =  000000          1665#
NOTNUF=   000050          1776#   2223    2226    2227    2237    2276    2278    2284    2335
NULEVT    015016          2751#   3459
NULL   =  000000          1736#
NUM    =  000014          1748#   2343
NXMM      017224          2993#   6196
N10$      010350          2214#
N100$     010762          2226    2229    2256#
N102$     010766          2257#
N104$     011012          2258    2259#
N110$     011040          2231    2265#
N111$     011044          2266#
N112$     011070          2267    2268#
N114$     011162          2283#
N115$     011156          2240    2242    2244    2246    2248    2251    2253    2282#   2288    2295    2297    2351    2353
```

```
                              2355    2357    2362
N116$    011200               2285#
N117$    011224               2286    2287    2293#
N118$    011246               2294    2295#
N120$    011326               2268    2270    2305#
N121$    011546               2310    2312    2314    2316    2318    2320    2322    2327    2333#   2344
N122$    011576               2336    2337#
N123$    011552               2334#
N125$    011774               2276    2278    2280    2284    2335    2366#
N126$    011616               2337    2339    2341#
N130$    011266               2296    2299#
N131$    011306               2300    2301#
N140$    011632               2303    2346#
N141$    011636               2349#
N142$    011664               2350    2351#
N143$    011704               2352    2353#
N144$    011732               2354    2355#
N150$    011760               2301    2359#
N20$     010470               2225    2226#
N25$     010512               2228    2229#
N30$     010530               2227    2230    2231#   2236    2237    2238    2249    2252    2257    2260    2266    2269    2275
                              2277    2279    2302    2306    2325    2338    2342    2343    2347    2356    2360    2361
N40$     010450               2223    2224#
N42$     010356               2215    2216#
N43$     010374               2217    2218#
N45$     010412               2219    2220#
N46$     010430               2221    2222#
N50$     011116               2224    2272#
N51$     011122               2273#
N52$     011126               2274#
N60$     011332               2308#
N61$     011352               2309    2310#
N62$     011374               2311    2312#
N63$     011414               2313    2314#
N64$     011434               2315    2316#
N65$     011454               2317    2318#
N66$     011474               2319    2320#
N67$     011514               2321    2324#
N70$     011520               2325#   2330
N71$     011530               2326    2327#
N72$     011534               2328#
N73$     011544               2329    2330#
N80$     010532               2222    2235#
N81$     010536               2236#
N82$     010600               2239    2240#
N83$     010622               2241    2242#
N84$     010644               2243    2244#
N85$     010666               2245    2246#
N86$     010716               2247    2248#
N87$     010744               2250    2251#
OFSET    007170               2062#   3071    5335*   5352*
OPBFPT   002520               1902#   5625
OPBUF    002524               1844    1903#   4878    5539    5546    5554    5560    5561    5573    5575    5607    5608    5630
                              5632
OPCNT    002166               1829#   4876*   5558*   5562    5563
OPEND    002646               1904#   5541
```

```
OPRMM    013575               2602#   5548
OPRMSG=  000015               1749#   2325
OPTN     036176               6299    6310#
OPTYP    012024               2390#   4288*   5674    5677    5711    5723    5760    6162    6200
OTINT =  000002               1711#   5933    5994    6086    6129
OUTHDL   035150               5936    6090    6128#
OUTHEX   035736               6160    6169    6218    6226    6231#
OUTH1    035240               6131    6149#
OUTH2    035660               6151    6220#
OUTH3    035714               6222    6228#
OUTH4    035722               6229#
OUTH6    035302               6157    6162#
OUTVEC   012020               2388#   4285*   4286*   4311
O$APTS=  000000               1403#   1447
O$AU  =  000001               1403#   1416#   1479
O$BGNR=  000001               1403#   1416#   1473
O$BGNS=  000000               1403#   1439
O$DU  =  000001               1403#   1416#   1481
O$ERRT=  000000               1403#   1489
O$GNSW=  000000               1403#   1443
O$POIN=  000001               1403#   1416#   1505
O$SETU=  000000               1403#   1433    6343
PARAM    007226               2085#   3358    3390    4486*   4497    4821    4934*   4944*   4954*   4969*   4972*   5077*   5208
                              5213    5260    5304    5427*   5538*   5600*
PAS   =  000002               1659#   4915    4945    5231    5248
PASC  =  000042               1770#   2361
PASMOD=  000030               1760#   2241
PCADD    007214               2076#   3120    3141    6019*   6067*   6128*
PCK      013547               2594#   3837
PCLKCT=  001600               1676#   4207
PCLKEN=  000111               1675#   4209
PCPM     014321               2675#   3121    3142
PEC      013557               2597#   3841
PLCK     031106               2099    5101#
PLCK2    031106               5102#
PLCK3    031122               5104#
PNCK     013545               2593#   3840
PNEC     013555               2596#   3844
PNST     013534               2590#   3836
PNT   =  001000 G             1638#
PRI   =  002000 G             1639#
PRIOR    036150               6291    6310#
PRI00 =  000000 G             1627#   4317
PRI01 =  000040 G             1626#
PRI02 =  000100 G             1625#
PRI03 =  000140 G             1624#
PRI04 =  000200 G             1623#
PRI05 =  000240 G             1622#
PRI06 =  000300 G             1621#
PRI07 =  000340 G             1620#   4357
PRNT  =  000055               1781#   2219    4552    4740
PROEM    017204               2990#   6199
PROTO =  000041               1769#
PROTOB=  000040               1684#   4967
PSCNT    007142               2047#   5023*   5385*   5388
PST      013536               2591#   3833
```

```
PTREND  007076          2024#
PTRTAB  006326          2023#   4458    4581    4622    4749    4764    4775    4782    5004
P$ACT   003314          2011#   3923    4528*
P$BUFA  003310          2009#   3878    4526*
P$CNT   003316          2012#   4020*   4027    4089*   4096*   4101
P$EXIT  023412          3881    3916#   3940    3945
P$GDBD  003325          2016#   3938*   3943*   3951    4002*   4045*   4514*   4531    4860*   4953*   4991*
P$NNUF  003324          2015#   4515*   4540    4718*   4745*   4758*   4777*   4832*   4909*   4916*   4935*   4999*
P$NUM   003320          2013#   4015*   4018*   4020    4021*   4023*   4027*   4029*   4035*   4825    4831    4852    4862
                        4864    4976
P$RADX  003322          2014#   3975*   3978*   3985*   3992    4025    4033
P$TREE  003312          2010#   3879    4527*
P$TRV   023274          3877#   4530
P$TR5   023304          3880#   3889    3909    3914
QCOPY = 000013          1747#   2338    4847
QRX   = 000004          1712#   5057    5105    5131    5181    5203    5233    5240    5251    5461    5496    5611    5830
                        5833    5975
QTX   = 000010          1713#   5080    5131    5191    5234    5242    5250    5259    5474    5478    5491    5565    5577
                        5886    5888    5956
QUALFG  003206          1970#   4529*   4844*   4847*   4850    4939*   4942    4955*   4970    4973*
QUALVL  003210          1971#   4604*   4650*   4841*   4864*   4865*
RDI   = 000200          1801#   6026
RDO   = 000200          1802#   6130    6231
REC   = 000000          1657#
RECMOD= 000031          1761#   2243
REPORT  021406          3436#   4125    4741
RESFLG  007216          2077#   4161*   4185*   4450
RESTRT  024660          4171    4197    4210    4229    4241#
RI    = 000200          1793#   2136
RMDLOP= 000047          1775#   2356
RPASS   007230          2093#   3357    4485*   4496    4504    4746*   4820    4976*   5392    5394*
RPT     021530          2184    3466#   3515    3535    3546    3560    3575    3591
RPTAA   021600          3481#
RPTDCK  022244          2183    3578#
RPTDDE  022170          2186    3563#
RPTDER  021754          2181    3517#
RPTDLE  022244          2185    3577#
RPTDSP  010306          2177#   3501
RPTDVI  022050          2182    3537#
RPTEOP  022114          2187    3547#
RPTMSB  022330          3514    3602#
RPTTXQ  021702          2177    2178    2179    2180    3503#
RPT0    021574          3457    3473    3477    3480#
RPT1    021562          3470    3476#
RQI   = 000040          1800#   6020
RSEL4   035752          5963    5973    6229*   6235#
RSEL6   035754          5964    5974    6230*   6236#
RTS   = 000040          1792#   2135
RUN   = 000004          1740#   2221    4554    4744
RUNSBM  020000          3021#   6170
RXBIT = 000004          1808#   6221
RXBUF   004326          2021#   5030    5462    5463    5482    5503    5509
RXC   = 000006          1699#   3345    3388
RXIDM   020022          3024#   6204
RXMTOT  007134          2043#   5018*   5056    5129    5308
RXM1    020122          3036#   5506
```

```
RXM2      020145       3040#   5511
RXNC      020102       3033#   5498
RXONLY    031014       2097    5054#
RXON2     031014       5055#
RXPTR     007076       2026#   4460*   5015*   5016*   5031    5055    5104    5130    5216    5306
RXQ     = 000004       1698#   3341
SCM       016111       2875#   3362
SCMD      016144       2881#   3370
SCML      016133       2879#   3366
SDVE      016100       2873#   3348
SDVI      016122       2877#   3353
SELO      011776       2373#   4268*   5672*   5681    5689    5715*   5718*   5903    5921    6020*   6026    6031    6074
SEL2      012002       2376#   4271*   4272*   5690    5714*   5904    5922    6032    6073    6130    6133    6149    6172
                       6221    6231*
SEL4      012006       2379#   4275*   4276*   5671*   5708*   5797    5836    5837*   5891    5892*   6224    6229
SEL6      012012       2382#   4279*   4280*   5670*   5710*   5713*   5727*   5747*   5750*   5755*   5757    5798*   5838*
                       5893*   6134    6153    6225    6230
SEOP      016155       2883#   3374
SETEXP= 000010         1744#   2267    4565    4837
SETTRN= 000011         1745#   2269    4840
SHFO      014165       2655#   3826
SHF1      014223       2663#   3851
SHMSG     013251       2549#   4810
SHOW    = 000002       1738#   2228    4563    4724    4759    4778
SHTAB     003250       1987#   4798    4804
SHTEND    003257       1990#   4801
SHTYP0    013305       1980    2554#
SHTYP1    013314       1980    2556#
SHTYP2    013321       1980    2557#
SHTYP3    013326       1980    2558#
SHTYP4    013333       1980    2559#
SHTYP5    013341       1980    2561#
SHTYP6    013346       1980    2562#
SHTYP7    013354       1980    2563#
SHTYTB    003230       1980#   4809
SHWOP     023022       3544    3810#   4498    4822
SIZE    = 000012       1746#   2336    4844    4850
SQD     = 040000       1794#   2137
SRXQ      016067       2871#   3340
STADD     007146       2049#   3451*   3658    4825*
START     024454       4166    4185#
STATB   = 000001       1680#   3390    3834    4961
STATUS= 000016         1750#   2294
STXC      016056       2869#   3335
STXQ      016045       2867#   3330
SVCGBL= 000000         1403#   1420    1429    1431    1433    1435    1437    1439    1441    1443    1445    1447    1449
                       1451    1453    1455    1457    1459    1461    1463    1466    1469    1471    1473    1475    1477
                       1479    1481    1483    1485    1487    1489    1491    1493    1495    1497    1499    1501    1503
                       1516    1530    1531    2410    2421    3065    3082    3095    3109    3131    3152    3166    3265
                       4122    4142    4158    4339    4353    4380    4402    5986    5993    6264    6345#   6346
SVCINS= 000001         1403#   1421    1422    1423    1424    1425    1426    1427    1428    1430    1432    1434    1436
                       1438    1440    1442    1444    1446    1448    1450    1452    1454    1456    1458    1460    1462
                       1464    1465    1467    1468    1470    1472    1474    1476    1478    1480    1482    1484    1486
                       1488    1490    1492    1494    1496    1498    1500    1502    1504    1515    1517    1529    2411
                       2413    2422    2429    3067    3068    3069    3070    3071    3072    3073    3074    3075    3076
                       3079    3084    3085    3086    3087    3088    3089    3092    3097    3098    3099    3100    3101
```

```
                         3102     3103     3106     3111     3112     3113     3114     3115     3116     3117     3118     3120     3121
                         3122     3123     3124     3125     3128     3133     3134     3135     3136     3137     3138     3139     3141
                         3142     3143     3144     3145     3146     3149     3154     3155     3156     3157     3158     3159     3160
                         3163     3168     3169     3170     3171     3172     3173     3174     3175     3178     3181     3182     3291
                         3399     3400     3401     3402     3403     3407     3408     3409     3410     3411     3441     3442     3443
                         3444     3445     3446     3459     3460     3461     3462     3463     3482     3483     3484     3485     3486
                         3492     3493     3494     3495     3496     3497     3498     3499     3500     3507     3508     3509     3510
                         3511     3512     3513     3521     3522     3523     3524     3525     3526     3528     3529     3530     3531
                         3532     3533     3534     3551     3552     3553     3554     3555     3556     3557     3558     3567     3568
                         3569     3570     3571     3572     3573     3574     3582     3583     3584     3585     3586     3587     3588
                         3589     3603     3604     3605     3606     3607     3623     3624     3625     3626     3627     3661     3662
                         3663     3664     3665     3666     3671     3672     3673     3674     3675     3676     3677     3680     3681
                         3682     3683     3684     3685     3725     3726     3727     3728     3729     3822     3823     3824     3825
                         3826     3827     3828     3829     3830     3848     3849     3850     3851     3852     3853     3854     3855
                         3997     3998     3999     4000     4001     4040     4041     4042     4043     4044     4133     4163     4164
                         4166     4168     4169     4171     4173     4174     4176     4179     4180     4182     4190     4191     4192
                         4194     4200     4201     4202     4204     4213     4215     4220     4221     4222     4223     4224     4225
                         4226     4227     4232     4233     4234     4235     4236     4238     4239     4257     4258     4259     4261
                         4292     4293     4294     4295     4296     4297     4302     4303     4304     4305     4306     4307     4309
                         4310     4311     4312     4313     4314     4317     4318     4320     4321     4328     4344     4357     4358
                         4360     4363     4364     4371     4384     4385     4392     4406     4407     4414     4444     4445     4446
                         4447     4448     4489     4490     4491     4492     4493     4501     4503     4507     4508     4517     4518
                         4519     4520     4521     4522     4523     4524     4534     4535     4536     4537     4538     4543     4544
                         4545     4546     4547     4571     4572     4573     4574     4575     4576     4586     4587     4588     4589
                         4590     4591     4612     4613     4614     4615     4616     4617     4632     4633     4634     4635     4636
                         4637     4729     4730     4731     4732     4733     4734     4808     4809     4810     4811     4812     4813
                         4814     4855     4856     4857     4858     4859     4885     4886     4887     4888     4889     4948     4949
                         4950     4951     4952     4994     4995     4996     4997     4998     5328     5329     5330     5331     5346
                         5347     5348     5349     5359     5360     5361     5362     5442     5443     5444     5445     5446     5451
                         5452     5453     5454     5544     5545     5546     5547     5548     5549     5550     5551     5602     5603
                         5604     5605     5606     5625     5626     5627     5628     5629     5684     5694     5695     5696     5697
                         5908     5909     5910     5911     5916     5927     5928     5929     5930     5948     5949     5950     5951
                         5967     5968     5969     5970     5990     5997     6025     6036     6037     6038     6039     6078     6079
                         6080     6081     6085     6138     6139     6140     6141     6146     6147     6214     6215     6216     6217
                         6246     6263     6270     6271     6272     6280     6281     6282     6283     6285     6286     6287     6288
                         6290     6291     6292     6293     6294     6298     6299     6300     6301     6302     6307     6342     6343
                         6344
SVCSUB= 000001           1403#
SVCTAG= 000001           1403#    1560     3078     3091     3105     3127     3148     3162     3177     3290     3447     4132     4228
                         4327     4343     4370     4391     4413     4525     5552     5989     5996     6245     6308
SVCTST= 000001           1403#    4429
S$LSYM= 010000           1403#    1561#    3079#    3092#    3106#    3128#    3149#    3163#    3178#    3291#    3442     3447     3448#
                         4133#    4221     4228     4229#    4328#    4344#    4371#    4392#    4414#    4518     4525     4526#    5545
                         5552     5553#    5990#    5997#    6246#    6309#
S1          024442       4176     4178#
S2          024516       4194     4199#
S3          024566       4204     4212#
S4          024634       4215     4231#
TABEX       013773       2632#    4586     4632
TAL     =   000005       1662#    4932     5634
TALCK       032750       2102     5537#    5574     5576
TALMOD=     000035       1765#    2252
TCURAD      007124       2038#    4469*    4472     4597     4602*    4785*
TEMP        007172       2063#    3331*    3336*    3341*    3345*    3349*    3354*    3363*    3367*    3371*    3375*    3388     3414
                         3415*    3416*    3417     3439*    3443     3448     3669*    3672     3731*    3732*    3733     3771*    3772*
                         3778     3812*    3825     3833*    3836*    3850     4806*    4808     4869*    4870*    4874     4877     5222*
```

```
                        5223*   5227
TEMP1   007174          2064#   3330*   3335*   3340*   3348*   3353*   3362*   3366*   3370*   3374*   3407    3819*   3823
                        3837*   3840*   3849
TEMP2   007176          2065#   3355*   3356*   3420    3774*   3776*   3780    3820*   3822    3841*   3844*   3848    5184*
                        5194*   5205*   5211    5225*   5226    5244*   5312*   5388*   5463*   5470*   5484*   5488*   5500*
                        5561*   5570*   5608*   5619*   5688*   5902*   5920*   5943*   5962*   6030*   6072*   6132*   6171*
TEMP3   007200          2066#   3097    3113    3134    3155    3170    3357*   3421    3815*   3818*   3824    5186*   5196*
                        5206*   5212    5223    5224    5225    5245*   5314*   5324    5389*   5465*   5472*   5482*   5489*
                        5501*   5562*   5571*   5610*   5620*   5689*   5903*   5921*   5944*   5963*   6031*   6074*   6133*
                        6172*
TEMP4   007202          2067#   3084    3112    3133    3154    3169    3358*   3385*   3422    5321*   5334*   5339*   5340
                        5355    5387*   5483*   5690*   5904*   5922*   5945*   5964*   6032*   6073*   6134*   6153*   6156
                        6164    6176    6179    6182    6185    6188    6191    6194    6197    6202    6205    6208
TIMERS  007276          2119#   3283    3287*   5896*   5900
TIMER1  007272          2117#   3277    3279*   4438*   4439    5669*   5685    6021*   6022    6068*   6069
TIMER2  007274          2118#   3280    3282*
TIMMIN  007264          2113#   3274*   3419    4241*
TIMOM   017327          3006#   6181
TIMSEC  007266          2114#   3271*   3272    3275*   3418    4242*
TIMTCK  007270          2115#   3268*   3270*   3285    3416    4243*
TM    = 001000          1795#   2138
TOINOT  034264          5897#   5931    5934    5938
TOIN1   034346          5901    5915#
TOIN2   034432          5918    5933#
TOORIO  035026          5707    5726    5740    5746    5796    5835    5890    6067#   6082
TOOR1   035112          6070    6084#
TOOR2   035130          6088    6091#
TOOR3   035040          6069#   6092
TOTCC   007166          2061#   3718*   3719    3730*   3732    3734*   4455*   4567*   4568    4601    4608*   4609    4649
TRA   = 000001          1658#   4929
TRAMOD= 000034          1764#   2250
TRVACT  023414          3910    3921#   3937    3942    3947    3950    3970    4036    4059    4080    4104
TRVALN  024206          3899    4063#
TRVALP  024142          3898    4049#
TRVBIF  023520          3895    3950#
TRVBR   023510          3894    3947#
TRVBRC  023434          3908    3928#   3948    3953    3972    4046    4061    4082    4108
TRVDEC  023614          3901    3975#
TRVERR  023452          3892    3937#
TRVEXI  023472          3893    3942#
TRVNMA  023634          3976    3979#
TRVNOB  023444          3933#   3954    3971    4037    4060    4081
TRVNUM  023626          3897    3978#
TRVOCT  023626          3900    3977#
TRVSPA  023542          3896    3956#
TRVSTR  024274          3902    4086#
TSEL4   035746          5944    5954    6224*   6233#
TSEL6   035750          5945    5955    6225*   6234#
TTL   = 000001          1666#   4509    5716
TTLLOP= 000044          1772#   2350
TTOTCC  007122          2037#   4456*   4567    4578    4601*   4781*
TXBUF   003326          2020#   4469    4784    5483
TXC   = 000002          1697#   3336
TXMTOT  007120          2036#   4475*   4580*   4582    4603*   4776    4780*   5079    5102    5127
TXNC    020062          3030#   5481
TXONLY  031046          2098    5077#
```

```
TXON2   031054          5078#
TXPTR   007100          2027#   4459*   4473    4581*   4595*   4596    4600*   4782*   4783    5005*   5078    5103    5128
TXQ   = 000000          1696#   3331
T$ARGC= 000001          1421#   1422#   1423#   1424#   1425#   1426#   3067#   3076    3084#   3089    3097#   3103    3111#
                        3118    3120#   3125    3133#   3139    3141#   3146    3154#   3160    3168#   3175    3399#   3403
                        3407#   3411    3459#   3463    3482#   3486    3492#   3500    3507#   3513    3521#   3526    3528#
                        3534    3551#   3558    3567#   3574    3582#   3589    3603#   3607    3623#   3627    3661#   3666
                        3671#   3677    3680#   3685    3725#   3729    3822#   3830    3848#   3855    3997#   4001    4040#
                        4044    4232#   4236    4444#   4448    4489#   4493    4534#   4538    4543#   4547    4571#   4576
                        4586#   4591    4612#   4617    4632#   4637    4729#   4734    4808#   4814    4855#   4859    4885#
                        4889    4948#   4952    4994#   4998    5442#   5446    5602#   5606    5625#   5629
T$CODE= 005032          3444#   4223#   4520#   5547#   6270#   6280#   6285#   6290#   6298#
T$ERRN= 000023          1403#   5329#   5347#   5360#   5452#   5695#   5909#   5928#   5949#   5968#   6037#   6079#   6139#
                        6215#
T$EXCP= 000000          4223#   4228    4520#   4525    5547#   5552    6280#   6284    6285#   6289    6290#   6295    6298#
                        6303
T$FLAG= 000040          3181#   3183    4238#   4320#   4363#   4384#   4386    4406#   4408    4507#   6146#
T$GMAN= 000000          1403#   4220#   4229#   4517#   4520    4526#   5544#   5547    5553#
T$HILI= 000007          4223#   4227    4520#   4524    5547#   5551    6280#   6283    6285#   6288    6290#   6294    6298#
                        6302
T$LAST= 000001          1403#   6343#
T$LOLI= 000000          4223#   4226    4520#   4523    5547#   5550    6280#   6282    6285#   6287    6290#   6293    6298#
                        6301
T$LSYM= 010000          1403#   1561    3079    3092    3106    3128    3149    3163    3178    3291    4133    4328    4344
                        4371    4392    4414    5990    5997    6246    6309
T$LTNO= 000001          6346#
T$NEST= 177777          1403#   1405#   1529#   1560#   3065#   3078#   3082#   3091#   3095#   3105#   3109#   3127#   3131#
                        3148#   3152#   3162#   3166#   3177#   3265#   3290#   4122#   4132#   4142#   4149#   4158#   4327#
                        4339#   4343#   4353#   4370#   4380#   4391#   4402#   4413#   4430#   5986#   5989#   5993#   5996#
                        6245#   6263#   6307#   6347#
T$NSO = 000000          1405#   6347
T$NS1 = 000004          1529#   1560    3065#   3078    3082#   3091    3095#   3105    3109#   3127    3131#   3148    3152#
                        3162    3166#   3177    3265#   3290    4122#   4132    4142#   4149    4158#   4327    4339#   4343
                        4353#   4370    4380#   4391    4402#   4413    4430#   6245    6263#   6307
T$NS2 = 000010          5986#   5989    5993#   5996
T$PTNU= 000000          1403#
T$SAVL= 177777          1403#
T$SEGL= 177777          1403#
T$SUBN= 000000          1403#   4429#
T$TAGL= 177777          1403#
T$TAGN= 010024          1403#   1529#   3065#   3082#   3095#   3109#   3131#   3152#   3166#   3265#   4122#   4142#   4158#
                        4339#   4353#   4380#   4402#   4430#   5986#   5993#   6263#
T$TEMP= 000000          1517#   1518#   1560#   3078#   3091#   3105#   3127#   3148#   3162#   3177#   3181#   3182    3290#
                        3444#   4132#   4149#   4223#   4238#   4239    4320#   4321    4327#   4343#   4363#   4364    4370#
                        4384#   4385    4391#   4406#   4407    4413#   4507#   4508    4520#   5547#   5989#   5996#   6146#
                        6147    6245#   6270#   6280#   6285#   6290#   6298#   6307#   6347#
T$TEST= 000001          1403#   4429#   6346
T$TSTM= 177777          1403#   3075    3079    3088    3092    3102    3106    3117    3124    3128    3138    3145    3149
                        3159    3163    3174    3178    3402    3410    3441    3462    3485    3499    3512    3525    3533
                        3557    3573    3588    3606    3626    3665    3676    3684    3728    3829    3854    4000    4043
                        4133    4164    4169    4174    4180    4191    4201    4213    4220    4235    4238    4258    4296
                        4306    4313    4318    4320    4328    4344    4358    4360    4363    4371    4392    4414    4447
                        4492    4501    4507    4517    4537    4546    4575    4590    4616    4636    4733    4813    4858
                        4888    4951    4997    5328    5346    5359    5445    5451    5544    5605    5628    5684    5694
                        5908    5916    5927    5948    5967    6025    6036    6078    6085    6138    6146    6214    6246
T$TSTS= 000001          1403#   4430#
```

```
T$$AU = 010017          4402#   4406    4413
T$$AUT= 010014          4339#   4343
T$$CLE= 010015          4353#   4363    4370
T$$DU = 010016          4380#   4384    4391
T$$HAR= 010023          6263#   6308
T$$HW = 010000          1529#   1560
T$$INI= 010013          4158#   4238    4320    4327
T$$MSG= 010007          3065#   3078    3082#   3091    3095#   3105    3109#   3127    3131#   3148    3152#   3162    3166#
                        3177    3181
T$$PRO= 010012          4142#
T$$RPT= 010011          4122#   4132
T$$SRV= 010022          3265#   3290    5986#   5989    5993#   5996
T$$TES= 010020          4430#   4507    6146    6245
T1      025324 G        1517    4429#
UAM   = 000200 G        1636#
UPTABL  031350          5208#
UPTA1   031436          5214    5221#
UPTA3   031434          5218    5220#
UPTA4   031374          5209    5213#
UPTEX   031506          5220    5230#
VECTOR  036115          6286    6310#
X$    = 000161          1406#   2214#   2215#   2216#   2217#   2218#   2219#   2220#   2221#   2222#   2223#   2224#   2225#
                        2226#   2227#   2228#   2229#   2230#   2231#   2232#   2236#   2237#   2238#   2239#   2240#   2241#
                        2242#   2243#   2244#   2245#   2246#   2247#   2248#   2249#   2250#   2251#   2252#   2253#   2257#
                        2258#   2259#   2260#   2261#   2266#   2267#   2268#   2269#   2270#   2273#   2274#   2275#   2276#
                        2277#   2278#   2279#   2280#   2283#   2284#   2285#   2286#   2287#   2288#   2294#   2295#   2296#
                        2297#   2300#   2301#   2302#   2303#   2306#   2309#   2310#   2311#   2312#   2313#   2314#   2315#
                        2316#   2317#   2318#   2319#   2320#   2321#   2322#   2325#   2326#   2327#   2328#   2329#   2330#
                        2331#   2334#   2335#   2336#   2337#   2338#   2339#   2342#   2343#   2344#   2347#   2350#   2351#
                        2352#   2353#   2354#   2355#   2356#   2357#   2360#   2361#   2362#   2367#
X$ALWA= 000000          1403#
X$FALS= 000040          1403#
X$OFFS= 000400          1403#
X$TRUE= 000020          1403#
$PATCH  036262          6337#
.     = 036346          1403#   1897#   1903#   1934    1941    1967#   1991#   2020#   2021#   2022#   2023#   2123#   2124#
                        2217#   2223#   2230#   2237#   2239#   2245#   2247#   2258#   2260#   2285#   2287#   2294#   2296#
                        2300#   2302#   2309#   2311#   2313#   2315#   2317#   2336#   2338#   2354#   2413#   2429#   2863#
                        3011#   3017#   3018#   3020#   3182    4239    4321    4364    4385    4407    4508    6147    6311#
                        6338#
```

```
BCOMPL       1#    1403#    4165    4170    4181    4502
BERROR       1#    1403#
BGNAU        1#    1403#    4401
BGNAUT       1#    1403#    4338
BGNCLN       1#    1403#    4352
BGNDU        1#    1403#    4379
BGNHRD       1#    1403#    6262
BGNHW        1#    1403#    1528
BGNINI       1#    1403#    4157
BGNMOD       1#    1403#    1404
BGNMSG       1#    1403#    3064    3081    3094    3108    3130    3151    3165
BGNPRO       1#    1403#    4141
BGNPTA       1#    1403#
BGNRPT       1#    1403#    4121
BGNSEG       1#    1403#
BGNSET       1#    1403#
BGNSFT       1#    1403#
BGNSRV       1#    1403#    3264    5985    5992
BGNSUB       1#    1403#
BGNSW        1#    1403#
BGNTST       1#    1403#    4428
BNCOMP       1#    1403#    4175    4193    4203    4214    4260
BNERRO       1#    1403#
BREAK        1#    1403#    5683    5915    6024    6084
BRESET       1#    1403#    4359
CKLOOP       1#    1403#
CLI       1407#    2213     2214    2215    2216    2217    2218    2219    2220    2221    2222    2223    2224    2225    2226
          2227     2228     2229    2230    2231    2235    2236    2237    2238    2239    2240    2241    2242    2243    2244
          2245     2246     2247    2248    2249    2250    2251    2252    2256    2257    2258    2259    2260    2265    2266
          2267     2268     2269    2272    2273    2274    2275    2276    2277    2278    2279    2282    2283    2284    2285
          2286     2287     2293    2294    2295    2296    2299    2300    2301    2302    2305    2308    2309    2310    2311
          2312     2313     2314    2315    2316    2317    2318    2319    2320    2321    2324    2325    2326    2327    2328
          2329     2330     2333    2334    2335    2336    2337    2338    2341    2342    2343    2346    2349    2350    2351
          2352     2353     2354    2355    2356    2359    2360    2361    2366
CLOCK        1#    1403#    4189    4199
CLOSE        1#    1403#
CLRVEC       1#    1403#
COMMEN       1#    1403#
DELAY        1#    1403#
DESCRI       1#    1403#    2420
DEVTYP       1#    1403#    2409
DISPAT       1#    1403#    1514
DISPLA       1#    1403#
DOCLN        1#    1403#
DODU         1#    1403#
DORPT        1#    1403#
ENDAU        1#    1403#    4412
ENDAUT       1#    1403#    4342
ENDCLN       1#    1403#    4369
ENDCOM       1#    1403#
ENDDU        1#    1403#    4390
ENDHRD       1#    1403#    6306
ENDHW        1#    1403#    1559
ENDINI       1#    1403#    4326
ENDMOD       1#    1403#    6346
ENDMSG       1#    1403#    3077    3090    3104    3126    3147    3161    3176
```

```
ENDPRO         1#    1403#    4148
ENDPTA         1#    1403#
ENDRPT         1#    1403#    4131
ENDSEG         1#    1403#
ENDSET         1#    1403#
ENDSFT         1#    1403#
ENDSRV         1#    1403#    3289     5988     5995
ENDSUB         1#    1403#
ENDSW          1#    1403#
ENDTST         1#    1403#    6244
EQUALS         1#    1403#    1576
ERRDF          1#    1403#
ERRHRD         1#    1403#    5450
ERROR          1#    1403#
ERRSF          1#    1403#
ERRSOF         1#    1403#    5327     5345     5358     5693     5907     5926     5947     5966     6035     6077     6137     6213
ERRTBL         1#    1403#
ESCAPE         1#    1403#    6145
EXIT           1#    1403#    3180     4237     4319     4362     4383     4405     4506
FEQUAL         1#    1403#
GETBYT         1#    1403#
GETPRI         1#    1403#
GETWOR         1#    1403#
GMANIA         1#    1403#
GMANID         1#    1403#    4219     4516     5543
GMANIL         1#    1403#    3440
GPHARD         1#    1403#    4256
GPRMA          1#    1403#    6279     6284
GPRMD          1#    1403#    4220#    4223     4517#    4520     5544#    5547     6289     6297
GPRML          1#    1403#    3441#    3444     6269
HEADER         1#    1403#    1419
INLOOP         1#    1403#
IOSETU         1#    1403#
IOSTAR         1#    1403#
KT11           1#    1403#
LASTAD         1#    1403#    6341
MANUAL         1#    1403#    4500
MEMORY         1#    1403#
M$BYTE         1#    1403#    1420#    1426     1427     1428
M$CHEC         1#    1403#    3181#    4238#    4320#    4363#    4384#    4406#    4507#
M$CNTO         1#    1403#    3444#    4223#    4520#    5547#    6270#    6280#    6285#    6290#    6298#
M$COUN         1#    1403#    3067#    3084#    3097#    3111#    3120#    3133#    3141#    3154#    3168#    3399#    3407#    3459#    3482#
            3492#    3507#    3521#    3528#    3551#    3567#    3582#    3603#    3623#    3661#    3671#    3680#    3725#    3822#    3848#
            3997#    4040#    4232#    4444#    4489#    4534#    4543#    4571#    4586#    4612#    4632#    4729#    4808#    4855#    4885#
            4948#    4994#    5442#    5602#    5625#
M$DATA         1#    1403#    1420#    1429     1431     1433     1435     1437     1439     1441     1443     1445     1447     1449     1451
            1453     1455     1457     1459#    1461     1463     1466     1469     1471     1473     1475     1477     1479     1481     1483
            1485     1487     1489     1491     1493     1495     1497     1499     1501     1503     2410#    2421#
M$DECR         1#    1403#    1560#    3078#    3091#    3105#    3127#    3148#    3162#    3177#    3290#    4132#    4149#    4327#    4343#
            4370#    4391#    4413#    5989#    5996#    6245#    6307#    6347#
M$DEFA         1#    1403#    3444#    4223#    4520#    5547#    6270#    6280#    6285#    6290#    6298#
M$ENDE         1#    1403#    1560#    3078#    3091#    3105#    3127#    3148#    3162#    3177#    3290#    4132#    4327#    4343#    4370#
            4391#    4413#    5989#    5996#    6245#    6307#    6347#
M$ERRI         1#    1403#    5328#    5346#    5359#    5451#    5694#    5908#    5927#    5948#    5967#    6036#    6078#    6138#    6214#
M$ESCA         1#    1403#    6146#    6147
M$ESCS         1#    1403#    6146#
```

```
M$EXCP      1#   1403#   4223#   4520#   5547#   6280#   6285#   6290#   6298#
M$EXIT      1#   1403#   3181#   4238#   4239    4320#   4321    4363#   4364    4384#   4406#   4507#   4508
M$EXSE      1#   1403#   3181#   4238#   4320#   4363#   4384#   4406#   4507#
M$EXTJ      1#   1403#   3181#   3182    4238#   4320#   4363#   4384#   4385    4406#   4407    4507#
M$GEN       1#   1403#   1420#   1429#   1431#   1433#   1435#   1437#   1439#   1441#   1443#   1445#   1447#   1449#   1451#
         1453#   1455#   1457#   1459#   1461#   1463#   1466#   1469#   1471#   1473#   1475#   1477#   1479#   1481#   1483#
         1485#   1487#   1489#   1491#   1493#   1495#   1497#   1499#   1501#   1503#   1516#   1530#   1531#   1560#   2410#
         2421#   3065#   3078#   3082#   3091#   3095#   3105#   3109#   3127#   3131#   3148#   3152#   3162#   3166#   3177#
         3265#   3290#   3447#   4122#   4132#   4142#   4158#   4228#   4327#   4339#   4343#   4353#   4370#   4380#   4391#
         4402#   4413#   4429#   4525#   5552#   5986#   5989#   5993#   5996#   6245#   6264#   6308#   6345#
M$GENB      1#   1403#   3441#   3442    4220#   4221    4517#   4518    5544#   5545
M$GETS      1#   1403#   1560#   3078#   3091#   3105#   3127#   3148#   3162#   3177#   3290#   4132#   4149#   4327#   4343#
         4370#   4391#   4413#   5989#   5996#   6245#   6307#   6347#
M$GETT      1#   1403#   3181#   4238#   4320#   4363#   4384#   4406#   4507#   6146#
M$GNGB      1#   1403#   1405#   1420#   1429#   1431#   1433#   1435#   1437#   1439#   1441#   1443#   1445#   1447#   1449#
         1451#   1453#   1455#   1457#   1459#   1461#   1463#   1466#   1469#   1471#   1473#   1475#   1477#   1479#   1481#
         1483#   1485#   1487#   1489#   1491#   1493#   1495#   1497#   1499#   1501#   1503#   1515#   1516    1529#   1530
         1531    2410#   2421#   3065#   3082#   3095#   3109#   3131#   3152#   3166#   3265#   4122#   4142#   4158#   4339#
         4353#   4380#   4402#   5986#   5993#   6263#   6264    6342#   6345
M$GNIN      1#   1403#   1420#   1421    1422    1423    1424    1425    1426#   1427#   1428#   1429#   1430    1431#   1432
         1433#   1434    1435#   1436    1437#   1438    1439#   1440    1441#   1442    1443#   1444    1445#   1446    1447#
         1448    1449#   1450    1451#   1452    1453#   1454    1455#   1456    1457#   1458    1459#   1460    1461#   1462
         1463#   1464    1465    1466#   1467    1468#   1469#   1470    1471#   1472    1473#   1474    1475#   1476    1477#
         1478    1479#   1480    1481#   1482    1483#   1484    1485#   1486    1487#   1488    1489#   1490    1491#   1492
         1493#   1494    1495#   1496    1497#   1498    1499#   1500    1501#   1502    1503#   1504    1515#   1517#   1529#
         2410#   2411    2413    2421#   2422    2429    3067#   3068    3069#   3070    3071#   3072#   3073#   3074    3075#
         3076    3079#   3084#   3085#   3086#   3087    3088#   3089    3092#   3097#   3098#   3099#   3100#   3101    3102#
         3103    3106#   3111#   3112#   3113#   3114#   3115#   3116    3117#   3118    3120#   3121#   3122#   3123    3124#
         3125    3128#   3133#   3134#   3135#   3136#   3137    3138#   3139    3141#   3142#   3143#   3144    3145#   3146
         3149#   3154#   3155#   3156#   3157#   3158    3159#   3160    3163#   3168#   3169#   3170#   3171#   3172#   3173
         3174#   3175    3178#   3181#   3182#   3290#   3291    3399#   3400#   3401    3402#   3403    3407#   3408#   3409
         3410#   3411    3441#   3442#   3443#   3444#   3445    3446    3459#   3460#   3461    3462#   3463    3482#   3483#
         3484    3485#   3486    3492#   3493#   3494#   3495#   3496#   3497#   3498    3499#   3500    3507#   3508#   3509#
         3510#   3511    3512#   3513    3521#   3522#   3523#   3524    3525#   3526    3528#   3529#   3530#   3531#   3532
         3533#   3534    3551#   3552#   3553#   3554#   3555#   3556    3557#   3558    3567#   3568#   3569#   3570#   3571#
         3572    3573#   3574    3582#   3583#   3584#   3585#   3586#   3587    3588#   3589    3603#   3604#   3605    3606#
         3607    3623#   3624#   3625    3626#   3627    3661#   3662#   3663#   3664    3665#   3666    3671#   3672    3673#
         3674#   3675    3676#   3677    3680#   3681#   3682#   3683    3684#   3685    3725#   3726#   3727    3728#   3729
         3822#   3823#   3824#   3825#   3826#   3827#   3828    3829#   3830    3848#   3849#   3850#   3851#   3852#   3853
         3854#   3855    3997#   3998#   3999    4000#   4001    4040#   4041#   4042    4043#   4044    4133#   4163#   4164#
         4166#   4168#   4169#   4171#   4173#   4174#   4176#   4179#   4180#   4182#   4190#   4191#   4192#   4194#   4200#
         4201#   4202#   4204#   4213#   4215#   4220#   4221#   4222#   4223#   4224    4225    4226    4227    4232#   4233#
         4234    4235#   4236    4238#   4239#   4257#   4258#   4259#   4261#   4292#   4293#   4294#   4295#   4296#   4297
         4302#   4303#   4304#   4305#   4306#   4307    4309#   4310#   4311#   4312#   4313#   4314    4317#   4318#   4320#
         4321#   4328#   4344#   4357#   4358#   4360#   4363#   4364#   4371#   4384#   4385#   4392#   4406#   4407#   4414#
         4444#   4445#   4446    4447#   4448    4489#   4490#   4491    4492#   4493    4501#   4503#   4507#   4508#   4517#
         4518#   4519#   4520#   4521    4522    4523    4524    4534#   4535#   4536    4537#   4538    4543#   4544#   4545
         4546#   4547    4571#   4572#   4573#   4574    4575#   4576    4586#   4587#   4588#   4589    4590#   4591    4612#
         4613#   4614#   4615    4616#   4617    4632#   4633#   4634#   4635    4636#   4637    4729#   4730#   4731#   4732
         4733#   4734    4808#   4809#   4810#   4811#   4812    4813#   4814    4855#   4856#   4857    4858#   4859    4885#
         4886#   4887    4888#   4889    4948#   4949#   4950    4951#   4952    4994#   4995#   4996    4997#   4998    5328#
         5329#   5330#   5331#   5346#   5347#   5348#   5349#   5359#   5360#   5361#   5362#   5442#   5443#   5444    5445#
         5446    5451#   5452#   5453#   5454#   5544#   5545#   5546#   5547#   5548    5549    5550    5551    5602#   5603#
         5604    5605#   5606    5625#   5626#   5627    5628#   5629    5684#   5694#   5695#   5696#   5697#   5908#   5909#
         5910#   5911#   5916#   5927#   5928#   5929#   5930#   5948#   5949#   5950#   5951#   5967#   5968#   5969#   5970#
         5989#   5990    5996#   5997    6025#   6036#   6037#   6038#   6039#   6078#   6079#   6080#   6081#   6085#   6138#
```

```
           6139#   6140#   6141#   6146#   6147#   6214#   6215#   6216#   6217#   6246#   6263#   6270#   6271    6272    6280#
           6281    6282    6283    6285#   6286    6287    6288    6290#   6291    6292    6293    6294    6298#   6299    6300
           6301    6302    6307#   6342#   6343#   6344#
M$GNLS        1#   1403#   3441#   3447    4220#   4228    4517#   4525    5544#   5552
M$GNSU        1#   1403#
M$GNTA        1#   1403#   1560#   3078#   3091#   3105#   3127#   3148#   3162#   3177#   3290#   4132#   4327#   4343#   4370#
           4391#   4413#   5989#   5996#   6245#   6307#   6308
M$GNTE        1#   1403#   4429#
M$HAPT        1#   1403#   1420#
M$HNAP        1#   1403#   1420#   1459
M$INCR        1#   1403#   1405#   1529#   3065#   3075#   3079#   3082#   3088#   3092#   3095#   3102#   3106#   3109#   3117#
           3124#   3128#   3131#   3138#   3145#   3149#   3152#   3159#   3163#   3166#   3174#   3178#   3265#   3402#   3410#
           3441#   3448    3462#   3485#   3499#   3512#   3525#   3533#   3557#   3573#   3588#   3606#   3626#   3665#   3676#
           3684#   3728#   3829#   3854#   4000#   4043#   4122#   4133#   4142#   4158#   4164#   4169#   4174#   4180#   4191#
           4201#   4213#   4220#   4229    4235#   4238#   4258#   4296#   4306#   4313#   4318#   4320#   4328#   4339#   4344#
           4353#   4358#   4360#   4363#   4371#   4380#   4392#   4402#   4414#   4429#   4430#   4447#   4492#   4501#   4507#
           4517#   4526    4537#   4546#   4575#   4590#   4616#   4636#   4733#   4813#   4858#   4888#   4951#   4997#   5328#
           5346#   5359#   5445#   5451#   5544#   5553    5605#   5628#   5684#   5694#   5908#   5916#   5927#   5948#   5967#
           5986#   5993#   6025#   6036#   6078#   6085#   6138#   6146#   6214#   6246#   6263#
M$IOSE        1#   1403#
M$LDRO        1#   1403#   4163#   4168#   4173#   4179#   4190#   4200#   4257#   4317#   4357#
M$MASK        1#   1403#
M$MCHI        1#   1403#
M$MCLO        1#   1403#
M$MSK1        1#   1403#
M$POP         1#   1403#   1560#   3078#   3091#   3105#   3127#   3148#   3162#   3177#   3290#   4132#   4149#   4327#   4343#
           4370#   4391#   4413#   5989#   5996#   6245#   6307#   6347#
M$PRIN        1#   1403#   3067#   3084#   3097#   3111#   3120#   3133#   3141#   3154#   3168#   3399#   3407#   3459#   3482#
           3492#   3507#   3521#   3528#   3551#   3567#   3582#   3603#   3623#   3661#   3671#   3680#   3725#   3822#   3848#
           3997#   4040#   4232#   4444#   4489#   4534#   4543#   4571#   4586#   4612#   4632#   4729#   4808#   4855#   4885#
           4948#   4994#   5442#   5602#   5625#
M$PUSH        1#   1403#   1405#   1529#   3065#   3082#   3095#   3109#   3131#   3152#   3166#   3265#   4122#   4142#   4158#
           4339#   4353#   4380#   4402#   4429#   4430    5986#   5993#   6263#
M$PUT         1#   1403#   3067#   3084#   3097#   3111#   3120#   3133#   3141#   3154#   3168#   3399#   3407#   3459#   3482#
           3492#   3507#   3521#   3528#   3551#   3567#   3582#   3603#   3623#   3661#   3671#   3680#   3725#   3822#   3848#
           3997#   4040#   4232#   4292#   4302#   4309#   4444#   4489#   4534#   4543#   4571#   4586#   4612#   4632#   4729#
           4808#   4855#   4885#   4948#   4994#   5442#   5602#   5625#
M$PUT1        1#   1403#   3067#   3069    3071    3072    3073    3084#   3085    3086    3097#   3098    3099    3100    3111#
           3112    3113    3114    3115    3120#   3121    3122    3133#   3134    3135    3136    3141#   3142    3143    3154#
           3155    3156    3157    3168#   3169    3170    3171    3172    3399#   3400    3407#   3408    3459#   3460    3482#
           3483    3492#   3493    3494    3495    3496    3497    3507#   3508    3509    3510    3521#   3522    3523    3528#
           3529    3530    3531    3551#   3552    3553    3554    3555    3567#   3568    3569    3570    3571    3582#   3583
           3584    3585    3586    3603#   3604    3623#   3624    3661#   3662    3663    3671#   3673    3674    3680#   3681
           3682    3725#   3726    3822#   3823    3824    3825    3826    3827    3848#   3849    3850    3851    3852    3997#
           3998    4040#   4041    4232#   4233    4292#   4293    4294    4295    4302#   4303    4304    4305    4309#   4310
           4311    4312    4444#   4445    4489#   4490    4534#   4535    4543#   4544    4571#   4572    4573    4586#   4587
           4588    4612#   4613    4614    4632#   4633    4634    4729#   4730    4731    4808#   4809    4810    4811    4855#
           4856    4885#   4886    4948#   4949    4994#   4995    5442#   5443    5602#   5603    5625#   5626
M$RADI        1#   1403#   3444#   4223#   4520#   5547#   6270#   6280#   6285#   6290#   6298#
M$RBRO        1#   1403#
M$RNRO        1#   1403#   4190#   4192    4200#   4202    4257#   4259
M$SETS        1#   1403#   1405#   1529#   3065#   3082#   3095#   3109#   3131#   3152#   3166#   3265#   4122#   4142#   4158#
           4339#   4353#   4380#   4402#   4430#   5986#   5993#   6263#
M$STAR        1#   1403#
M$SVC         1#   1403#   3067#   3075    3078#   3079    3084#   3088    3091#   3092    3097#   3102    3105#   3106    3111#
           3117    3120#   3124    3127#   3128    3133#   3138    3141#   3145    3148#   3149    3154#   3159    3162#   3163
```

```
          3168#   3174    3177#   3178    3181#   3399#   3402    3407#   3410    3441#   3459#   3462    3482#   3485    3492#
          3499    3507#   3512    3521#   3525    3528#   3533    3551#   3557    3567#   3573    3582#   3588    3603#   3606
          3623#   3626    3661#   3665    3671#   3676    3680#   3684    3725#   3728    3822#   3829    3848#   3854    3997#
          4000    4040#   4043    4132#   4133    4163#   4164    4168#   4169    4173#   4174    4179#   4180    4190#   4191
          4200#   4201    4213#   4220#   4232#   4235    4238#   4257#   4258    4292#   4296    4302#   4306    4309#   4313
          4317#   4318    4320#   4327#   4328    4343#   4344    4357#   4358    4360#   4363#   4370#   4371    4384#   4391#
          4392    4406#   4413#   4414    4444#   4447    4489#   4492    4501#   4507#   4517#   4534#   4537    4543#   4546
          4571#   4575    4586#   4590    4612#   4616    4632#   4636    4729#   4733    4808#   4813    4855#   4858    4885#
          4888    4948#   4951    4994#   4997    5328    5346    5359    5442#   5445    5451    5544#   5602#   5605    5625#
          5628    5684#   5694    5908    5916#   5927    5948    5967    6025#   6036    6078    6085#   6138    6146#   6214
          6245#   6246
M$TLAB       1#   1403#   3075#   3079#   3088#   3092#   3102#   3106#   3117#   3124#   3128#   3138#   3145#   3149#   3159#
          3163#   3174#   3178#   3402#   3410#   3441#   3462#   3485#   3499#   3512#   3525#   3533#   3557#   3573#   3588#
          3606#   3626#   3665#   3676#   3684#   3728#   3829#   3854#   4000#   4043#   4133#   4164#   4169#   4174#   4180#
          4191#   4201#   4213#   4220#   4235#   4238#   4258#   4296#   4306#   4313#   4318#   4320#   4328#   4344#   4358#
          4360#   4363#   4371#   4392#   4414#   4447#   4492#   4501#   4507#   4517#   4537#   4546#   4575#   4590#   4616#
          4636#   4733#   4813#   4858#   4888#   4951#   4997#   5328#   5346#   5359#   5445#   5451#   5544#   5605#   5628#
          5684#   5694#   5908#   5916#   5927#   5948#   5967#   6025#   6036#   6078#   6085#   6138#   6146#   6214#   6246#
M$TSTL       1#   1403#   3075#   3079#   3088#   3092#   3102#   3106#   3117#   3124#   3128#   3138#   3145#   3149#   3159#
          3163#   3174#   3178#   3402#   3410#   3441#   3462#   3485#   3499#   3512#   3525#   3533#   3557#   3573#   3588#
          3606#   3626#   3665#   3676#   3684#   3728#   3829#   3854#   4000#   4043#   4133#   4164#   4169#   4174#   4180#
          4191#   4201#   4213#   4220#   4235#   4238#   4258#   4296#   4306#   4313#   4318#   4320#   4328#   4344#   4358#
          4360#   4363#   4371#   4392#   4414#   4447#   4492#   4501#   4507#   4517#   4537#   4546#   4575#   4590#   4616#
          4636#   4733#   4813#   4858#   4888#   4951#   4997#   5328#   5346#   5359#   5445#   5451#   5544#   5605#   5628#
          5684#   5694#   5908#   5916#   5927#   5948#   5967#   6025#   6036#   6078#   6085#   6138#   6146#   6214#   6246#
M$WORD       1#   1403#   1459#   1468    1515#   1517    3181#   3441#   3443    3444#   4220#   4222    4223#   4238#   4320#
          4363#   4384#   4406#   4507#   4517#   4519    4520#   5328#   5329    5330    5331    5346#   5347    5348    5349
          5359#   5360    5361    5362    5451#   5452    5453    5454    5544#   5546    5547#   5694#   5695    5696    5697
          5908#   5909    5910    5911    5927#   5928    5929    5930    5948#   5949    5950    5951    5967#   5968    5969
          5970    6036#   6037    6038    6039    6078#   6079    6080    6081    6138#   6139    6140    6141    6214#   6215
          6216    6217    6270#   6280#   6285#   6290#   6298#   6343    6344
M$XFER       1#   1403#
NODCL     1409#   2214    2215    2216    2217    2218    2219    2220    2221    2222    2223    2224    2225    2226    2227
          2228    2229    2230    2231    2232    2236    2237    2238    2239    2240    2241    2242    2243    2244    2245
          2246    2247    2248    2249    2250    2251    2252    2253    2257    2258    2259    2260    2261    2266    2267
          2268    2269    2270    2273    2274    2275    2276    2277    2278    2279    2280    2283    2284    2285    2286
          2287    2288    2294    2295    2296    2297    2300    2301    2302    2303    2306    2309    2310    2311    2312
          2313    2314    2315    2316    2317    2318    2319    2320    2321    2322    2325    2326    2327    2328    2329
          2330    2331    2334    2335    2336    2337    2338    2339    2342    2343    2344    2347    2350    2351    2352
          2353    2354    2355    2356    2357    2360    2361    2362    2367
OPEN         1#   1403#
POINTE       1#   1403#   1415
PRINTB       1#   1403#   3066    3083    3096    3110    3119    3132    3140    3153    3167
PRINTF       1#   1403#   3398    3406    3660    3670    3679    3724    3996    4039    4231    4443    4488    4533    4542
          4570    4585    4611    4631    4728    4807    4854    4884    4947    4993    5441    5601    5624
PRINTS       1#   1403#   3458    3481    3491    3506    3520    3527    3550    3566    3581    3602    3622    3821    3847
PRINTX       1#   1403#
READBU       1#   1403#   4212
READEF       1#   1403#   4162    4167    4172    4178
RFLAGS       1#   1403#
SETPRI       1#   1403#   4316    4356
SETVEC       1#   1403#   4291    4301    4308
SLASH        1#   1403#
STARS        1#   1403#
SVC          1#   1403#
XFER         1#   1403#   3181#   4238#   4320#   4363#   4384#   4406#   4507#
```

```
XFERF        1#    1403#
XFERT        1#    1403#

. ABS.  036346     000

 ERRORS DETECTED:  0

 CZCLKA/I,CZCLKA.SEQ/CRF/SOL=SVC34R.MLB,CZCLKA.P11
 RUN-TIME: 21 27 3 SECONDS
 RUN-TIME RATIO: 68/52=1.3
 CORE USED:  19K  (37 PAGES)
```