VT62

VT62/DUP11 SELF-TEST/LOOPBACKEP-DZDPG-A-DL

MD-11-DZDPG-A

COPYRIGHT © 1977

FICHE 1 OF 1

JAN 1978

digital

MADE IN USA

```
000000                    .REPT 0


                                        IDENTIFICATION


                          PRODUCT CODE          MAINDEC-11-DZDPG-A-D

                          PRODUCT NAME          VT62/DUP11 SELFTEST/LOOPBACK

                          DATE                  1-SEP-77

                          MAINTAINER            DIAGNOSTIC ENGINEERING

                          AUTHOR                MICHAEL DENSMORE
```

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 WITHIN

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1977 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

REVISION HISTORY

REVISION	BY	
A	M DENSMORE	ORIGINAL RELEASE

TABLE OF CONTENTS

1 GENERAL	PG	4
1 1 ABSTRACT	PG	4
1 2 DEFINITIONS	PG	4
1.3 DESIGN NOTES	PG	4
1.4 DDCMP	PG	5
2 EQUIPMENT REQUIREMENTS	PG	5
3 LOAD PROCEDURE	PG	5
4. STARTING AND RUNNING THE PROGRAM	PG	5
4.1 START	PG	6
4.2 RESTART	PG	8
4.3 CONTINUE	PG	8
4.4 ZERO FLAGS	PG	9
4.5 ADD UNITS	PG	9
4.6 DROP UNITS	PG	9
4.7 DISPLAY PTABLE	PG.	9
4.8 SWITCH OPTIONS	PG	9
5. ERROR AND INFORMATIONAL MESSAGES	PG	10
6 RESTRICTIONS	PG.	11
7 MISC	PG	11
8 TEST DESCRIPTIONS	PG	12
8.1 MICRODIAGNOSTIC	PG	12
8 2 LOOPBACK	PG	12
9 INDEX	PG	13

```
1 0     GENERAL
  1 1   ABSTRACT
        MAINDEC-11-DZDPG RUNS THE SELFTEST FOR THE VT62  TERMINAL  AND  EXERCISES
        THE  LINK  BETWEEN  THE  TERMINAL  AND THE HOST   THIS PARTICULAR VERSION
        SUPPORTS  MULTIDROP,   FULL   DUPLEX   COMMUNICATION   VIA   THE   DUP-11
        COMMUNICATION    INTERFACE.    OTHER   VERSIONS   (MAINDEC-11-DZDZC   AND
        MAINDEC-11-DZDVP) SUPPORT THE DZ-11 AND  DV-11  COMMUNICATION  INTERFACES
        RESPECTIVELY.   THE  PROGRAM  IS  XXDP COMPATIBLE AND SUPPORTS SEQUENTIAL
        TESTING OF UP TO 8 VT62'S.  PLEASE READ ALL  SECTIONS  OF  THIS  DOCUMENT
        CAREFULLY BEFORE USING THE PROGRAM   PAY PARTICULAR ATTENTION TO SECTIONS
        3 0 THROUGH 6 0

  1 2   DEFINITIONS
        MODEM -  EQUIPMENT,  ALSO  KNOWN  AS  A  "DATASET",  WHICH  PERFORMS  THE
              MODULATION/DEMODULATION OF DATA SIGNALS   THIS UNIT PROVIDES TIMING
              FOR SYNCHRONOUS COMMUNICATION DEVICES
        MODEM ELIMINATOR - EQUIPMENT WHICH CONNECTS TWO SYNCHRONOUS DEVICES USING
              A  CABLE  RATHER  THAN TELEPHONE VOICE LINES   THE ELIMINATOR HAS A
              CLOCK FOR TIMING.
        DDCMP PROTOCOL - DEC'S COMMUNICATION LINE PROTOCOL
        UUT - UNIT UNDER TEST
        PTABLE - A TABLE OF INFORMATION, WITH ENTRIES FOR EACH UNIT  UNDER  TEST,
              THAT DESCRIBES THE HARDWARE (BUS ADDRESS, VECTOR, ETC. )
        DUP - SINGLE LINE, SYNCHRONOUS COMMUNICATION INTERFACE
        DZ - EIGHT LINE, MULTIPLEXED, SYNCHRONOUS COMMUNICATION INTERFACE
        DV - SIXTEEN LINE, MULTIPLEXED, SYNCHRONOUS COMMUNICATION INTERFACE.
        CRC CHECKSUM - CHECKSUM USED IN DEC NETWORK PROTOCOL  (CYCLIC  REDUNDANCY
              CHECK)
        MICRODIAGNOSTIC - THE MICROCODE IN THE VT62 THAT PERFORMS THE  DIAGNOSTIC
              CHECKOUT OF THE HARDWARE   (ALSO KNOWN AS "SELF TEST)
        SELF TEST - SEE "MICRODIAGNOSTIC"

  1 3   DESIGN NOTES
        THIS PROGRAM WAS DESIGNED AND IMPLEMENTED  USING  STRUCTURED  TECHNIQUES.
        THE PROCEDURE CAUSED THE PROGRAM TO HAVE SOME OPERATIONAL CHARACTERISTICS
        WHICH ARE OF INTEREST TO THE USER   THESE CHARACTERISTICS  ARE  DISCUSSED
        IN THIS SECTION   THIS DISCUSSION IS NOT VITAL TO LEARNING HOW TO OPERATE
        THE PROGRAM HOWEVER

        THE MOST IMPORTANT OPERATIONAL CHARACTERISTIC OF STRUCTURED DESIGN ,AS IT
        AFFECTS THE USER, IS HOW ERRORS ARE REPORTED   THE TOP-DOWN DESIGN OF THE
        PROGRAM RESULTS IN A BOTTOM-UP  REPORT  OF  ERROR  INFORMATION.   SEVERAL
        LINES  OF  ERROR  INFORMATION ARE PRINTED AS EACH PROGRAM SECTION REPORTS
        ITS LEVEL OF INFORMATION   AN EXAMPLE OF THIS TYPE OF ERROR LOG WOULD  BE
        A  DUP  ERROR    THE  FIRST  ERROR  MESSAGE  WOULD  GIVE  SPECIFIC DEVICE
        INFORMATION SUCH AS REGISTER CONTENTS AND THEN RETURN CONTROL TO THE NEXT
        LEVEL    AT  THE  NEXT  LEVEL, THE LOGICAL DEVICE NUMBER OR DEVICE SERIAL
        NUMBER WOULD BE REPORTED   AT THE LEVEL ABOVE THAT,  STILL  MORE  GENERAL
        INFORMATION WOULD BE GIVEN   THE REPORTS CONTINUE UNTIL THE HIGHEST LEVEL
        IS REACHED
```

COMMENT FIELDS IN THE LISTING WITH SQUARE BRACKETS ( ) REFER TO FUNCTIONAL BLOCKS IN THE DESIGN AND HAVE NO SIGNIFICANCE TO THE USER THEY ARE RETAINED FOR USE BY PROGRAM MAINTAINERS.

1 4 DDCMP

THIS SECTION GIVES A VERY BRIEF DESCRIPTION OF DDCMP AS USED BY THIS DIAGNOSTIC. THE DESCRIPTION IS BY NO MEANS COMPLETE; IT IS INTENDED TO HELP THE USER UNDERSTAND ERROR REPORTS.

THERE ARE THREE TYPES OF DDCMP MESSAGES: CONTROL, DATA AND BOOTSTRAP THE BOOTSTRAP MESSAGE IS NOT USED BY THIS DIAGNOSTIC. THE DATA MESSAGES CONSIST OF A 6-BYTE HEADER, A 2-BYTE CRC FOR THE HEADER, THE DATA ITSELF AND A 2-BYTE CRC FOR THE DATA. CONTROL MESSAGES CONSIST OF THE HEADER (AND ITS CRC) ONLY. ALL CONTROL MESSAGES HAVE AN "ENQ" CHARACTER AS THE FIRST BYTE ALL DATA MESSAGES ARE NUMBERED SO THAT EACH STATION ON THE LINK CAN KEEP TRACK OF WHICH MESSAGES ARE TRANSMITTED PROPERLY. ALL MESSAGES HAVE A STATION ADDRESS IN THE HEADER SO THAT MESSAGES ARE DIRECTED TO AND FROM THE PROPER STATIONS ON A MULTIDROP LINE.

THERE ARE FOUR CONTROL MESSAGES USED BY THIS DIAGNOSTIC: ACK, NAK, STRT AND STACK ACK IS AN ACKNOWLEDGEMENT THAT THE LAST MESSAGE SENT WAS PROPERLY RECEIVED NORMALLY, MESSAGES ARE ACKNOWLEDGED BY SENDING A DATA MESSAGE WITH THE NUMBER OF THE LAST MESSAGE IN THE HEADER. ACK'S ARE USED WHEN THERE IS NO DATA MESSAGE TO SEND BACK. NAK IS A NEGATIVE ACKNOWLEDGEMENT FOR THE LAST MESSAGE SENT AND INDICATES THE REASON FOR THE ERROR. STRT IS USED TO INITIATE COMMUNICATION BETWEEN TWO STATIONS. THE RECEIVING STATION SETS ITSELF UP AND ACKNOWLEDGES THE STRT WITH A STACK, OR START ACKNOWLEDGE THE STRT-STACK SEQUENCE IS USED BEFORE AND AFTER EACH TEST ALL MESSAGES ARE REQUIRED TO BE ACKNOWLEDGED WITH EITHER A DATA MESSAGE OR AN ACK

2 0 EQUIPMENT REQUIREMENTS

THIS PROGRAM REQUIRES A PDP-11 WITH A MINIMUM OF 16K CORE AND A CONSOLE TERMINAL. A CLOCK OPTION IS NOT REQUIRED, BUT ONLY ROUGH TIMING IS AVAILABLE WITHOUT ONE THE PDP-11 CONSOLE SWITCH REGISTER IS NOT USED ONE OR MORE DUP-11'S MUST BE CONNECTED TO THE TERMINALS UNDER TEST VIA COMMUNICATION LINES THROUGH EITHER MODEM'S OR MODEM ELIMINATORS THE VT62'S MUST BE IN SYNCHRONOUS MODE AND ON-LINE THE TERMINAL ADDRESS SWITCHES ON THE CONTROL BOARD MUST BE SET TO MATCH THE TERMINAL ADDRESS ASSIGNED TO THE LOGICAL UNIT (SEE SECTION 4 0)

3 0 LOADING PROCEDURE

THIS PROGRAM MAY BE LOADED FROM PAPER TAPE USING THE ABSOLUTE LOADER IT MAY ALSO BE LOADED FROM PAPER TAPE OR DISK USING XXDP PROCEDURES

4 0 STARTING AND RUNNING THE PROGRAM

THE STARTING ADDRESS FOR THE PROGRAM IS 200 THE PROGRAM WILL FIRST ASK IF A CLOCK (KW11-L OR KW11-P) IS AVAILABLE IT WILL THEN ASK IF A LINE

PRINTER IS AVAILABLE. IF A LINE PRINTER IS SPECIFIED, THE PROGRAM WILL ASK FOR ITS ADDRESS (DEFAULT = 177514). THE PROGRAM ALSO ASKS FOR MEMORY SIZE, Q-BUS OPTION AND LINE FREQUENCY. IF NEITHER CLOCK IS SPECIFIED, THE PROGRAM WILL ASK THE USER TO TYPE TWO CHARACTERS, FIVE SECONDS APART, IN ORDER TO PROVIDE ROUGH TIMING.

THE PROGRAM WILL THEN PROMPT THE USER BY TYPING A ">". NOW THE USER MAY TYPE ONE OF SEVEN COMMANDS STA(RT), RES(TART), CON(TINUE), ZFL(AG), ADD, DRO(P) OR DIS(PLAY), WHERE THE LETTERS IN PARENTHESIS NEED NOT BE TYPED IF A TYPING MISTAKE IS MADE, THE RUBOUT KEY IS USED TO DELETE THE LAST CHARACTER(S) TYPED THE PROGRAM WILL TYPE A MESSAGE IF THE COMMAND IS INVALID EACH OF THESE COMMANDS IS EXPLAINED IN THE SECTIONS BELOW. SWITCHES ARE USED TO SET SOFTWARE FLAGS SUCH AS "LOOP ON ERROR" THE FLAGS USED BY THIS PROGRAM ARE DESCRIBED IN SECTION 4 8

4 1 START

THE START COMMAND (STA) ZEROS ALL FLAGS (PRINT ALL ERROR MESSAGES AND DO NOT LOOP WITHIN TESTS) AND INITIATES THE OPERATOR DIALOGUE AT THIS POINT, THE DEVICE TABLE (PTABLE) IS EMPTY AND THE OPERATOR MUST USE THE DIALOGUE TO DESCRIBE ALL UUT'S AS SHOWN BELOW THE NUMBERS IN SQUARE BRACKETS, , RELATE THE TYPEOUT TO THE EXPLANATION BELOW AND ARE NOT TYPED IN ACTUAL OPERATION THE PROCEDURES FOR HANDLING TYPING ERRORS ARE ALSO DESCRIBED BELOW CR REFERS TO THE RETURN KEY (CARRIAGE RETURN).

DIALOGUE

```
# UNITS ? (DECIMAL)        1

DUP ADDRESS  ? (OCTAL) NNNNNN        2

DUP VECTOR  ? (OCTAL) NNN       3

DUP BUS PRIORITY  ? (OCTAL) N       4

TERMINAL ADDRESS  ? (OCTAL) N       5
  CHANGE S W  ? (Y OR N)       6

HOW MANY INITIATE TRIES BEFORE ABORT ? (OCTAL) 10     7

USE ALL ZERO'S FOR LOOPBACK DATA ? (Y OR N) Y      8
```

EXPLANATION

1 TYPE THE TOTAL NO OF UUT'S IN DECIMAL QUESTIONS 2 THROUGH 5 WILL BE REPEATED UNTIL ALL UUT'S HAVE BEEN DESCRIBED. THE USER DOES NOT HAVE TO ANSWER THESE QUESTIONS FOR EACH DUP HOWEVER, IF THERE ARE MULTIPLE UUT'S PER DUP. THE EXPLANATION FOR QUESTION 5 SHOWS HOW TO ENTER MULTIPLE UUT'S

2 TYPE THE BUS ADDRESS OF THE DUP THIS MUST BE AN EVEN OCTAL NUMBER NOT GREATER THAN 177776 THE USER MAY TYPE CR TO USE

```
                THE DEFAULT VALUE:  NNNNNN.
        3       TYPE THE VECTOR ADDRESS FOR THE DUP   THIS MUST BE  AN  EVEN
                OCTAL  NUMBER NOT GREATER THAN 376.  THE USER MAY TYPE CR TO
                USE THE DEFAULT VALUE:  NNN.
        4       TYPE THE BUS PRIORITY LEVEL OF THE  DUP    THIS  MUST  BE  A
                DIGIT  FROM  0 TO 7.  AGAIN, THE USER MAY TYPE CR TO GET THE
                DEFAULT VALUE:  N.
        5       TYPE THE TERMINAL ADDRESS FOR EACH TERMINAL ATTACHED TO  THE
                DUP    SEPARATE  EACH  ADDRESS  WITH  A  COMMA   THIS IS THE
                ADDRESS SELECTED BY THE SWITCHES ON THE  I/O  BOARD  IN  THE
                TERMINAL AND IS AN OCTAL NUMBER BETWEEN 0 AND 377   THE USER
                MAY TYPE CR TO GET THE DEFAULT VALUE.  N.
        6       IF THE USER WISHES TO CHANGE  EITHER  OF  THE  TWO  SOFTWARE
                PARAMETERS  DESCRIBED  FOR  QUESTIONS 7 OR 8, HE MUST ANSWER
                THIS QUESTION BY TYPING "Y".  IF THE USER WISHES  TO  BYPASS
                QUESTIONS 7 AND 8, HE TYPES "N".
        7       TYPE THE NUMBER OF  TIMES  THE  PROGRAM  IS  TO  ATTEMPT  TO
                INITIALIZE  A  DUP  BEFORE  ABORTING THE UNIT   THE USER MAY
                TYPE ANY OCTAL NUMBER BETWEEN 0  AND  177777    TYPE  CR  TO
                DEFAULT TO 10 (OCTAL) ATTEMPTS
        8       IF THE USER TYPES "Y", THE LOOPBACK DATA WILL  BE  255  ZERO
                BYTES.   IF  THE USER TYPES "N", THE LOOPBACK DATA WILL BE A
                PATTERN OF ALTERNATING 1 AND 0 BITS   SEE  SECTION  8 2  FOR
                DETAILS ON THE LOOPBACK TEST

                                EXAMPLE OF DIALOGUE
                ASSUME  THE  FOLLOWING  HARDWARE   CONFIGURATION     A   DUP
                (160000,300,3)  WITH  TWO  TERMINALS  (ADDRESSES 0 AND 1), A
                SECOND DUP (160010,320,3) WITH A SINGLE TERMINAL (ADDRESS 5)
                AND A THIRD DUP (170000,340,4) WITH TWO TERMINALS (ADDRESSES
                1 AND 2)   THE USER WOULD USE THE FOLLOWING DIALOGUE TO  SET
                UP THE PROGRAM
                     >STA

                     # UNITS ? (DECIMAL)  5

                     DUP ADDRESS. ? (OCTAL) 160000

                     DUP VECTOR. ? (OCTAL) 310   300

                     DUP BUS PRIORITY. ? (OCTAL) 4 3

                     TERMINAL ADDRESS  ? (OCTAL) 0  0,1

                     DUP ADDRESS. ? (OCTAL) 160000   160010

                     DUP VECTOR  ? (OCTAL) 300    320

                     DUP BUS PRIORITY  ? (OCTAL) 3

                     TERMINAL ADDRESS  ? (OCTAL  1   5
```

```
                    DUP ADDRESS  ? (OCTAL) 160010   170000

                    DUP VECTOR  ? (OCTAL) 320   340

                    DUP BUS PRIORITY  ? (OCTAL) 3  5

                    TERMINAL ADDRESS  ? (OCTAL) 5   1,2

                       CHANGE S  W.  ? (Y OR N)  N

                           (NOTE WHERE CR WAS USED TO ACCEPT THE DEFAULT)

                                 TYPING ERRORS

                (1)  THE RUBOUT KEY IS USED TO DELETE ANY CHARACTERS WHICH ARE IN
                     ERROR
                (2)  IF THE RESPONSE IS NOT WITHIN LIMITS  OR  IS  IN  THE  WRONG
                     FORMAT,  THE  PROGRAM  WILL  NOTIFY  THE  USER  AND  ASK THE
                     QUESTION AGAIN
                (3)  WHEN ENTERING MULTIPLE TERMINAL ADDRESSES FOR A SINGLE  DUP,
                     THE USER MAY NEGLECT TO TYPE ONE OR MORE ADDRESSES   IN SUCH
                     A CASE, THE USER MERELY ENTERS THE SAME DUP ADDRESS,  VECTOR
                     AND PRIORITY IN THE NEXT SERIES OF QUESTIONS AND THEN ENTERS
                     THE REMAINING TERMINAL ADDRESSES   REMEMBER,  THE SERIES  OF
                     QUESTIONS  (2  THROUGH  5)  WILL BE RE-ASKED UNTIL ALL UUT'S
                     HAVE BEEN ACCOUNTED FOR

          4 2        RESTART
                     THE RESTART COMMAND (RES) STARTS THE PROGRAM WITHOUT CLEARING THE PTABLE
                     ALL FLAGS REVERT TO THEIR INITIAL STATE UNLESS THE FLAG SWITCH IS USED TO
                     CHANGE THEM (SECTION 4 8)   THE OPERATOR DIALOGUE DESCRIBED IN  4 1  DOES
                     NOT  TAKE PLACE UNLESS REQUESTED   WHEN THIS COMMAND IS USED, THE PROGRAM
                     WILL ASK IF THE USER WISHES TO CHANGE THE HARDWARE TABLE AND IF HE WISHES
                     TO CHANGE THE SOFTWARE TABLE   IF THE USER ANSWERS YES (BY TYPING "Y") TO
                     THE HARDWARE TABLE QUESTION, THAT PORTION OF THE  DIALOGUE  DESCRIBED  IN
                     SECTION  4.1  BEGINNING  WITH QUESTION 1 ENDING WITH QUESTION 5 WILL TAKE
                     PLACE   IF THE USER ANSWERS YES TO THE SOFTWARE TABLE QUESTION,  HE  WILL
                     BE ASKED QUESTIONS 7 AND 8

                     THE MOST COMMON USE OF THIS COMMAND IS TO CHANGE  RUN-TIME  FLAGS  OR  TO
                     ADD/DROP  UNITS    THESE  FLAGS  ARE  DESCRIBED  IN  SECTION 4 8 - SWITCH
                     OPTIONS   TYPING A " C" WILL FORCE  THE  SUPERVISOR  TO  ISSUE  A  PROMPT
                     CHARACTER  (">")  SO  THAT  THE USER MAY USE THE RESTART COMMAND FOR THIS
                     PURPOSE   NOTE   IF THE USER USES THE START COMMAND AFTER A  C,  HE  WILL
                     HAVE TO REENTER ALL DEVICE INFORMATION!

          4 3        CONTINUE
                     THE CONTINUE COMMAND WILL RESTART THE  PROGRAM  AT  THE  TEST  WHICH  WAS
                     INTERRUPTED   UNLIKE  THE  RESTART  COMMAND, THIS COMMAND DOES NOT FORCE
                     FLAGS BACK TO THE INITIAL STATE   THE FLAGS WILL REMAIN AS SET UP  DURING
                     THE  LAST  START  OR  RESTART   THE  USER HAS THE SAME DIALOGUE AND FLAG
```

OPTIONS AS DESCRIBED FOR THE RESTART COMMAND.

4 4 ZERO FLAGS
THIS COMMAND WILL ZERO ALL FLAGS (SECTION 4.8) AND TYPE A PROMPT SECTION 4.8 DESCRIBES THE EFFECTS OF CLEARING ALL FLAGS, BUT ESSENTIALLY, ALL LOOPING AND HALTS WILL BE DISABLED AND ALL ERROR INFORMATION WILL BE PRINTED IN THE EVENT OF AN ERROR.

4 5 ADD UNITS
WHILE TESTING MULTIPLE UNITS, THE USER MAY WISH TO DROP A PARTICULAR UNIT FROM THE TEST PROCEDURE AND, LATER ON, BEGIN TESTING THAT UNIT AGAIN. IN THIS CASE, THE USER NEED NOT STOP THE DIAGNOSTIC AND REBUILD THE PTABLE. AFTER A UNIT HAS BEEN DROPPED (SEE SECTION 4.6), THE USER MAY ADD THE UNIT BY TYPING " C" AND USING THE ADD COMMAND AS FOLLOWS.

ADD/UNI.N

WHERE N IS A DECIMAL UNIT NUMBER THAT SPECIFIES THE UNIT TO BE ADDED UNITS ARE NUMBERED FROM 1 TO 32 ACCORDING TO THE ORDER IN WHICH THEY WERE SPECIFIED FOR THE HARDWARE TABLE MULTIPLE UNITS MAY BE ADDED BY TYPING EACH UNIT NUMBER SEPARATED BY COLONS (I.E , ADD/UNI 2 4 22) OR BY TYPING THE FIRST AND LAST UNIT NUMBERS OF A CONTIGUOUS GROUP OF UNITS SEPARATED BY A DASH (I E , ADD/UNI 1-10 WILL ADD THE FIRST TEN UNITS). NOTE AFTER A START COMMAND, ALL UNITS ARE CONSIDERED ADDED THERE IS NO NEED TO USE THE ADD COMMAND UNLESS THE UNIT WAS DROPPED

4 6 DROP UNITS
TO REMOVE A UNIT FROM TESTING, THE DROP COMMAND (DRO) IS USED TO USE THIS COMMAND, TYPE C" AND DROP THE UNIT(S) USING THE DROP COMMAND AS FOLLOWS

DRO/UNI N

WHERE "N" IS A DECIMAL UNIT NUMBER WHICH SPECIFIES THE UNIT TO BE DROPPED UNITS ARE NUMBERED FROM 1 TO 32 ACCORDING TO THE ORDER IN WHICH THEY WERE SPECIFIED FOR THE HARDWARE TABLE MULTIPLE UNITS MAY BE DROPPED BY TYPING EACH UNIT NUMBER SEPARATED BY COLONS (I E , DRO/UNI 1 3 7) OR BY TYPING THE FIRST AND LAST UNIT NUMBERS OF A CONTIGUOUS GROUP SEPARATED BY A DASH (I E , DRO/UNI 1-9 CAUSES THE FIRST NINE UNITS TO BE DROPPED) AFTER THE PROGRAM IS RESTARTED, THE SPECIFIED UNIT(S) WILL NOT BE INCLUDED FOR TESTING

4 7 DISPLAY PTABLE
WHEN THE DISPLAY COMMAND IS TYPED, THE ENTIRE CONTENTS OF THE PTABLE WILL BE PRINTED UNITS THAT HAVE BEEN DROPPED WILL BE SO MARKED THIS COMMAND IS USEFUL FOR VERIFYING THE PTABLE CONTENTS

4 8 SWITCH OPTIONS
THERE ARE THREE SWITCH OPTIONS AVAILABLE FOR USE WITH THE COMMANDS /TES , /PAS AND /FLA THE TEST SWITCH, /TES , IS USED ONLY WITH THE START COMMAND AND THE RESTART COMMAND IT IS USED TO SPECIFY THE INDIVIDUAL TEST(S) TO BE RUN IN CASES WHERE ALL TESTS ARE NOT TO BE RUN FOR EXAMPLE, IF THE USER WISHED TO RUN THE LOOPBACK TEST ONLY, HE WOULD TYPE

>STA/TES 2

IF THIS SWITCH IS NOT USED, ALL TESTS WILL BE RUN AS THE DEFAULT CONDITION.

THE PASS SWITCH IS USED TO SPECIFY THE NUMBER OF PASSES TO BE RUN TO RUN THE LOOPBACK TEST TWENTY TIMES, THE USER WOULD TYPE

```
>STA/TES 2/PAS:20
```

THIS SWITCH MAY BE USED WITH THE START, RESTART AND CONTINUE COMMANDS THE DEFAULT CONDITION IS ESSENTIALLY AN INFINITE NUMBER OF PASSES

THE FLAG SWITCH IS USED TO SET FLAGS WHICH CONTROL PROGRAM OPERATION AND ERROR PRINTOUTS THIS SWITCH MAY BE USED WITH THE CONTINUE COMMAND AS WELL AS THE START COMMAND AND RESTART COMMAND THE AVAILABLE FLAGS ARE SHOWN IN THE TABLE BELOW

FLAG	EFFECT IF SET
LOE	LOOP ON ERROR
HLT	HALT ON ERROR
IER	INHIBIT ERROR REPORTS
IBE	INHIBIT BASIC ERROR REPORT*
IXE	INHIBIT EXTENDED ERROR REPORT*
PRI	SEND ALL TYPEOUTS TO THE PRINTER
BOE	BELL ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
PNT	PRINT TEST NUMBER

*SEE SECTION 5 0

EXAMPLE
IF THE USER WISHES TO RUN THE PROGRAM WITHOUT ERROR MESSAGES AND HAVE IT LOOP ON ERROR, HE WOULD TYPE

```
>RES/FLA LOE IER
```

EXAMPLE
IF THE USER LATER WANTED TO STOP THE LOOP ON ERROR FEATURE, HE WOULD TYPE

```
>CON/FLA LOE=0
```

NOTE THE ERROR REPORTS WOULD STILL BE INHIBITED

ALL FLAGS ARE DEFAULTED TO ZERO (NOT SET)

5 0 ERROR AND INFORMATIONAL MESSAGES

AS NOTED IN THE DESIGN NOTES (SECTION 1 3), A SINGLE ERROR MAY RESULT IN SEVERAL MESSAGES BEING TYPED. THIS IS DUE TO THE STRUCTURE OF THE PROGRAM EACH SECTION, IN THE EVENT OF AN ERROR, REPORTS THE INFORMATION AVAILABLE TO IT. THE FINAL MESSAGE FOR EACH ERROR CONTAINS THE UNIT NUMBER (DECIMAL - 1 TO 32 - SEE SECTIONS 4 5 AND 4 6) AND THE PHRASE "END OF ERROR REPORT"

THERE ARE THREE LEVELS OF ERROR REPORTS BY USING THE FLAG SWITCHES (SECTION 4.8), ONE OR MORE LEVELS OF REPORT MAY BE INHIBITED THE FIRST LEVEL CONSISTS OF AN ERROR NUMBER, PC CONTENTS AND ERROR TYPE (HARD, SOFT OR FATAL) THE ERROR NUMBER CAN BE RELATED TO THE TABLE AT THE END OF

THIS SECTION. THE PC CONTENTS GIVE THE ADDRESS WHERE THE ERROR REPORT MECHANISM WAS INVOKED. THE NEXT LEVEL (BASIC ERROR REPORT) GIVES AN ENGLISH DESCRIPTION OF THE ERROR. AN EXAMPLE OF THIS TYPE OF MESSAGE IS "HEADER HAS CRC ERROR". THE FINAL LEVEL (EXTENDED ERROR REPORT) GIVES ADDITIONAL DATA CONCERNING THE ERROR, SUCH AS REGISTER CONTENTS.

TO INHIBIT SOME OR ALL ERROR MESSAGES, USE THE FLAGS DESCRIBED IN SECTION 4 8 THE INHIBIT-EXTENDED-ERROR-REPORT FLAG (IXE) INHIBITS THAT LEVEL ONLY THE INHIBIT-BASIC-ERROR-REPORT FLAG (IBE) INHIBITS BOTH BASIC AND EXTENDED ERROR REPORTS. FINALLY, THE INHIBIT-ERROR-REPORT FLAG (IER)) INHIBITS ALL TYPEOUTS

TABLE OF ERROR NUMBERS

NO	ERROR DESCRIPTION
00	CANNOT INITIALIZE DUP
01	REPLY MESSAGE FROM TERM. WAS NOT AN ENQ
02	REPLY FROM TERM. WAS NOT EXPECTED TYPE OF CONTROL MESSAGE
03	TERM. ADDRESS IN RESPONSE WAS NOT CORRECT
04	RESPONSE HAD CRC ERROR IN THE HEADER
20	DUP TRANSMIT ERROR
21	DUP RECEIVE ERROR
22	MESSAGE TO TERMINAL WAS NOT ACK'ED
23	ERROR OCCURED WHILE INITIATING DUP
24	ERROR OCCURED DURING TRANSMISSION
25	DDCMP STRT MESSAGE DID NOT GET A STACK
26	ERROR DURING RECEPTION
30	CANNOT GET CLEAR TO SEND SIGNAL
31	TIMEOUT WHILE WAITING FOR RESPONSE
40	SEE 21
41	SEE 04
42	SEE 03
43	MESSAGE DATA HAD CRC ERROR
50	ERROR OCCURED DURING SELFTEST (TEST #1)
51	ERROR OCCURED DURING LOOPBACK (TEST #2)
52	ERROR IN RECEIVED DATA
70	SELFTEST RESPONSE IN ERROR (NOT RECOGNIZED)
71	TERMINAL MEMORY ERROR
72	TERMINAL CONTROL-ROM ERROR

6 0 RESTRICTIONS

THIS DIAGNOSTIC DOES NOT ATTEMPT TO TEST THE DUP11 IF A LARGE NUMBER OF COMM ERRORS OCCUR, OR IF ATTEMPTS TO SEND OR RECIEVE DATA TO AND FROM THE TERMINAL FAIL, USE THE SPECIFIC DUP11 DIAGNOSTICS TO VERIFY THE DUP11

7 0 MISCELLANEOUS

PLEASE REPORT ALL PROGRAM PROBLEMS VIA THE AIDS REPORTING SYSTEM. REFER ALL COMMENTS REGARDING DOCUMENTATION TO THE DIAGNOSTIC ENGINEERING

TERMINAL GROUP IN MAYNARD, MASS.

8 0 TEST DESCRIPTIONS

8 1 MICRODIAGNOSTIC (TEST #1)

THE MICRODIAGNOSTIC, OR SELFTEST, IS ROM RESIDENT CODE IN THE VT62 THAT IS EXECUTED UPON COMMAND FROM THE HOST COMPUTER OR DURING POWER UP. A FULL DESCRIPTION OF THE TEST IS AVAILABLE IN THE HARDWARE MANUAL FOR THE VT62 ESSENTIALLY, THE TEST VERIFIES THE MICROPROCESSOR, CHECKS ROM MEMORY, EXERCISES RAM MEMORY AND EXERCISES THE SCREEN. THE HOST MAY REQUEST THE RESULTS OF THE TEST. IN ANY CASE, THE RESULTS ARE ALWAYS DISPLAYED ON THE LAST LINE OF THE SCREEN. THE FIRST CHARACTERS ARE A CODED ERROR REPORT FOR THE HOST COMPUTER AND THE REMAINDER OF THE LINE IS AN ENGLISH DESCRIPTION OF THE ERROR STATUS.

8 2 LOOPBACK (TEST #2)

IN THIS TEST, THE TERMINAL IS COMMANDED TO ENTER LOOPBACK MODE AND A SINGLE, 256(DECIMAL) BYTE DATA MESSAGE IS SENT TO THE TERMINAL. THIS DATA MESSAGE WILL HAVE A DDCMP PROTOCOL ENVELOPE LOOPBACK DATA IS DISPLAYED ON THE VT62'S SCREEN. WHEN THE DATA IS RETURNED, THE TERMINAL IS TAKEN OUT OF LOOPBACK MODE USING A DDCMP "STRT" SEQUENCE. THE CRC CHECKSUMS ARE VALIDATED FOR BOTH THE MESSAGE HEADER AND THE MESSAGE DATA. THE DATA IS THEN CHECKED, BYTE BY BYTE. THUS THERE ARE THREE POSSIBLE ERRORS DETECTED. NO TERMINAL RESPONSE, BAD CRC OR BAD DATA.

INTERPRETATION OF ANY ERROR MESSAGES REQUIRES SOME EDUCATED GUESSWORK BECAUSE OF THE NUMBER OF COMPONENTS IN THE COMMUNICATION LINK INTERMITTENT ERRORS WOULD GENERALLY INDICATE LINE PROBLEMS UNLESS THE LINE IS PHYSICALLY SHORT OR THE ENVIRONMENT IS RELATIVELY NOISE FREE. IF EITHER OF THESE CONDITIONS IS THE CASE OR IF THE ERROR IS NOT INTERMITTENT, THEN THE TERMINAL I/O BOARD, THE MODEM AND THE DUP ARE SUSPECT. IF THERE IS NO TERMINAL RESPONSE, RUN THE TERMINAL SELF TEST OFF-LINE THIS WILL SHOW WHETHER THE TERMINAL IS COMPLETELY "DEAD" OR THE PORTIONS OF THE COMMUNICATION LINK MENTIONED ABOVE ARE BROKEN

THERE ARE TWO DATA PATTERNS AVAILABLE FOR LOOPBACK: ALL ZERO'S AND ALTERNATING 1/0 BITS (I E , BYTES OF 252,125,252,125,.) THE ZERO PATTERN IS NOISE-SUSCEPTIBLE, BUT APPEARS ON THE SCREEN AS NULLS HOWEVER THE ALTERNATING BITS PATTERN EXERCISES THE DATA PATHS BETTER SECTION 4 0 SPECIFIES HOW TO SELECT THE PATTERN

9 0 INDEX

ACK 5, 11
ADD COMMAND 9

CONTINUE COMMAND 8, 10
CRC 4-5, 11

DDCMP 4-5
DISPLAY COMMAND 9
DROP COMMAND 9

ENQ 5, 11
ERROR NUMBER 10-11
ERROR REPORT 10-11

FLAGS 6, 8-10

LOOPBACK 7, 11-12

MULTIPLE UNITS 9
MULTIPLE UUT'S 6

NAK 5

PTABLE 4, 6, 9

RESTART COMMAND 8-10

STACK 5, 11
START COMMAND 6, 8-10
STRT 5, 11
SWITCH OPTIONS 8-9
SWITCHES 6

UNIT NUMBER 9-10

ENDR

```
 5712          002000                          =2000
 5713                                  ;
 5714                                  ;       HEADER - VT62 ACCEPTANCE TEST
 5715                                  ;
 5725  002000     104                           .ASCII  @D@
  (5)  002001     132                            ASCII  @Z@
  (5)  002002     104                           .ASCII  @D@
  (5)  002003     120                           .ASCII  @P@
  (5)  002004     107                            ASCII  @G@
  (6)  002005     000                           .BYTE   0
  (6)  002006     000                           .BYTE   0
  (5)  002007     000                            BYTE   0
  (4)  002010     101                            ASCII  @A@
  (4)  002011     060                            ASCII  @0@
  (4)  002012     001                            BYTE   C$REVISION
  (4)  002013     006                            BYTE   C$EDIT
  (4)  002014  000000                            WORD   0
  (4)  002016  000000                            WORD   0
  (4)  002020  000000                           .WORD   0
  (4)  002022  000000                            WORD   0
  (4)  002024  000000                            WORD   0
  (5)  002026  000000                            WORD   0
  (4)  002030  000000                            WORD   0
  (4)  002032  000000                            WORD   0
  (4)  002034  000000                            WORD   0
  (4)  002036  000000                            WORD   0
  (4)  002040  011450                            WORD   L$DISPATCH
  (4)  002042  011462                            WORD   L$INIT
  (4)  002044  012026                            WORD   L$CLEAN
  (4)  002046  021472                            WORD   L$HARD
  (4)  002050  021642                            WORD   L$SOFT
  (4)  002052  002202                            WORD   L$DVTYP
  (4)  002054  011454                            WORD   L$RPT
  (4)  002056  011430                            WORD   L$HW
  (4)  002060  011442                            WORD   L$SW
  (4)  002062  002220                            WORD   L$DR
  (4)  002064  002224                            WORD   L$DRST
  (4)  002066  000000                            WORD   0
  (4)  002070  011456                            WORD   L$AU
  (4)  002072  011460                            WORD   L$DU
  (4)  002074  000000                            WORD   0
  (4)  002076  021774                            WORD   L$LAST
```

```
                                ;*********************************************
                                ;        GLOBAL EQUATES                     *
                                ;*********************************************
                                ,
                                ; 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
                                ,   EF16 EF01 AVAILABLE FOR PROGRAM USE
                                ,
        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
                                .
        000020                  EF16==  16
        000017                  EF15==  15
        000016                  EF14==  14
        000015                  EF13==  13
        000014                  EF12==  12
        000013                  EF11==  11
        000012                  EF10==  10
        000011                  EF09==  9
        000010                  EF08==  8
        000007                  EF07==  7
        000006                  EF06==  6
        000005                  EF05==  5
```

```
(1)          000004                  EF04==  4
(1)          000003                  EF03==  3
(1)          000002                  EF02==  2
(1)          000001                  EF01==  1
(1)                                  ;
(1)
(1)                                  ; PRIORITY LEVEL DEFINITIONS
(1)                                  ,
(1)          000340                  PRI07== 340
(1)          000300                  PRI06== 300
(1)          000240                  PRI05== 240
(1)          000200                  PRI04== 200
(1)          000140                  PRI03== 140
(1)          000100                  PRI02== 100
(1)          000040                  PRI01== 40
(1)          000000                  PRI00== 0
5749
5750                                 ,
5751                                 ,       VT62 DIAGNOSTIC EQUATES
5752                                 ,
5753         000201                          SOH=201         ,START OF HEADER    (DDCMP)
5754         000001                          ACK=1           ,ACKNOWLEDGE          "
5755         000002                          NAK=2           ;NEGATIVE ACKNOWLEDGE  "
5756         000003                          REP=3           ,REPLY
5757         000005                          ENQ=5           ,ENQUIRY              "
5758         000006                          STRT=6          ,START                "
5759         000007                          STACK=7         ,START ACKNOWLEDGE    "
5761         000226                          SYNC=226        ;SYNCHRONIZATION CHARACTEP
5766         000220                          DLE=220         ;DATA LINK ESCAPE
5767         000024                          OSOP=24         ,LOOPBACK.
5768         000200                          PARITY=200      ,PARITY BIT
5776         000002                          TSTS=2          ,NO  OF TESTS
5782         000000                          RCVCSR=0                ,OFFSETS  RECEIVER CSR
5783         000002                          RCVBUF=2                ,         RECEIVER BUFFER REG
5784         000004                          XMTCSR=4                ,         TRANSMITTER CSR
5785         000006                          XMTBUF=6                ,         TRANSMITTER BUFFER REG
5787
5789                                                                 ,HARDWARE REGISTER BIT DEFINITIONS
5790         000400                          TSOM=400        ,TRANSMIT, START OF MESSAGE
5791         000002                          DTR=2           ,DATA TERMINAL READY
5792         000400                          MASCLR=400      ,MASTER CLEAR
5793         101226                          PARAM=101226    ,PARAM. REG. - DEC MODE, AUTO-CRC INHIB  AND SYNC
5794         000004                          RTS=4           ,REQUEST TO SEND
5795         020000                          CTS=20000       ,CLEAR TO SEND
5796         000100                          TXINTE=100      ,TRANSMIT INTERRUPT ENABLE
5797         000020                          SENDFD=20       ,SEND -- FULL DUX
5798         000120                          RXINTE=120      ,RECEIVE INTERRUPT ENABLE
5799         000020                          RCENBL=20       ,RECEIVER ENABLE
5865         177400                          HYBYTE=177400           ,MASK FOR HIGH BYTE OF A WORD
5866         000377                          LOBYTE=377              ,MASK FOR LOW BYTE OF A WORD
```

```
                                ;*****************************************
                                ;       GLOBAL DATA                     *
                                ;*****************************************
002100  000000                  R5STCK  0               ;FLAG TO INDICATE IF R5 STACK IS SET UP
002102  000000                  SCRTCH  0               ;SCRATCH LOCATIONS
002104  000000                  TEMP    0
002106  000000                  ABORT   0               ;TEST ABORT FLAG
002110  020000                  TIMLIM  20000           ;TIMEOUT LIMIT (IN 100 MICROSEC  INCREMENTS)
002112  000000                  STIME   0               ;SHORT TIMEOUT LIMIT (APPROX  20 MICROSEC  INCREM )
002114  000010                  ILIMIT  10              ;LIMIT ON DEVICE INITIALIZATION ATTEMPTS
002116  000000                  INERR   0               ;INPUT ERROR
002120  000000                  OUTERR  0               ;OUTPUT ERROR
002122  000001                  PASCNT  1
002124  000000                  EOMSG   0               ;END_OF_MESSAGES FLAG
002126  000000                  LOGDEV  0               ;LOGICAL DEVICE NUMBER
002130  000000                  EXPINP  0               ;EXPECT_INPUT FLAG
002132  000000                  OUTMSG  0               ;OUTPUT MESSAGE POINTER AND SIZE
002134  000000                          0
002136  000000                  INMSG   0               ;INPUT MESSAGE POINTER AND SIZE
002140  000000                          0
002142  000000                  OUTBUF  0               ;OUTPUT BUFFER POINTER AND SIZE
002144  000000                  OU SIZ  0
002146  000000                  INBUF   0               ;INPUT BUFFER POINTER AND SIZE
002150  000000                  INSIZ   0

002152                          MSGAP
002152  000000                  OUT     0               ;TEST MESSAGE AREA POINTER
002154  000000                  INITMA  0               ;TEMPORARY MESSAGE POINTER AND SIZE
002156  000000                  INITMS  0               ;USED FOR INITIALIZATION OF OUTPUT ROUTINES
002160  000000                  BAPNT   0               ;GENERAL BUFFER AREA POINTER AND SIZE
002162  000000                  BASIZE  0
002164  000000                  MSGNO   0               ;MESSAGE NUMBER (DDCMP)
002166  000000                  RMSGNO  0
002170  000000                  PTABLE  0                PRESENT UNIT PTABLE DATA
002172  000000                          0
002174  000000                          0
002176  000000                          0
002200                          PTEND
002200  000000                  DEVPRI  0               ;DEVICE PRIORITY

-002202  052126  031066  042054          .ASCIZ  @VT62,DUP11@
 (3)  002210  050125  030461     000
 (2)          002216                     .EVEN
002216  000001                           WORD   1
 (6)  002220  000017                     WORD   17
 (4)  002222  000004                     WORD   4
 (2)  002224  000004                     BLKW   4
```

```
                                        ;*******************************************
                                        ;       GLOBAL TEXT                        *
                                        ;*******************************************

002234  040503  020116  047516  MSG00   .ASCIZ /CAN NOT INITIALIZE COMM DEVICE/
002242  020124  047111  052111
002250  040511  044514  042532
002256  041440  046517  020115
002264  042504  044526  042503
002272     000
002273     122  050105  054514  MSG01    ASCIZ /REPLY WAS NOT AN ENQ/
002300  053440  051501  047040
002306  052117  040440  020116
002314  047105  000121
002320  047111  047503  051122  MSG02    ASCIZ /INCORRECT REPLY TYPE/
002326  041505  020124  042522
002334  046120  020131  054524
002342  042520     000
002345                          MSG42
002345     122  050105  054514  MSG03    ASCIZ /REPLY HAS INCORRECT STATION ADDRESS/
002352  044040  051501  044440
002360  041516  051117  042522
002366  052103  051440  040524
002374  044524  047117  040440
002402  042104  042522  051523
002410     000
002411                          MSG41
002411     110  040505  042504  MSG04    ASCIZ /HEADER HAS CRC ERROR/
002416  020122  040510  020123
002424  051103  020103  051105
002432  047522  000122
002436  051105  047522  020122  MSG10    ASCIZ /ERROR DURING KEYBOARD TEST  ** END OF REPORT **/
002444  052504  044522  043516
002452  045440  054505  047502
002460  051101  020104  042524
002466  052123  020040  025052
002474  042440  042116  047440
002502  020106  042522  047520
002510  052122  025040  000052
002516  040502  020104  042522  MSG11    ASCIZ /BAD RESPONSE FROM KEYBOARD
002524  050123  047117  042523
002532  043040  047522  020115
002540  042513  041131  040517
002546  042122     000
002551     124  047517  046440  MSG12    ASCIZ /TOO MANY CHARACTERS SENT FROM KEYBOARD
002556  047101  020131  044103
002564  051101  041501  042524
002572  051522  051440  047105
002600  020124  051106  046517
002606  045440  054505  047502
002614  051101  000104
002620  051124  047101  046523  MSG20    ASCIZ /TRANSMIT ERROR
002626  052111  042440  051122
002634  051117     000
002637                          MSG40
002637     122  041505  050105  MSG21    ASCIZ /RECEPTION ERROR
```

```
        002644  044524  047117  042440
        002652  051122  051117     000
  5980  002657     116  020117  041501  MSG22   ASCIZ /NO ACK RECEIVED AFTER TRANSMISSION/
        002664  020113  042522  042503
        002672  053111  042105  040440
        002700  052106  051105  052040
        002706  040522  051516  044515
        002714  051523  047511  000116
  5981  002722  051105  047522  020122  MSG23   ASCIZ /ERROR WHILE INITIALIZING DEVICE/
        002730  044127  046111  020105
        002736  047111  052111  040511
        002744  044514  044532  043516
        002752  042040  053105  041511
        002760  000105
  5982  002762  051105  047522  020122  MSG24   ASCIZ /ERROR OCCURRED DURING TRANSMISSION/
        002770  041517  052503  051122
        002776  042105  042040  051125
        003004  047111  020107  051124
        003012  047101  046523  051511
        003020  044523  047117     000
  5983  003025     116  020117  052123  MSG25.  ASCIZ /NO STACK IN RESPONSE TO STRT/
        003032  041501  020113  047111
        003040  051040  051505  047520
        003046  051516  020105  047524
        003054  051440  051124  000124
  5984  003062  051105  047522  020122  MSG26   ASCIZ /ERROR WHILE RECEIVING BUFFER/
        003070  044127  046111  020105
        003076  042522  042503  053111
        003104  047111  020107  052502
        003112  043106  051105     000
  5985  003117     103  047101  052047  MSG30   ASCIZ /CAN'T GET CLEAR TO SEND
        003124  043440  052105  041440
        003132  042514  051101  052040
        003140  020117  042523  042116
        003146     000
  5986  003147     124  046511  020105  MSG31   ASCIZ  TIME OUT WHILE WAITING FOR TERMINAL RESPONSE/
        003154  052517  020124  044127
        003162  046111  020105  040527
        003170  052111  047111  020107
        003176  047506  020122  042524
        003204  046522  047111  046101
        003212  051040  051505  047520
        003220  051516  000105
  5987  003224  040504  040524  041440  MSG43   ASCIZ /DATA CRC ERROR/
        003232  041522  042440  051122
        003240  051117     000
  5988  003243     116  044517  042523  MSG45   ASCIZ /NOISE PATTERN SUBTEST
        003250  050040  052101  042524
        003256  047122  051440  041125
        003264  042524  052123     000
  5989  003271     105  051122  051117  MSG50   ASCIZ /ERROR DURING MICRODIAGNOSTIC
        003276  042040  051125  047111
        003304  020107  044515  051103
        003312  042117  040511  047107
        003320  051517  044524  000103
  5990  003326  051105  047522  020122  MSG51   ASCIZ /ERROR DURING LOOPBACK
```

```
        003334  052504  044522  043516
        003342  046040  047517  041120
        003350  041501  000113
 5991   003354  040504  040524  042440  MSG52   ASCIZ /DATA ERROR/
        003362  051122  051117     000
 5992   003367     111  046114  043505  MSG70   ASCIZ /ILLEGAL SELF TEST RESPONSE/
        003374  046101  051440  046105
        003402  020106  042524  052123
        003410  051040  051505  047520
        003416  051516  000105
 5993   003422  042515  047515  054522  MSG71   ASCII /MEMORY ERROR   TYPE /
        003430  042440  051122  051117
        003436  020040  052040  050131
        003444  020105
 5994   003446     000     000          MTY     BYTE 0,0
 5995   003450  020115  020055  040502          ASCII /M - BANK #/
        003456  045516  021440
 5996   003462     000     000          BNO     BYTE 0,0
 5997                                            EVEN
 5998   003464  051103  046517  042440  MSG72   ASCIZ /CROM ERROR/
        003472  051122  051117     000
 5999           003500                          EVEN
 6000
 6001   003500     005                  STRTB   BYTE ENQ
 6002   003501     006                          BYTE STRT
 6003   003502     300                          BYTE 300
 6004   003503     000                          BYTE 0
 6005   003504     000                          BYTE 0
 6006   003505     000                  STRTSA  BYTE 0
 6007   003506  000000                  STRTC   WORD 0
 6008
 6009                                   ;*****************************************
 6010                                           GLOBAL ERROR REPORT              *
 6011                                   ;*****************************************
 6013   003510  016746  176400                  MOV     !LIMIT,-(SP)
  (7)   003514  012746  003536                  MOV     #FORM00,-(SP)
  (6)   003520  012746  000002                  MOV     #2,-(SP)
  (3)   003524  010600                          MOV     SP,R0
  (4)   003526  104015                          EMT     C$PNTX
  (4)   003530  062706  000006                  ADD     #6,SP
 6014   003534                          L10000
  (3)   003534  104023                          EMT     C$MSG
 6015   003536  040445  043101  042524  FORM00  ASCIZ  %AAFTER %6.A ATTEMPTS%N
        003544  020122  047445  022466
        003552  020101  052101  042524
        003560  050115  051524  047045
        003566     000
 6016           003570                          EVEN
 6017
 6019   003570                          L10301
  (3)   003570  104023                          EMT     C$MSG
 6020
 6022   003572  016746  000030                  MOV     AD2B,-(SP)
  (8)   003576  016746  000022                  MOV     AD2A,-(SP)
  (7)   003602  012746  003630                  MOV     #FORM02 -(SP)
  (6)   003606  012746  000003                  MOV     #3,-(SP)
```

```
  (3)  003612  010600                          MOV     SP,R0
  (4)  003614  104015                          EMT     C$PNTX
  (4)  003616  062706  000010                  ADD     #10,SP
 6023  003622                        L10002
  (3)  003622  104023                          EMT     C$MSG
 6024  003624  000000                  A02A    0
 6025  003626  000000                  A02B    0
 6026  003630  040445  042522  042503  FORM02   ASCIZ /%ARECEIVED %03%A -- EXPECTED %03%N/
       003636  053111  042105  022440
       003644  031517  040445  026440
       003652  020055  054105  042520
       003660  052103  042105  022440
       003666  031517  047045     000
 6027          003674                          EVEN
 6028
 6030  003674                        AREA42
 6031  003674  016746  176202                  MOV     SCRTCH,-(SP)
  (8)  003700  016746  176200                  MOV     TEMP,-(SP)
  (7)  003704  012746  003630                  MOV     #FORM02,-(SP)
  (6)  003710  012746  000003                  MOV     #3,-(SP)
  (3)  003714  010600                          MOV     SP,R0
  (4)  003716  104015                          EMT     C$PNTX
  (4)  003720  062706  000010                  ADD     #10,SP
 6032  003724                        L10003
  (3)  003724  104023                          EMT     C$MSG
 6033
 6035  003726                        AREA41
 6036  003726                        AREA43
 6037  003726  016746  000030                  MOV     A04EX -(SP)
  (8)  003732  016746  000022                  MOV     A04RC,-(SP)
  (7)  003736  012746  003764                  MOV     #FORM04,-(SP)
  (6)  003742  012746  000003                  MOV     #3,-(SP)
  (3)  003746  010600                          MOV     SP,R0
  (4)  003750  104015                          EMT     C$PNTX
  (4)  003752  062706  000010                  ADD     #10,SP
 6038  003756                        L10004
  (3)  003756  104023                          EMT     C$MSG
 6039  003760  000000                  A04RC   0
 6040  003762  000000                  A04EX   0
 6041  003764  040445  042522  042503  FORM04   ASCIZ  %ARECEIVED %06%A -- EXPECTED %06%N
       003772  053111  042105  022440
       004000  033117  040445  026440
       004006  020055  054105  042520
       004014  052103  042105  022440
       004022  033117  047045     000
 6042          004030                          EVEN
 6043
 6045  004030  016746  000030                  MOV     KEX,-(SP)
  (8)  004034  016746  000022                  MOV     KRC,-(SP)
  (7)  004040  012746  003764                  MOV     #FORM04,-(SP)
  (6)  004044  012746  000003                  MOV     #3,-(SP)
  (3)  004050  010600                          MOV     SP,R0
  (4)  004052  104015                          EMT     C$PNTX
  (4)  004054  062706  000010                  ADD     #10,SP
 6046  004060                        L10005.
  (3)  004060  104023                          EMT     C$MSG
```

```
 6047  004062  000000                    KRC     0
 6048  004064  000000                    KEX     0
 6050  004066                            L10006
  (3)  004066  104023                            EMT     C$MSG
 6052  004070                            L10007
  (3)  004070  104023                            EMT     C$MSG
 6054  004072                            AREA24
 6055  004072                            AREA40
 6056  004072  016767  176030  000064            MOV     LOGDEV,TEMP2
 6057  004100  005767  000060                    TST     TEMP2
  (9)  004104  002002                            BGE     50000$
 6058  004106  005067  000052                    CLR     TEMP2
 6059  004112                            50000$
 6060  004112  005267  000046                    INC     TEMP2
 6061  004116  016746  000042                    MOV     TEMP2,-(SP)
  (7)  004122  012746  004166                    MOV     #FORM23,-(SP)
  (6)  004126  012746  000002                    MOV     #2,-(SP)
  (3)  004132  010600                            MOV     SP,R0
  (4)  004134  104015                            EMT     C$PNTX
  (4)  004136  062706  000006                    ADD     #6,SP
 6062  004142  012746  004206                    MOV     #FORM40,-(SP)
  (6)  004146  012746  000001                    MOV     #1,-(SP)
  (3)  004152  010600                            MOV     SP,R0
  (4)  004154  104015                            EMT     C$PNTX
  (4)  004156  062706  000004                    ADD     #4,SP
 6063  004162                            L10010
  (3)  004162  104023                            EMT     C$MSG
 6064  004164  000000                    TEMP2   0
 6065  004166  040445  042504  044526  FORM23   ASCIZ   /%ADEVICE  %D3%N/
       004174  042503  020072  042045
       004202  022463  000116
 6066  004206  040445  020040  020040  FORM40   ASCIZ   /A          ***** END OF ERROR REPORT *****%N/
       004214  020040  020040  025040
       004222  025052  025052  042440
       004230  042116  047440  020106
       004236  051105  047522  020122
       004244  042522  047520  052122
       004252  025040  025052  025052
       004260  047045     000
 6067          004264                            ELEN
 6068
 6070  004264                            L10011
  (3)  004264  104023                            EMT     C$MSG
 6071
 6074  004266                            AREA31
 6075  004266  010146                            MOV     R1,-(SP)
 6076  004270  012701  002224                    MOV     #DUPREG,R1
 6077  004274  016146  000000                    MOV     RCVCSR(R1),-(SP)
  (7)  004300  012746  004420                    MOV     #FORM30,-(SP)
  (6)  004304  012746  000002                    MOV     #2,-(SP)
  (3)  004310  010600                            MOV     SP,R0
  (4)  004312  104015                            EMT     C$PNTX
  (4)  004314  062706  000006                    ADD     #6,SP
 6078  004320  016146  000002                    MOV     RCVBUF(R1),-(SP)
  (7)  004324  012746  004442                    MOV     #FRM30A,-(SP)
  (6)  004330  012746  000002                    MOV     #2,-(SP)
```

```
  (3)  004334  010600                          MOV     SP,R0
  (4)  004336  104015                          EMT     C$PNTX
  (4)  004340  062706  000006                  ADD     #6,SP
 6079  004344  016146  000004                  MOV     XMTCSR(R1),-(SP)
  (7)  004350  012746  004465                  MOV     #FRM30B,-(SP)
  (6)  004354  012746  000002                  MOV     #2,-(SP)
  (3)  004360  010600                          MOV     SP,R0
  (4)  004362  104015                          EMT     C$PNTX
  (4)  004364  062706  000006                  ADD     #6,SP
 6080  004370  016146  000006                  MOV     XMTBUF(R1),-(SP)
  (7)  004374  012746  004507                  MOV     #FRM30C,-(SP)
  (6)  004400  012746  000002                  MOV     #2,-(SP)
  (3)  004404  010600                          MOV     SP,R0
  (4)  004406  104015                          EMT     C$PNTX
  (4)  004410  062706  000006                  ADD     #6,SP
 6081  004414  012601                          MOV     (SP)+,R1
 6082  004416                        L10012
  (3)  004416  104023                          EMT     C$MSG
 6083  004420  040445  052504  020120  FORM30   ASCIZ /%ADUP RCSR  %06%N/
       004426  041522  051123  020072
       004434  047445  022466  000116
 6084  004442  040445  052504  020120  FRM30A   ASCIZ /%ADUP RBUFF  %06%N/
       004450  041122  043125  035106
       004456  022440  033117  047045
       004464     000
 6085  004465     045  042101  050125  FRM30B   ASCIZ /%ADUP TCSR  %06%N/
       004472  052040  051503  035122
       004500  022440  033117  047045
       004506     000
 6086  004507     045  042101  050125  FRM30C   ASCIZ /%ADUP TBUFF. %06%N
       004514  052040  052502  043106
       004522  020072  047445  022466
       004530  000116
 6087                                          EVEN
 6127
 6129  004532  012746  004554                  MOV     #FORM60,-(SP
  (6)  004536  012746  000001                  MOV     #1,-(SP)
  (3)  004542  010600                          MOV     SP,R0
  (4)  004544  104015                          EMT     C$PNTX
  (4)  004546  062706  000004                  ADD     #4,SP
 6130  004552                        L10013
  (3)  004552  104023                          EMT     C$MSG
 6131  004554  040445  040503  023516  FORM60   ASCIZ %ACAN'T GET INTO LOOPBACK%N
       004562  020124  042507  020124
       004570  047111  047524  046040
       004576  047517  041120  041501
       004604  022513  000116
 6132                                          EVEN
 6134  004610  012746  004702                  MOV     #FORM61,-(SP)
  (6)  004614  012746  000001                  MOV     #1,-(SP)
  (3)  004620  010600                          MOV     SP,R0
  (4)  004622  104015                          EMT     C$PNTX
  (4)  004624  062706  000004                  ADD     #4,SP
 6135  004630  016767  175272  177322          MOV     LOGDEV TEMP2
 6136  004636  005767  177322                  TST     TEMP2
  (9)  004642  002002                          BGE     5$0015
```

```
  6137  004644  005067  177314                    CLR     TEMP2
  6138  004650                          50001$
  6139  004650  005267  177310                    INC     TEMP2
  6140  004654  016746  177304                    MOV     TEMP2,-(SP)
   (7)  004660  012746  004166                    MOV     #FORM23,-(SP)
   (6)  004664  012746  000002                    MOV     #2,-(SP)
   (3)  004670  010600                            MOV     SP,R0
   (4)  004672  104015                            EMT     C$PNTX
   (4)  004674  062706  000006                    ADD     #6,SP
  6141  004700                          L10014
   (3)  004700  104023                            EMT     C$MSG
  6142  004702  040445  040503  023516  FORM61    ASCIZ /%ACAN'T GET OUT OF LOOPBACK%N/
        004710  020124  042507  020124
        004716  052517  020124  043117
        004724  046040  047517  041120
        004732  041501  022513  000116
  6143                                            EVEN
  6145  004740  016746  000074                    MOV     A52B,-(SP)
   (8)  004744  016746  000066                    MOV     A52A,-(SP)
   (7)  004750  012746  004772                    MOV     #FORM52,-(SP)
   (6)  004754  012746  000003                    MOV     #3,-(SP)
   (3)  004760  010600                            MOV     SP,R0
   (4)  004762  104015                            EMT     C$PNTX
   (4)  004764  062706  000010                    ADD     #10,SP
  6146  004770                          L10015
   (3)  004770  104023                            EMT     C$MSG
  6147  004772  040445  054105  042520  FORM52    ASCIZ /%AEXPECTED  %03%A  -RECEIVED  %03%N/
        005000  052103  042105  020072
        005006  047445  022463  020101
        005014  026440  042522  042503
        005022  053111  042105  020072
        005030  047445  022463  000116
  6148                                            EVEN
  6149  005036  000000                    A52A    0
  6150  005040  000000                    A52B    0
  6151
  6153  005042                          L10016
   (3)  005042  104023                            EMT     C$MSG
  6154
  6156  005044                          L10017
   (3)  005044  104023                            EMT     C$MSG
  6157
  6158  005046                          TEMBUF
  6159  005046  001452                            BLKW 810
  6160  010172                          TMPBUF
  6161  010172  000416                            BLKW 270
  6162  011226  000100                            BLKW 100
  6163  011426                          SSTACK                          ;STACK FOR SUBROUTINE LINKAGE
  6164
```

```
                                ;*************************************************
                                ;       DEFAULT HARDWARE P-TABLE AND SOFTWARE P-TABLE    *
                                ;*************************************************
011426  000004                          WORD   L10020-L$HW/2
011430  160010                          160010
011432  000320                          320
011434  000140                          140
011436  000000                          0
011440                          L10020
011440  000002                          WORD   L10021-L$SW/2
011442  000010                          000010
011444  000001                  PATTERN 000001
011446                          L10021

                                ;*************************************************
                                ;       DISPATCH TABLE                                   *
                                ;*************************************************
011446  000002                          WORD   TSTS
(6)     011450  017174                  WORD   T1
(6)     011452  020200                  WORD   T2
```

```
                                ;*********************************************
                                ;STUB AREAS FOR SUPERVISOR COMPATIBILITY    *
                                ;*********************************************
011454                          L10022:
(3) 011454  104025                      EMT     C$RPT

011456                          L10023
(3) 011456  104054                      EMT     C$AU

011460                          L10024:
(3) 011460  104055                      EMT     C$DU

                                ;*********************************************
                                ;       INITIALIZE CODING                   *
                                ;*********************************************
                                ;
                                ;SAVE R1
                                ;
011462  010146                          MOV     R1,-(SP)
                                ;
                                ;CHECK START EVENT FLAG TO SEE IF WE GOT HERE BECAUSE OF A START
                                ;
011464  012700  000040                  MOV     #EF.START,R0
(3) 011470  104050                      EMT     C$REFG

                                ;IF A START, THEN...
                                ;
011472  103005                          BCC     50002$

                                ;INITIALIZE THE SUBROUTINE LINKAGE STACK
                                ;
011474  012705  011426                  MOV     #SSTACK,R5
011500  005267  170374                  INC     R5STCK
011504  000414                          BR      50003$
(3) 011506                      50002$:
                                ;
                                ;IF NOT START THEN...
                                ;
                                ;CHECK FOR RESTART AND IF IT IS,
                                ;UPDATE LOGICAL DEVICE AND STAT TABLE ENTRY POINTER,
                                ;BUT DON'T EXCEED THE NUMBER OF UNITS SPECIFIED
                                ;
011506  012700  000037                  MOV     #EF.RESTART,R0
(3) 011512  104050                      EMT     C$REFG
011514  103410                          BCS     50004$
011516  005267  170404                  INC     LOGDEV
011522  026767  170400  170264          CMP     LOGDEV,L$UNIT
(9) 011530  001002                      BNE     50005$
011532  005067  170370                  CLR     LOGDEV
011536                          50005$:
011536                          50004$:
011536                          50003$:
                                ;
                                ;CHECK FOR NEW PASS . IF IT IS, FORCE LOGICAL UNIT TO ZERO
                                ;
```

```
 6475  011536  012700  000035                  MOV     #EF.NEW,R0
  (3)  011542  104050                          EMT     C$REFG
 6476  011544  103002                          BCC     50006$
 6477  011546  005067  170354                  CLR     LOGDEV
 6481  011552                          50006$:
 6482                                 ;
 6483                                 ;END OF START CODE
 6484                                 ;FORCE THE COMM DEVICE TO BE INITIALIZED
 6485                                 ;
 6486  011552  005067  003640                  CLR     INIT
 6487  011556  005067  170324                  CLR     ABORT
 6488                                 ;
 6489                                 ;SET UP THE INPUT BUFFERS QUEUE
 6490                                 ;
 6491  011562  012767  015442  003650          MOV     #RQUET,RQUE
 6492  011570  016767  003644  003640          MOV     RQUE,RQUEUE
 6493  011576  012767  015460  003652          MOV     #RBQUET,RBQUE
 6494  011604  016701  003646                  MOV     RBQUE,R1
 6495  011610  012767  005046  170264          MOV     #TEMBUF,SCRTCH
 6496  011616                          50007$:
 6497  011616  016721  170260                  MOV     SCRTCH,(R1)+
 6498  011622  062767  000416  170252          ADD     #270.,SCRTCH
 6499  011630  020127  015464                  CMP     R1,#RBQUEB
  (6)  011634  003770                          BLE     50007$
 6516                                 ;
 6517                                 ;GET THE PTABLE ENTRY FOR THIS LOGICAL DEVICE
 6518                                 ;
 6519  011636                          50010$:
 6520  011636  005201                          INC     R1
 6521  011640  016700  170262                  MOV     LOGDEV,R0
  (3)  011644  104042                          EMT     C$GPHRD
  (3)  011646  010067  170230                  MOV     R0,SCRTCH
 6522                                 ;
 6523                                 ;IF UNIT HAS BEEN DROPPED, UPDATE POINTERS ETC  AND TRY AGAIN
 6524                                 ;
 6525  011652  103411                          BCS     50011$
 6526  011654  005267  170246                  INC     LOGDEV
 6530  011660  026767  170242  170126          CMP     LOGDEV,L$UNIT
  (9)  011666  001002                          BNE     50012$
 6531  011670  005067  170232                  CLR     LOGDEV
 6535  011674                          50012$:
 6536  011674  005001                          CLR     R1
 6537  011676                          50011$:
 6538  011676  005701                          TST     R1
  (6)  011700  001756                          BEQ     50010$
 6551                                 ;
 6552                                 ;NOW GET THE PTABLE DATA FROM THE SUPERVISOR AND STOPE WITHIN THE
 6553                                 ;DIAGNOSTIC PROGRAM
 6554                                 ;
 6555  011702  012701  002170                  MOV     #PTABLE,R1
 6556  011706                          50013$:
 6557  011706  017711  170170                  MOV     @SCRTCH,(R1)
 6558  011712  062701  000002                  ADD     #2,R1
 6559  011716  062767  000002  170156          ADD     #2,SCRTCH
 6560  011724  020127  002200                  CMP     R1,#PTEND
  (6)  011730  001366                          BNE     50013$
```

```
                                        ;
                                        ;RESTORE R1 AND EXIT INIT CODE
                                        ;
011732  012601                                  MOV     (SP)+,R1
011734                                  L10025
 (3)    011734  104011                          EMT     C$INIT
6588    011736  047045  040445  052126  VMSG    .ASCIZ /%N%AVT62 ACCEPTANCE TEST   --   VERSION 10 TERMINALS%N/
        011744  031066  040440  041503
        011752  050105  040524  041516
        011760  020105  042524  052123
        011766  020040  026440  020055
        011774  020040  042526  051522
        012002  047511  020116  030061
        012010  052040  051105  044515
        012016  040516  051514  047045
        012024     000
6589            012026                          EVEN
                                        ;*****************************************
                                        ;       CLEAN-UP CODING                 *
                                        ;*****************************************
                                        ;
                                        ;MAKE SURE DEVICE IS SHUT DOWN
                                        ;
012026  010146                                  MOV     R1,-(SP)
012030  016701  170134                          MOV     PTABLE,R1
012034  052761  000400  000004                  BIS     #MASCLR,XMTCSP(R1)
012042  012601                                  MOV     (SP)+,R1
                                        ;
                                        ;UPDATE PASS COUNT
                                        ;
012044  005267  170052                          INC     PASCNT
                                        ;
                                        ;EXIT CLEAN-UP CODE
                                        ;
012050                                  L10026
 (3)    012050  104012                          EMT     C$CLEAN
```

```
                                ;*****************************************************
                                ;       GLOBAL SUBROUTINES                          *
                                ;*****************************************************
                                ;
                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;       ROUTINE TO PROCESS TERMINAL MESSAGES FOR ALL TESTS      +
                                ;       RETURNS POINTER TO MESSAGE ("OUTMSG") AND ITS SIZE      +
                                ;       IN BYTES ("OUTSIZ")                                     +
                                ;                                                               +
                                ;       SIZE MAY BE ANY NUMBER OF BYTES                         +
                                ;                                                               +
                                ;       SETS A FLAG ("EXPINP") IF RESPONSE OTHER THAN "ACK" IS  +
                                ;       TO BE EXPECTED                                          +
                                ;                                                               +
                                ;       SETS ANOTHER FLAG ("EOMSG") IF THIS IS THE LAST MESSAGE +
                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
012052                          MSG:
                                
012052  010446                          MOV     R4,-(SP)
(2)     012054  010604                  MOV     SP,R4
(3)     012056  162706  000002          SUB     #2,SP
                                ;
                                ;GET MESSAGE POINTER
                                ;
012062  017764  170064  177776          MOV     @MSGAP,A(R4)
                                ;
                                ;IF MESSAGE POINTER IS ODD,
                                ;SET END_OF_MESSAGE INDICATOR
                                ;
012070  016467  177776  170004          MOV     A(R4),SCRTCH
(6)     012076  016746  170000          MOV     SCRTCH,-(SP)
(6)     012102  042716  000001          BIC     #1,(SP)
(6)     012106  042667  167770          BIC     (SP)+,SCRTCH
012112  005767  167764                  TST     SCRTCH
(9)     012116  001405                  BEQ     50002$
012120  005267  170000                  INC     EOMSG
012124  042764  000001  177776          BIC     #1,A(R4)
012132                          50002$:
                                ;
                                ;IF MESSAGE_POINTER IS NEGATIVE,
                                ;SET EXPECT_INPUT INDICATOR
                                ;
012132  005764  177776                  TST     A(R4)
(9)     012136  002006                  BGE     50003$
012140  005267  167764                  INC     EXPINP
012144  042764  100000  177776          BIC     #100000,A(R4)
                                ;
                                ;OTHERWISE, CLEAR EXPECT_INPUT
                                ;
012152  000402                          BR      50004$
(3)     012154                  50003$:
012154  005067  167750                  CLR     EXPINP
012160                          50004$:
                                ;
                                ;SET MESSAGE_ADDRESS AND
                                ;MESSAGE_SIZE
```

```
                                        ;
012160  016467  177776  167744                  MOV     A(R4),OUTMSG
012166  062767  000002  167756                  ADD     #2,MSGAP
012174  017767  167752  167732                  MOV     @MSGAP,OUTMSG+2
                                        ;
                                        ;UPDATE MESSAGE_AREA_POINTER
                                        ;
012202  067767  167744  167742                  ADD     @MSGAP,MSGAP
012210  062767  000002  167734                  ADD     #2,MSGAP
                                        ;
                                        ;IF MESSAGE_AREA_POINTER IS ODD
                                        ;ADD 1 TO MAKE IT EVEN
                                        ;
012216  016767  167730  167656                  MOV     MSGAP,SCRTCH
012224  016746  167652                          MOV     SCRTCH,-(SP)
012230  042716  000001                          BIC     #1,(SP)
012234  042667  167642                          BIC     (SP)+,SCRTCH
012240  005767  167636                          TST     SCRTCH
012244  001402                                  BEQ     50005$
012246  005267  167700                          INC     MSGAP
012252                                  50005$:
012252                                  50000$:
012252                                  50001$:
012252  010406                                  MOV     R4,-(SP)
012254  012604                                  MOV     (SP)+,R4
012256  000207                                  RTS     PC
                                        ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                                ROUTINE TO HANDLE OUTPUT OF MESSAGES TO THE TERMINAL      +
                                        ;       UNDER TEST   MESSAGES MAY BE OF ANY LENGTH SINCE          +
                                        ;       ROUTINES USED BY THIS ROUTINE SPLIT THE MESSAGES INTO     +
                                        ;       BUFFERS OF APPROPRIATE LENGTH FOR DDCMP                   +
                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
012260                                  TEROUT:
012260  005767  003132                          TST     INIT
012264  001011                                  BNE     50002$
012266  004767  000222                          JSR     PC,XSTRT
012272  103002                                  BCC     50003$
012274  000261                                  SEC
012276  000504                                  BR      50001$
012300                                  50003$:
012300  005067  167660                          CLR     MSGNO
012304  005067  167656                          CLR     RMSGNO
012310                                  50002$:
                                        ;
                                        ;INITIALIZE THE BUFFER ROUTINES  3.2.20.4
                                        ;
012310  016767  167616  167636                  MOV     OUTMSG,INITMA
012316  012767  002132  167556                  MOV     #OUTMSG,SCRTCH
012324  062767  000002  167550                  ADD     #2,SCRTCH
012332  017767  167544  167616                  MOV     @SCRTCH,INITMS
                                                BUFFER #1,BAPNT
012340  012767  010172  167612                  MOV     #TMPBUF,BAPNT
```

```
012346  012767  000412  167606          MOV     #266.,BASIZE
                                ;
                                ;GET A BUFFER OF THE MESSAGE
                                ;
012354  162705  000004                  SUB     #2*2,R5
(3)     012360  004767  000410          JSR     PC,GETOB
(4)     012364  012567  167552          MOV     (R5)+,OUTBUF
(4)     012370  012567  167550          MOV     (R5)+,OUTSIZ
                                ;
                                ;PERFORM THE FOLLOWING LOOP WHILE
                                ;A AND ABORT ARE CLEAR AND OUTSIZ
                                ;IS GREATER THAN 0
                                ;
012374                          50004$
(6)     012374  005767  167544          TST     OUTSIZ
(9)     012400  003442                  BLE     50005$
(6)     012402  005767  167500          TST     ABORT
(9)     012406  001037                  BNE     50005$
                                ;
                                ;ADD PROTOCOL ENVELOPE AND TRANSMIT
                                ;
012410  162705  000004                  SUB     #2*2,R5
(3)     012414  010546                  MOV     R5,-(SP)
(5)     012416  016745  167522          MOV     OUTSIZ,-(R5)
(4)     012422  016745  167514          MOV     OUTBUF,-(R5)
(3)     012426  004767  000470          JSR     PC,PROBUF
(3)     012432  012605                  MOV     (SP)+,R5
(4)     012434  012567  002762          MOV     (R5)+,XBUFFER
(4)     012440  012567  002760          MOV     (R5)+,XSIZE
012444  004767  001756                  JSR     PC,XMIT
                                ;
                                ;IF TRANSMIT ERROR, REPORT IT AND SET ERRXMT
                                ;
012450  103005                          BCC     50006$
012452  104423                          TRAP    T$ERCODE
(5)     012454  000024                   WORD   20
(5)     012456  002620                   WORD   MSG20
012460  000261                          SEC
(4)     012462  000412                  BR      50001$
012464                          50006$
                                ;
                                ;GET THE NEXT BUFFER
                                ;
012464  162705  000004                  SUB     #2*2,R5
(3)     012470  004767  000300          JSR     PC,GETOB
(4)     012474  012567  167442          MOV     (R5)+,OUTBUF
(4)     012500  012567  167440          MOV     (R5)+,OUTSIZ
                                ;
                                ;END OF LOOP
                                ;
012504  000733                          BR      50004$
(3)     012506                  50005$
                                ;
                                ;IF ERROR OCCURRED  DO AN ERROR RETURN
                                ;
012506                          50000$
```

```
  (2)  012506  000241                          CLC
  (3)  012510                        50001$
  (2)  012510  000207                          RTS     PC
 6766  012512  000000                TYPE      0                ;TYPE OF MESSAGE EXPECTED
 6767
 6768
 6769
 6770
 6771                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 6772                                ;       ROUTINE TO SEND A "STRT" MESSAGE TO THE TERMINAL     +
 6773                                ;                                                            +
 6774                                ;       A "STACK" MUST BE RETURNED BY THE TERMINAL           +
 6775                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 6776  012514                        XSTRT
 6779  012514  010446                          MOV     R4,-(SP)
  (2)  012516  010604                          MOV     SP,R4
  (3)  012520  162706  000002                  SUB     #2,SP
 6780                                ;
 6781                                ;THE DDCMP STRT MESSAGE IS 10 BYTES LONG (INCLUDING CRC) AND IS IN
 6782                                ;A BUFFER STARTING AT "STRTB"  -- PUT THIS INFORMATION
 6783                                ;IN THE TRANSMIT ROUTINE'S POINTERS
 6784
 6785  012524  012767  000010  002672          MOV     #10,XSIZE
 6786  012532  012767  002500  002662          MOV     #STRTB,XBUFFER
 6787
 6788                                ;GET THE TERMINAL ADDRESS AND PUT INTO THE STRT MESSAGE
 6789
 6790  012540  012764  002170  177776          MOV     #PTABLE.A(R4)
  (6)  012546  062764  000005  177776          ADD     #5,A(R4)
 6791  012554  117467  177776  170723          MOVB    @A(R4),STRTSA
 6792
 6793                                ;COMPUTE THE CRC FOR THE STRT MESSAGE AND PUT IN THE LAST TWO BYTES
 6794                                ;OF THE BUFFER ("STRTC")
 6795
 6796  012562  162705  000002                  SUB     #1*2,R5
  (3)  012566  010546                          MOV     R5,-(SP)
  (5)  012570  012745  000006                  MOV     #6,-(R5)
  (4)  012574  016745  002622                  MOV     XBUFFER,-(R5)
  (3)  012600  004767  004014                  JSR     PC,CRC
  (3)  012604  012605                          MOV     (SP)+,R5
  (4)  012606  012567  170674                  MOV     (R5)+,STRTC
 6797
 6798                                ;TRANSMIT THE MESSAGE
 6799
 6800  012612  005067  000154                  CLR     DDSTRY
 6801  012616                        50002$
 6802  012616  004767  001604                  JSR     PC,XMIT
 6803
 6804                                ;IF AN ERROR OCCURS, REPORT IT AND EXIT WITH THE ERROR FLAG SET
 6805
 6806  012622  103005                          BCC     50003$
 6807  012624  104423                          TRAP    T$ERCODE
  (5)  012626  000024                           WORD   20
  (5)  012630  002620                           WORD   MSG20
 6808  012632  000261                          SEC
  (4)  012634  000453                          BR      50001$
```

```
 6809  012636                                50003$
 6810                                        ;
 6811                                        ;OTHERWISE, GET THE TERMINAL'S RESPONSE
 6812                                        ;
 6813  012636  010546                                MOV     R5,-(SP)
  (4)  012640  012745  000001                        MOV     #1,-(R5)
  (3)  012644  004767  003456                        JSR     PC,RCVE
  (3)  012650  012605                                MOV     (SP)+,R5
 6814                                        ;
 6815                                        ;IF RECEPTION ERROR, REPORT IT AND EXIT WITH THE ERROR FLAG SET
 6816                                        ;
 6817  012652  103003                                BCC     50004$
 6818  012654  005267  000112                        INC     DDSTRY
 6819  012660  000402                                BR      50005$
  (3)  012662                                50004$
 6820  012662  005067  000104                        CLR     DDSTRY
 6821  012666                                50005$
 6822  012666  005767  000100                        TST     DDSTRY
  (6)  012672  001404                                BEQ     50006$
  (4)  012674  026727  000072  000003                CMP     DDSTRY,#3
  (7)  012702  001345                                BNE     50002$
  (4)  012704                                50006$
 6823  012704  005767  000062                        TST     DDSTRY
  (9)  012710  001405                                BEQ     50007$
 6824  012712  104423                                TRAP    T$EPCODE
  (5)  012714  000025                                 WORD   21
  (5)  012716  002637                                 WORD   MSG21
 6825  012720  000261                                SEC
  (4)  012722  000420                                BR      50001$
 6826  012724                                50007$
 6827
 6828                                        ;OTHERWISE, SET TYPE TO "STACK" AND CHECK TO SEE THAT IT WAS THE REPLY
 6829                                        ;
 6830  012724  012767  000007  177560                MOV     #STACK,TYPE
 6831  012732  010546                                MOV     R5,-(SP)
  (4)  012734  016745  177552                        MOV     TYPE,-(R5)
  (3)  012740  004767  001072                        JSR     PC,CHKREP
  (3)  012744  012605                                MOV     (SP)+,R5
 6832
 6833                                        ;IF REPLY WASN'T A "STACK", REPORT THE ERROR AND EXIT WITH THE ERROR FLAG SET
 6834                                        ;
 6835  012746  103005                                BCC     50010$
 6836  012750  104423                                TRAP    T$EPCODE
  (5)  012752  000031                                 WORD   25
  (5)  012754  003025                                 WORD   MSG25
 6837  012756  000261                                SEC
  (4)  012760  000401                                BR      50001$
 6838  012762                                50010$
 6839
 6840                                        ;END OF XSTRT ROUTINE
 6841
 6842  012762                                50000$:
  (2)  012762  000241                                CLC
  (3)  012764                                50001$
  (3)  012764  010406                                MOV     R4,SP
  (3)  012766  012604                                MOV     (SP)+,R4
```

```
 (2)  012770  000207                          RTS     PC
 6843 012772  000000                 DDSTRY   0
 6844
 6845
 6846
 6847
 6848
 6849                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 6850                                ;        THIS ROUTINE SPLITS A MESSAGE OF ANY LENGTH INTO            +
 6851                                ;        SEGMENTS OF A SIZE SUITABLE FOR TRANSMISSION OVER A DATA    +
 6852                                ;        LINK -- RETURNS BUFFER ADDRESS, BUFFER SIZE AND END_OF_MESSAGE+
 6853                                ;        (BUFFER SIZE = 0 IS END_OF_MESSAGE)                         +
 6854                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 6855 012774                         GETCB
 6856 012774  010446                          MOV     R4,-(SP)
 (2)  012776  010604                          MOV     SP,R4
 (3)  013000  162706  000002                  SUB     #2,SP
 6859                                ;
 6860                                ;LEAVE ENOUGH ROOM FOR A DDCMP HEADER
 6861                                ;
 6862 013004  016765  167150  000000          MOV     BAPNT,A(R5)
 (6)  013012  062765  000010  000000          ADD     #10,A(R5)
 6863
 6864                                ;INIT SIZE TO ZERO
 6865                                ;
 6866 013020  005065  000002                  CLR     B(R5)
 6867
 6868                                ;SCRTCH HAS MAXIMUM DDCMP MESSAGE SIZE
 6869
 6870 013024  016767  167132  167050          MOV     BASIZE,SCRTCH
 (6)  013032  162767  000012  167042          SUB     #12,SCRTCH
 6871
 6872                                ;GET A COPY OF THE BUFFER POINTER
 6873                                ;
 6874 013040  016564  000000  177776          MOV     A(R5),C(R4)
 6875                                ;
 6876                                ;                                   REPEAT LOOP UNTIL MAX SIZE OR END OF DATA
 6877                                ;
 6878 013046                         50002$
 (6)  013046  026567  000002  167026          CMP     B(R5),SCRTCH
 (9)  013054  002017                          BGE     50003$
 (6)  013056  005767  167074                  TST     INITMS
 (9)  013062  003414                          BLE     50003$
 6879
 6880                                ;MOVE DATA INTO MESSAGE BUFFER
 6881
 6882 013064  117774  167064  177776          MOVB    @INITMA,@C(R4)
 6883 013072  005367  167060                  DEC     INITMS
 6884 013076  005267  167052                  INC     INITMA
 6885 013102  005265  000002                  INC     B(R5)
 6886 013106  005264  177776                  INC     C(R4)
 6887 013112  000755                          BR      50002$
 (3)  013114                         50003$
 6888                                ;
 6889                                                                END OF LOOP
 6890
```

```
  6891  013114                          50000$
   (3)  013114                          50001$
   (3)  013114  010406                          MOV     R4,SP
   (3)  013116  012604                          MOV     (SP)+,R4
   (2)  013120  000207                          RTS     PC
  6892
  6893
  6894
  6895
  6896
  6897                                  ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  6898                                  ;       ROUTINE TO PROVIDE LINE PROTOCOL HEADER AND CHECKSUMS     +
  6899                                  ;       CALL WITH DATA AREA ADDRESS AND SIZE (IN BYTES)           +
  6900                                  ;       RETURN WITH MODIFIED ADDRESS AND SIZE                     +
  6901                                  ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
  6902  013122                          PROBUF
  6903  013122  010146                          MOV     R1,-(SP)
  6904  013124  010246                          MOV     R2,-(SP)
  6905                                  ;
  6906                                  ;CHANGE POINTER TO INCLUDE DDCMP HEADER
  6907                                  ;
  6908  013126  016502  000000                  MOV     A(R5),R2
   (6)  013132  162702  000010                  SUB     #10,R2
  6909                                  ;
  6910                                  ;FIRST BYTE IS SOH (START OF HEADER)
  6911                                  ;
  6912  013136  112722  000201                  MOVB    #SOH,(R2)+
  6913                                  ;
  6914                                  ;NEXT TWO BYTES GET MESSAGE SIZE
  6915                                  ;
  6916  013142  016501  000002                  MOV     B(R5),R1
  6917  013146  110122                          MOVB    R1,(R2)+
  6918  013150  000301                          SWAB    R1
  6919  013152  110112                          MOVB    R1,(R2)
  6920                                  ;
  6921                                  ;SET SELECT BIT
  6922                                  ;
  6923  013154  152712  000200                  BISB    #200,(R2)
  6924  013160  005202                          INC     R2
  6925                                  ;
  6926                                  ;NEXT BYTE IS LAST GOOD RESPONSE
  6927                                  ;
  6928  013162  116722  167000                  MOVB    RMSGNO,(R2)+
  6929                                  ;
  6930                                  ;NEXT BYTE GETS MESSAGE NUMBER --- POP NUMBER FOR NEXT MESSAGE (IF ANY)
  6931                                  ;
  6932  013166  005267  166772                  INC     MSGNO
  6933  013172  116722  166766                  MOVB    MSGNO,(R2)+
  6934                                  ;
  6935                                  ;GET THE TERMINAL ADDRESS FOR THE NEXT HEADER BYTE
  6936                                  ;
  6937  013176  012701  002170                  MOV     #PTABLE,R1
   (6)  013202  062701  000005                  ADD     #5,R1
  6938  013206  111122                          MOVB    (R1),(R2)+
  6939
  6940                                  ;GET HEADER START ADDRESS AND SIZE IN BYTES
```

```
                                        ;
013210  016567  000000  166664                  MOV     A(R5),SCRTCH
 (6)    013216  162767  000010  166656          SUB     #10,SCRTCH
013224  012767  000006  166652                  MOV     #6,TEMP
                                        ;
                                        ;COMPUTE HEADER CRC
                                        ;
013232  162705  000002                          SUB     #1*2,R5
 (3)    013236  010546                          MOV     R5,-(SP)
 (5)    013240  016745  166640                  MOV     TEMP,-(R5)
 (4)    013244  016745  166632                  MOV     SCRTCH,-(R5)
 (3)    013250  004767  003344                  JSR     PC,CRC
 (3)    013254  012605                          MOV     (SP)+,R5
 (4)    013256  012567  166622                  MOV     (R5)+,TEMP
                                        ;
                                        ;PUT CRC INTO BUFFER
                                        ;
013262  016712  166616                          MOV     TEMP,(R2)
                                        ;
                                        ;GET ADDRESS OF MESSAGE AND MESSAGE SIZE IN BYTES
                                        ;
013266  016567  000000  166606                  MOV     A(R5),SCRTCH
013274  016567  000002  166602                  MOV     B(R5),TEMP
                                        ;
                                        ;COMPUTE DATA CRC
                                        ;
013302  162705  000002                          SUB     #1*2,R5
 (3)    013306  010546                          MOV     R5,-(SP)
 (5)    013310  016745  166570                  MOV     TEMP,-(R5)
 (4)    013314  016745  166562                  MOV     SCRTCH,-(R5)
 (3)    013320  004767  003274                  JSR     PC,CRC
 (3)    013324  012605                          MOV     (SP)+,R5
 (4)    013326  012567  166552                  MOV     (R5)+,TEMP
                                        ;
                                        ;PUT DATA CRC IN LAST TWO BYTES OF BUFFER
                                        ;
013332  016502  000000                          MOV     A(R5),R2
 (6)    013336  066502  000002                  ADD     B(R5),R2
013342  116722  166536                          MOVB    TEMP,(R2)+
013346  000367  166532                          SWAB    TEMP
013352  116712  166526                          MOVB    TEMP,(R2)
                                        ;
                                        ;RETURN WITH MODIFIED ADDRESS AND SIZE
                                        ;
013356  016565  000002  000006                  MOV     B(R5),D(R5)
 (6)    013364  062765  000012  000006          ADD     #12,D(R5)
013372  016565  000000  000004                  MOV     A(R5),C(R5)
 (6)    013400  162765  000010  000004          SUB     #10,C(R5)
013406  012602                                  MOV     (SP)+,R2
013410  012601                                  MOV     (SP)+,R1
013412                                  50000$
 (3)    013412                          50001$
 (2)    013412  000207                          RTS     PC
```

```
                                        ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                        ;       ROUTINE TO GET TERMINAL INPUT                       +
                                        ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
013414                                  TERIN:
                                        ;
                                        ;SAVE R1
                                        ;
013414  010146                                  MOV     R1,-(SP)
                                        ;
                                        ;GET TERMINAL RESPONSE
                                        ;
013416  010546                                  MOV     R5,-(SP)
(4)     013420  012745  000000                  MOV     #0,-(R5)
(3)     013424  004767  002676                  JSR     PC,RCVE
(3)     013430  012605                          MOV     (SP)+,R5
                                        ;
                                        ;IF RECEIVE ERROR, REPORT IT AND EXIT WITH THE ERROR FLAG SET
                                        ;
013432  103007                                  BCC     50002$
013434  104463                                  TRAP    T$ERCODE
(5)     013436  000050                           WORD   40
(5)     013440  002637                           WORD   MSG40
(5)     013442  004072                          .WORD   AREA40
                                        ;
                                        ;       RESTORE R1 FIRST
                                        ;
013444  012601                                  MOV     (SP)+,R1
013446  000261                                  SEC
(4)     013450  000571                          BR      50001$
013452                                  50002$
                                        ;
                                        ;IF MESSAGE IS ONLY 10 BYTES, SEE IF IT IS AN ACK
                                        ;IT'S OK TO BE AN ACK, BUT RECEPTION IS INCOMPLETE (CLEAR FLAG).
                                        ;
013452  026727  166472  000010                  CMP     INSIZ,#10
(9)     013460  001027                          BNE     50003$
013462  005065  000000                          CLR     A(R5)
013466  012767  000001  177016                  MOV     #ACK,TYPE
013474  010546                                  MOV     R5,-(SP)
(4)     013476  016745  177010                  MOV     TYPE,-(R5)
(3)     013502  004767  000330                  JSR     PC,CHKREP
(3)     013506  012605                          MOV     (SP)+,R5
013510  103003                                  BCC     50004$
013512  012601                                  MOV     (SP)+,R1
013514  000261                                  SEC
(4)     013516  000546                          BR      50001$
013520                                  50004$
013520  016701  166422                          MOV     INBUF,R1
(6)     013524  062701  000004                  ADD     #4,R1
013530  111167  166432                          MOVB    (R1),RMSGNO
013534  012601                                  MOV     (SP)+,R1
013536  000535                                  BR      50000$
013540                                  50003$
                                        ;
```

```
                                        ;IF RECEIVE OK, CHECK THE HEADER CRC
                                        ;COMPUTE EXPECTED CRC
                                        ;
013540  162705  000002                          SUB     #1*2,R5
013544  010546                                  MOV     R5,-(SP)
013546  012745  000006                          MOV     #6,-(R5)
013552  016745  166370                          MOV     INBUF,-(R5)
013556  004767  003036                          JSR     PC,CRC
013562  012605                                  MOV     (SP)+,R5
013564  012567  170172                          MOV     (R5)+,A04EX
                                        ,
                                        ,GET THE CRC RECEIVED
                                        ,
013570  016701  166352                          MOV     INBUF,R1
013574  062701  000010                          ADD     #10,R1
013600  114167  170154                          MOVB    -(R1),A04RC
013604  000367  170150                          SWAB    A04RC
013610  114167  170144                          MOVB    -(R1),A04RC
                                        ,
                                        ;IF THEY AREN'T THE SAME, REPORT THE ERROR, RESTORE R1 AND EXIT WITH
                                        ,THE ERROR FLAG SET
                                        ,
013614  026767  170142  170136                  CMP     A04EX,A04RC
013622  001407                                  BEQ     50005$
013624  104463                                  TRAP    T$EPCODE
013626  000051                                   WORD   41
013630  002411                                   WORD   MSG41
013632  003726                                   WORD   AREA41
013634  012601                                  MOV     (SP)+,R1
013636  000261                                  SEC
013640  000475                                  BR      50001$
013642                                  50005$
                                        ,
                                        ,COMPARE TERMINAL ADDRESS RECEIVED WITH THAT EXPECTED
                                        ,AND IF THEY AREN'T THE SAME, REPORT THE ERROR, RESTORE R1 AND
                                        ,EXIT WITH THE ERRROR FLAG SET
                                        ,
013642  124167  167637                          CMPB    -(R1),STRTSA
013646  001407                                  BEQ     50006$
013650  104463                                  TRAP    T$ERCODE
013652  000052                                  .WORD   42
013654  002345                                   WORD   MSG42
013656  003674                                   WORD   AREA42
013660  012601                                  MOV     (SP)+,R1
013662  000261                                  SEC
013664  000463                                  BR      50001$
013666                                  50006$
                                        ,
                                        ;MOVE THE POINTERS PAST THE HEADER AND COMPUTE THE DATA CRC
                                        ,
013666  062767  000010  166252                  ADD     #10,INBUF
013674  162767  000012  166246                  SUB     #12,INSIZ
013702  162705  000002                          SUB     #1*2,R5
013706  010546                                  MOV     R5,-(SP)
013710  016745  166234                          MOV     INSIZ,-(R5)
013714  016745  166226                          MOV     INBUF,-(R5)
```

```
  (3)  013720  004767  002674                    JSR     PC,CRC
  (3)  013724  012605                            MOV     (SP)+,R5
  (4)  013726  012567  170030                    MOV     (R5)+,A04EX
 7058                                  ;
 7059                                  ;GET THE DATA CRC RECEIVED
 7060                                  ;
 7061  013732  016701  166210                    MOV     INBUF,R1
  (6)  013736  066701  166206                    ADD     INSIZ,R1
 7062  013742  062701  000002                    ADD     #2,R1
 7063  013746  114167  170006                    MOVB    -(R1),A04RC
 7064  013752  000367  170002                    SWAB    A04RC
 7065  013756  114167  167776                    MOVB    -(R1),A04RC
 7066                                  ;
 7067                                  ;IF THEY AREN'T THE SAME; REPORT THE ERROR, RESTORE R1 AND
 7068                                  ;EXIT WITH THE ERROR FLAG SET
 7069                                  ;
 7070  013762  026767  167772  167772            CMP     A04RC,A04EX
  (9)  013770  001407                            BEQ     50007$
 7071  013772  104463                            TRAP    T$ERCODE
  (5)  013774  000053                            .WORD   43
  (5)  013776  003224                             WORD   MSG43
  (5)  014000  003726                            .WORD   AREA43
 7072  014002  012601                            MOV     (SP)+,R1
 7073  014004  000261                            SEC
  (4)  014006  000412                            BR      50001$
 7074  014010                                50007$
 7075                                  ;
 7076                                  ;OTHERWISE, SET COMPLETE FLAG, RESTORE R1 AND EXIT NORMALLY
 7077                                  ;
 7078  014010  005265  000000                    INC     A(R5)
 7079  014014  016701  166126                    MOV     INBUF,R1
  (6)  014020  162701  000004                    SUB     #4,R1
 7080  014024  111167  166136                    MOVB    (R1),RMSGNO
 7081  014030  012601                            MOV     (SP)+,R1
 7082  014032                                50000$
  (2)  014032  000241                            CLC
  (3)  014034                                50001$
  (2)  014034  000207                            RTS     PC
 7083
 7084
 7085
 7086
 7087
 7088
 7089                                  ++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 7090                                  ;       ROUTINE TO CHECK AN INPUT BUFFER FOR A STANDARD      +
 7091                                  ;       RESPONSE ("ACK" OR "STACK")                          +
 7092                                  ;                                                            +
 7093                                  ;       LOCATION "TYPE" CONTAINS THE RESPONSE CODE EXPECTED  +
 7094                                  ;                                                            +
 7095                                  ;       INPUT BUFFER POINTER IS IN "INBUF"                   +
 7096                                  ;                                                            +
 7097                                  ;       INPUT BUFFER SIZE (IN BYTES) IS IN "INSIZ"           +
 7098                                  ++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 7099  014036                          CHKREP
 7102  014036  010446                            MOV     R4,-(SP)
```

```
  (2)  014040  010604                          MOV     SP,R4
  (3)  014042  162706  000002                  SUB     #2,SP
 7103                                  ;
 7104                                  ;IF MESSAGE IS NOT 10 BYTES LONG, IT CAN'T BE A CONTROL MESSAGE
 7105                                  ;TYPE AN ERROR MESSAGE AND EXIT WITH ERROR SET
 7106                                  ;
 7107  014046  026727  166076  000010          CMP     INSIZ,#10
  (9)  014054  001406                          BEQ     50002$
 7108  014056  104463                          TRAP    T$ERCODE
  (5)  014060  000001                           WORD   1
  (5)  014062  002273                           WORD   MSG01
  (5)  014064  003570                           WORD   AREA01
 7109  014066  000261                          SEC
  (4)  014070  000553                          BR      50001$
 7110  014072                                 50002$:
 7111                                  ;
 7112                                  ;OTHERWISE, COMPUTE THE CRC AND CHECK IT AGAINST THAT SENT
 7113                                  ;
 7114  014072  162705  000002                  SUB     #1*2,R5
  (3)  014076  010546                          MOV     R5,-(SP)
  (5)  014100  012745  000006                  MOV     #6,-(R5)
  (4)  014104  016745  166036                  MOV     INBUF,-(R5)
  (3)  014110  004767  002504                  JSR     PC,CRC
  (3)  014114  012605                          MOV     (SP)+,R5
  (4)  014116  012567  165762                  MOV     (R5)+,TEMP
 7115  014122  016764  166020  177776          MOV     INBUF,A(R4)
  (6)  014130  062764  000007  177776          ADD     #7,A(R4)
 7116  014136  117467  177776  165736          MOVB    @A(R4),SCRTCH
 7117  014144  000367  165732                  SWAB    SCRTCH
 7118  014150  005364  177776                  DEC     A(R4)
 7119  014154  117467  177776  165720          MOVB    @A(R4),SCRTCH
 7120                                  ;
 7121                                  ;IF CRC'S MATCH GO ON . OTHERWISE SKIP FOLLOWING CODE
 7122                                  ;
 7123  014162  026767  165716  165712          CMP     TEMP,SCRTCH
  (9)  014170  001076                          BNE     50004$
 7124                                  ;
 7125                                  ;COMPARE CONTROL MESSAGE TYPE WITH THAT PASSED AS AN ARGUEMENT
 7126                                  ;
 7127  014172  016764  165750  177776          MOV     INBUF,A(R4)
  (6)  014200  005264  177776                  INC     A(R4)
 7128  014204  117464  177776  177776          MOVB    @A(R4),A(R4)
 7129  014212  042764  177400  177776          BIC     #HYBYTE,A(R4)
 7130                                  ;
 7131                                  ;IF NOT THE EXPECTED TYPE, PRINT ERROR MESSAGE AND EXIT WITH ERROR SET
 7132                                  ;
 7133  014220  026465  177776  000000          CMP     A(R4),B(R5)
  (9)  014226  001414                          BEQ     50005$
 7134  014230  016467  177776  167366          MOV     A(R4),A02A
 7135  014236  016567  000000  167362          MOV     B(R5),A02B
 7136  014244  104463                          TRAP    T$ERCODE
  (5)  014246  000002                           WORD   2
  (5)  014250  002320                           WORD   MSG02
  (5)  014252  003572                           WORD   AREA02
 7137  014254  000261                          SEC
  (4)  014256  000460                          BR      50001$
```

```
                                        ;
                                        ;OTHERWISE, CHECK TERMINAL ADDRESS
                                        ;
014260                                  50005$:
014260  012767  002170  165614                  MOV     #PTABLE,SCRTCH
 (6)    014266  062767  000005  165606          ADD     #5,SCRTCH
014274  117767  165602  165600                  MOVB    @SCRTCH,SCRTCH
014302  042767  177400  165572                  BIC     #HYBYTE,SCRTCH
014310  016767  165632  165566                  MOV     INBUF,TEMP
 (6)    014316  062767  000005  165560          ADD     #5,TEMP
014324  117767  165554  165552                  MOVB    @TEMP,TEMP
014332  042767  177400  165544                  BIC     #HYBYTE,TEMP
                                        ;
                                        ;IF NOT THE EXPECTED TERMINAL ADDRESS, THEN REPORT THE ERROR AND
                                        ;EXIT WITH ERROR SET
                                        ;
014340  026767  165540  165534                  CMP     TEMP,SCRTCH
 (9)    014346  001406                          BEQ     50007$
014350  104463                                  TRAP    T$ERCODE
 (5)    014352  000003                           WORD   3
 (5)    014354  002345                           WORD   MSG03
 (5)    014356  003674                           WORD   AREA03
014360  000261                                  SEC
 (4)    014362  000416                          BR      50001$
014364                                  50007$:
014364                                  50006$:

                                        ;IF CRC WAS IN ERROR, SKIP ABOVE CODE, COME TO HERE, REPORT ERROR AND
                                        ;EXIT WITH ERROR SET
                                        ;
014364  000414                                  BR      50010$
 (3)    014366                          50004$:
014366  016767  165510  167364                  MOV     SCRTCH,A04RC
014374  016767  165504  167360                  MOV     TEMP,A04EX
014402  104463                                  TRAP    T$ERCODE
 (5)    014404  000004                           WORD   4
 (5)    014406  002411                           WORD   MSG04
 (5)    014410  003726                           WORD   AREA04
014412  000261                                  SEC
 (4)    014414  000401                          BR      50001$
014416                                  50010$:
014416                                  50003$:
                                        ;
                                        ;NORMAL RETURN
                                        ;
014416                                  50000$:
 (2)    014416  000241                          CLC
 (3)    014420                          50001$:
 (3)    014420  010406                          MOV     R4,SP
 (3)    014422  012604                          MOV     (SP)+,R4
 (2)    014424  000207                          RTS     PC
```

```
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                ;       THIS ROUTINE SETS UP AND CHECKS FOR COMPLETE TRANSMISSION      +
                                ;       RETURNS AN ERROR FLAG IF ANY ERROR OCCURS DURING TRANSMISSION  +
                                ;       SET "ABORT" IF INITIALIZATION OF THE DEVICE FAILS              +
                                ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
014426                          XMIT:
014426  010446                          MOV     R4,-(SP)
(2)  014430  010604                     MOV     SP,R4
(3)  014432  162706  000002             SUB     #2,SP
                                ;
                                ;CLEAR ERROR COUNT, TRANSMIT DONE FLAG AND RECEIVE DONE FLAG
                                ;
014436  005064  177776                  CLR     B(R4)
014442  005067  000270                  CLR     XDONE
014446  005067  000266                  CLR     RDONE
                                ;
                                ;                       START OF LOOP
                                ;
014452                          50002$:
                                ;
                                ;TRY TO INIT DEVICE
                                ;
014452  004767  000264                  JSR     PC,DEVINIT
                                ;
                                ;IF ERROR OCCURRED...
                                ;
014456  103033                          BCC     50003$
014460  010146                          MOV     R1,-(SP)
014462  010246                          MOV     R2,-(SP)
                                ;
                                ;GET A COPY OF THE DEVICE REGISTERS
                                ;
014464  016701  165500                  MOV     PTABLE,R1
014470  012702  002224                  MOV     #DUPREG,R2
014474  016162  000000  000000          MOV     RCVCSR(R1),RCVCSR(R2)
014502  016162  000002  000002          MOV     RCVBUF(R1),RCVBUF(R2)
014510  016162  000004  000004          MOV     XMTCSR(R1),XMTCSR(R2)
014516  016162  000006  000006          MOV     XMTBUF(R1),XMTBUF(R2)
014524  012602                          MOV     (SP)+,R2
014526  012601                          MOV     (SP)+,R1
                                ;
                                ;REPORT THE ERROR AND UPDATE THE ERROR COUNT
                                ;
014530  104463                          TRAP    T$ERCODE
(5)  014532  000027                      .WORD  23
(5)  014534  002722                      .WORD  MSG23
(5)  014536  004072                      .WORD  AREA23
014540  005264  177776                  INC     B(R4)
                                ;
                                ;OTHERWISE, CLEAR THE ERROR COUNT
                                ;
014544  000402                          BR      50004$
(3)  014546                     50003$:
014546  005064  177776                  CLR     B(R4)
014552                          50004$:
                                ;
```

```
                                        ;IF ERROR COUNT HAS EXCEEDED LIMITS, ABORT AND NEGATE THE COUNT
                                        ;
014552  026467  177776  165334                   CMP     B(R4),ILIMIT
(9) 014560  001010                               BNE     50005$
014562  104462                                   TRAP    T$ERCODE
(5) 014564  000000                               .WORD   0
(5) 014566  002234                               .WORD   MSG00
(5) 014570  003510                               .WORD   AREA00
014572  005267  165310                           INC     ABORT
014576  005464  177776                           NEG     B(R4)
014602                                  50005$:
                                        ;
                                        ;                       END OF LOOP
                                        ;EXIT LOOP WHEN ERROR COUNT IS ZERO OR NEGATIVE (ABORT)
                                        ;
014602  005764  177776                           TST     B(R4)
(6) 014606  003321                               BGT     50002$
                                        ;
                                        ;IF NOT ABORT THEN EXECUTE THE FOLLOWING CODE
                                        ;OTHERWISE SKIP AND RETURN WITH ERROR SET
                                        ;
014610  005764  177776                           TST     B(R4)
(9) 014614  001042                               BNE     50006$
                                        ;
                                        ;WAIT FOR TRANSMIT DONE FLAG TO SET
                                        ;
014616                                  50007$:
014616  104022                                   EMT     C$BRK
014620  005767  000112                           TST     XDONE
(6) 014624  001774                               BEQ     50007$
                                        ;
                                        ;IF TRANSMIT FLAG IS NEGATIVE, AN ERROR OCCURRED
                                        ;
014626  005767  000104                           TST     XDONE
(9) 014632  002032                               BGE     50010$
014634  010146                                   MOV     R1,-(SP)
014636  010246                                   MOV     R2,-(SP)
                                        ;
                                        ;IF ERROR, GET COPY OF DEVICE REGISTERS, PRINT MESSAGE AND EXIT WITH ERROR SET
                                        ;OTHERWISE DO A NORMAL EXIT
                                        ;
014640  016701  165324                           MOV     PTABLE,R1
014644  012702  002224                           MOV     #DUPREG,R2
014650  016162  000000  000000                   MOV     RCVCSR(R1),RCVCSR(R2)
014656  016162  000002  000002                   MOV     RCVBUF(R1),RCVBUF(R2)
014664  016162  000004  000004                   MOV     XMTCSR(R1),XMTCSR(R2)
014672  016162  000006  000006                   MOV     XMTBUF(R1),XMTBUF(R2)
014700  012602                                   MOV     (SP)+,R2
014702  012601                                   MOV     (SP)+,R1
014704  104463                                   TRAP    T$ERCODE
(5) 014706  000030                                WORD   24
(5) 014710  002762                                WORD   MSG24
(5) 014712  004072                                WORD   AREA24
014714  000261                                   SEC
(4) 014716  000404                               BR      50001$
014720                                  50010$:
```

```
  014720  000402                          BR      50011$
  014722                          50006$:
  014722  000261                          SEC
  014724  000401                          BR      50001$
  014726                          50011$
  014726                          50000$
  014726  000241                          CLC
  014730                          50001$:
  014730  010406                          MOV     R4,SP
  014732  012604                          MOV     (SP)+,R4
  014734  000207                          RTS     PC
  014736  000000                  XDONE   0
  014740  000000                  RDONE   0


                                  ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                  ;       ROUTINE TO INITIALIZE A DEVICE                                +
                                  ;                                                                     +
                                  ;       IF "INIT" IS NON-ZERO, THE ROUTINE WILL EXECUTE AND SET IT    +
                                  ;                                                                     +
                                  ;       RETURNS WITH AN ERROR IF CTS CANNOT BE SET                    +
                                  ;                                                                     +
                                  ;       RETURNS WITH ERROR AND "ABORT" SET IF THE DEVICE DOES         +
                                  ;       NOT EXIST                                                     +
                                  ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

  014742                          DEVINIT
  014742  010446                          MOV     R4,-(SP)
  014744  010604                          MOV     SP,R4
  014746  162706  000002                  SUB     #2,SP
  014752  010146                          MOV     R1,-(SP)
                                  ;
                                  ;GET THE DEVICE ADDRESS
                                  ;
  014754  016701  165210                  MOV     PTABLE,R1
                                  ;
                                  ;DO MASTER CLEAR, SET DATA TERMINAL READY, AND SET
                                  ;PARAMETER REGISTER (DEC MODE, AUTO-CRC INHIBIT, SYNC CHARACTAR
                                  ;
  014760  052761  000400  000004          BIS     #MASCLR,XMTCSR(R1)
  014766  052761  000002  000000          BIS     #DTR,RCVCSR(R1)
  014774  052761  101226  000002          BIS     #PARAM,RCVBUF(R1)
                                  ;
                                  ;SET TIMER TO ZERO
                                  ;
  015002  005067  000402                  CLR     TIMER
                                  ;
                                  ;                               LOOP

  015006                          50002$
                                  ;
                                  ;SET REQUEST TO SEND AND WAIT 10000 MICROSECS
```

```
 015006  052711  000004                  BIS     #RTS,(R1)
 015012  012700  000100                  MOV     #100,R0
 015016  104027                          EMT     C$WTU
                                 ;
                                 ;GET THE C  R TO SEND BIT AND UPDATE THE TIMER
                                 ;
 015020  011164  177776                  MOV     (R1),@(R4)
 015024  016446  177776                  MOV     @(R4),-(SP)
 015030  042716  020000                  BIC     #CTS,(SP)
 015034  042664  177776                  BIC     (SP)+,@(R4)
 015040  005267  000344                  INC     TIMER
                                 ;
                                 ;                                       END LOOP
                                 ;   LOOP ENDS WHEN TIMER EXCEEDS LIMIT OR CLEAR TO SEND IS UP
                                 ;
 015044  026767  000340  165036          CMP     TIMER,TIMLIM
 015052  001403                          BEQ     50003$
 015054  005764  177776                  TST     @(R4)
 015060  001752                          BEQ     50002$
 015062                          50003$
                                 ;
                                 ;IF TIME LIMIT NOT EXCEEDED
                                 ;
 015062  026767  000322  165020          CMP     TIMER,TIMLIM
 015070  001507                          BEQ     50004$
                                 ;
                                 ;GET THE DEVICE PRIORITY AND SET UP THE RECEIVE INTERRUPT HANDLER
                                 ;
 015072  012767  002170  165100          MOV     #PTABLE,DEVPRI
 015100  062767  000004  165072          ADD     #4,DEVPRI
 015106  117767  165066  165064          MOVB    @DEVPRI,DEVPRI
 015114  042767  177400  165056          BIC     #HYBYTE,DEVPRI
 015122  012764  002170  177776          MOV     #PTABLE,@(R4)
 015130  062764  000002  177776          ADD     #2,@(R4)
 015136  016746  165036                  MOV     DEVPRI,-(SP)
 015142  012746  015634                  MOV     #RINT,-(SP)
 015146  017446  177776                  MOV     @@(R4),-(SP)
 015152  012746  000003                  MOV     #3,-(SP)
 015156  104037                          EMT     C$SVEC
 015160  062706  000010                  ADD     #10,SP
                                 ;
                                 ;CLEAR THE NEW MESSAGE FLAG
                                 ;
 015164  005067  000236                  CLR     NMSG
 015170  017464  177776  177776          MOV     @@(R4),@(R4)
 015176  062764  000004  177776          ADD     #4,@(R4)
                                 ;
                                 ;SET UP THE TRANSMIT INTERRUPT HANDLER
                                 ;
 015204  016746  164770                  MOV     DEVPRI,-(SP)
 015210  012746  015466                  MOV     #XINT,-(SP)
 015214  016446  177776                  MOV     @(R4),-(SP)
 015220  012746  000003                  MOV     #3,-(SP)
 015224  104037                          EMT     C$SVEC
 015226  062706  000010                  ADD     #10,SP
                                 ;
```

```
                                        ;SET UP COPIES OF THE DEVICE BUFFER REGISTER ADDRESSES
                                        ;
015232  010167  000146                          MOV     R1,XMTADD
 (6)    015236  062767  000006  000140          ADD     #6,XMTADD
015244  010167  000136                          MOV     R1,RCVADD
 (6)    015250  062767  000002  000130          ADD     #2,RCVADD
                                        ;
                                        ;SET COUNTERS FOR FOUR SYNC'S AND FOUR FILL CHARACTERS
                                        ;
015256  012767  000004  000126                  MOV     #4,SYNCNT
015264  012767  000004  000122                  MOV     #4,FLLCNT
                                        ;
                                        ;ENABLE TRANSMIT AND RECEIVE INTERRUPTS.  SET TO FULL DUPLEX
                                        ;
015272  052761  000120  000004                  BIS     #TXINTE!SENDFD,XMTCSR(R1)
015300  052761  000120  000000                  BIS     #RXINTE,RCVCSR(R1)
                                        ;
                                        ;IF TIME LIMIT WAS EXCEEDED, GET REGISTER CONTENTS, PRINT MESSAGE,
                                        ;AND RETURN WITH ERROR SET
                                        ;
015306  000427                                  BR      50005$
 (3)    015310                          50004$
015310  010246                                  MOV     R2,-(SP)
015312  012702  002224                          MOV     #DUPREG,R2
015316  016162  000000  000000                  MOV     RCVCSR(R1),RCVCSR(R2)
015324  016162  000002  000002                  MOV     RCVBUF(R1),RCVBUF(R2)
015332  016162  000004  000004                  MOV     XMTCSR(R1),XMTCSR(R2)
015340  016162  000006  000006                  MOV     XMTBUF(R1),XMTBUF(R2)
015346  012602                                  MOV     (SP)+,R2
015350  012601                                  MOV     (SP)+,R1
015352  104463                                  TRAP    T$ERCODE
 (5)    015354  000036                           WORD   30
 (5)    015356  003117                           WORD   MSG30
 (5)    015360  004266                           WORD   AREA30
015362  000261                                  SEC
 (4)    015364  000404                          BR      50001$
015366                                  50005$
                                        ;
                                        ;IF NO ERROR, DO NORMAL EXIT
                                        ;
015366  012601                                  MOV     (SP)+,R1
015370  005267  000022                          INC     INIT
015374                                  50000$
 (2)    015374  000241                          CLC
 (3)    015376                          50001$
 (3)    015376  010406                          MOV     R4,SP
 (3)    015400  012604                          MOV     (SP)+,R4
 (2)    015402  000207                          RTS     PC
                                        ;
                                        ;
015404  000000                          XMTADD  0               ;TRANSMIT BUFFER REGISTER ADDRESS
015406  000000                          RCVADD  0               ;RECEIVE BUFFER REGISTER ADDRESS
015410  000000                          TIMER   0               ;TIMER FOR CLEAR TO SEND OPERATION
015412  000000                          SYNCNT  0               ;NUMBER OF SYNC CHARACTERS INSERTED BEFORE DDCMP MESSAGE
015414  000000                          FLLCNT  0               ;NUMBER OF FILL CHARACTERS AFTER MESSAGE
015416  000000                          INIT    0               ;INITIALIZE COMPLETE FLAG
```

```
015420  000377          FIL      377              ;FILL CHARACTER
015422  000000          XBUFFER. 0                ;CURRENT TRANSMIT BUFFER POINTER
015424  000000          XSIZE    0                ;CURRENT TRANSMIT BUFFER SIZE
015426  000000          NMSG     0                ;NEW MESSAGE EXPECTED FLAG
015430  000000          RTEMP1   0                ;CURRENT RECEIVE BUFFER POINTER
015432  000000          RTEMP2   0                ;CURRENT RECEIVE BUFFER SIZE
015434  000000          CNTFLG   0                ;CONTROL FLAG FOR INPUTTING DDCMP MESSAGES
                        ;
                        ;        RECEIVER QUEUES
                        ;
015436  000000          RQUEUE   0                ;QUEUE POINTER FOR BUFFER PROCESSING
015440  000000          RQUE     0                ;QUEUE POINTER FOR NEXT AVAILABLE BUFFER
015442  000000          RQUET    0                ;QUEUE OF BUFFER ADDRESSES AND SIZES
015444  000000                   0
015446  000000                   0
015450  000000                   0
015452  000000                   0
015454  000000          RQUEB    0
015456  000000          RBQUE    0                ;QUEUE POINTER FOR NEXT AVAILABLE BUFFER AREA
015460  000000          RBQUET   0                ;QUEUE OF BUFFER AREAS
015462  000000                   0
015464  000000          RBQUEB   0
```

```
                                        ;
                                        ;       TRANSMISSION INTERRUPT HANDLER
                                        ;
                                        ;
                                        ;SET PROCESSOR PRIORITY LEVEL
                                        ;
015466  016700  164506                          MOV     DEVPRI,R0
(3)     015472  104041                          EMT     C$SPRI
                                        ;
                                        ;IF SYNC COUNT EQUAL TO ZERO
                                        ;
015474  005767  177712                          TST     SYNCNT
(9)     015500  001047                          BNE     50006$
                                        ;
                                        ;   CHECK BUFFER SIZE
                                        ;   IF GREATER THAN ZERO, THEN
                                        ;
015502  010146                                  MOV     R1,-(SP)
015504  005767  177714                          TST     XSIZE
(9)     015510  003416                          BLE     50007$
                                        ;
                                        ;       SEND THE NEXT CHARACTER AND UPDATE POINTER AND COUNT
                                        ;
015512  117701  177704                          MOVB    @XBUFFER,R1
015516  042701  177400                          BIC     #HYBYTE,R1
015522  042777  000400  177654                  BIC     #TSOM,@XMTADD
015530  110177  177650                          MOVB    R1,@XMTADD
015534  005267  177662                          INC     XBUFFER
015540  005367  177660                          DEC     XSIZE
                                        ;
                                        ;     IF ALL CHARACTERS SENT, CHECK FILL COUNT
                                        ;
015544  000423                                  BR      50010$
(3)     015546                          50007$
015546  005767  177642                          TST     FLLCNT
(9)     015552  001406                          BEQ     50011$
                                        ;
                                        ;       IF FILL COUNT ISN'T ZERO SEND A FILL CHAR  AND DECREMENT COUNT
                                        ;
015554  056777  177640  177622                  BIS     FIL,@XMTADD
015562  005367  177626                          DEC     FLLCNT
015566  000412                                  BR      50012$
(3)     015570                          50011$
                                        ;
                                        ;        IF ALL TRANSMIT DONE, DISABLE TRANSMISSION AND SET DONE FLAG
                                        ;
015570  016701  177610                          MOV     XMTADD,R1
(6)     015574  162701  000006                  SUB     #6,R1
015600  005267  177132                          INC     XDONE
015604  104021                                  EMT     C$ABRT
015606  042761  000120  000004                  BIC     #TXINTE!SENDFD,XMTCSR(R1)
015614                          50012$
015614                          50010$
015614  012601                                  MOV     (SP)+,R1

                                            OTHERWISE, SEND A SYNC AND DECREMENT THE SYNC COUNT
```

```
                                        ;
015616  000405                                  BR      50013$
 (3)    015620                          50006$
015620  052777  000626  177556                  BIS     #SYNC!TSOM,@XMTADD
015626  005367  177560                          DEC     SYNCNT
015632                                  50013$
                                        ;
                                        ;DO A SUPERVISOR-TYPE RTI
                                        ;
015632                                  L10027
 (2)    015632  000002                          RTI
                                        ;
                                        ;       RECEPTION INTERRUPT ROUTINE
                                        ;
                                        ;
                                        ;SET PROCESSOR PRIORITY
                                        ;
015634  016700  164040                          MOV     DEVPRI,R0
 (3)    015640  104041                          EMT     C$SPRI
015642  010146                                  MOV     R1,-(SP)
                                        ;
                                        ;IF NEW MESSAGE FLAG IS RESET
                                        ;
015644  005767  177556                          TST     NMSG
 (9)    015650  001111                          BNE     50014$
                                        ;
                                        ;   OBTAIN A BUFFER FROM THE QUEUE AND GET THE RECEIVED BYTE
                                        ;
015652  017767  177600  177550                  MOV     @RBQUE,RTEMP1
015660  016777  177544  177552                  MOV     RTEMP1,@RQUE
015666  117777  177514  177534                  MOVB    @RCVADD,@RTEMP1
015674  117701  177530                          MOVB    @RTEMP1,R1
                                        ;
                                        ;   CHECK FOR SYNC/FILL CHARACTER
                                        ;   (LOOKING FOR FIRST NON-SYNC/NON-FILL)
                                        ;
015700  120127  000226                          CMPB    R1,#SYNC
 (9)    015704  001472                          BEQ     50015$
 (6)    015706  120167  177506                  CMPB    R1,FIL
 (9)    015712  001467                          BEQ     50015$
                                        ;
                                        ;   NOT FILL OR SYNC. CHECK FOR ENQ (DDCMP CONTROL MESSAGE)
                                        ;
015714  120127  000005                          CMPB    R1,#ENQ
 (9)    015720  001027                          BNE     50016$
                                        ;
                                        ;   IF IT IS AN ENQ, SET EXPECTED BYTE COUNT TO 7, SET MESSAGE SIZE
                                        ;   TO 10 (OCTAL) AND UPDATE THE BUFFER QUEUES
                                        ;
015722  012767  000007  177502                  MOV     #7,RTEMP2
015730  062767  000002  177502                  ADD     #2,RQUE
```

```
015736  012777  000010  177474          MOV     #10,@RQUE
015744  062767  000002  177466          ADD     #2,RQUE
015752  026727  177462  015454          CMP     RQUE,#RQUEB
(9) 015760  003403                      BLE     50017$
015762  012767  015442  177450          MOV     #RQUET,RQUE
015770                          50017$:
                                ;
                                ;   SET COUNTER FLAG TO -1 (MESSAGE SIZE COUNT COMPLETE)
                                ;
015770  012767  177777  177436          MOV     #-1,CNTFLG
015776  000417                          BR      50020$
(3) 016000                      50016$
                                ;
                                ;   NOT FILL, SYNC OR ENQ  CHECK FOR SOH OR DLE (DDCMP DATA MESSAGE)
                                ;
016000  120127  000201                  CMPB    R1,#SOH
(8) 016004  001403                      BEQ     50021$
(6) 016006  120127  000220              CMPB    R1,#DLE
(9) 016012  001007                      BNE     50022$
(6) 016014                      50021$
                                ;
                                ;   MOVE QUEUE POINTER TO SIZE ENTRY AND SET COUNTER FLAG TO 1 (EXPECT
                                ;   FIRST COUNT BYTE)
                                ;
016014  062767  000002  177416          ADD     #2,PQUE
016022  012767  000001  177404          MOV     #1,CNTFLG
016030  000402                          BR      50023$
(3) 016032                      50022$
                                ;
                                ;   BYTE WASN'T ANY OF THE LEGAL CHARACTERS    FORCE
                                ;   A RESTART BY FORCING THE NEW MESSAGE FLAG TO RESET ON EXIT
                                ;
016032  005367  177370                  DEC     NMSG
016036                          50023$:
016036                          50020$
                                ;
                                ;   COUNT THE CHARACTER AND UPDATE THE FREE BUFFER QUEUE
                                ;
016036  005267  177366                  INC     RTEMP1
016042  062767  000002  177406          ADD     #2,RBQUE
016050  026727  177402  015464          CMP     RBQUE,#RBQUEB
(9) 016056  003403                      BLE     50024$
016060  012767  015460  177370          MOV     #RBQUET,RBQUE
016066                          50024$:
                                ;
                                ;   SET THE NEW MESSAGE FLAG
                                ;      EXIT ROUTINE
                                ;
016066  005267  177334                  INC     NMSG
016072                          50015$:
                                ;
                                ;NEW MESSAGE FLAG IS SET .  CONTINUE BUILDING MESSAGE
                                ;
016072  000513                          BR      50025$
(3) 016074                      50014$:
                                ;
```

```
                                        ;NOT A NEW MESSAGE...GET BYTE AND STORE IT
                                        ;
016074  117777  177306  177326                  MOVB    @RCVADD,@RTEMP1
                                        ;
                                        ;       IF COUNTER FLAG IS GREATER THAN ZERO, BYTE IS ALSO THE FIRST COUNT BYTE
                                        ;
016102  005767  177326                          TST     CNTFLG
 (9)    016106  003413                          BLE     50026$
                                        ;
                                        ;       PUT THE BYTE IN RTEMP2 (EXPECTED COUNT), UPDATE THE BUFFER POINTER
                                        ;       DECREMENT THE COUNTER FLAG (SET TO GET SECOND COUNT BYTE) AND
                                        ;       GO DO AN RTI
                                        ;
016110  117767  177314  177314                  MOVB    @RTEMP1,RTEMP2
016116  042767  177400  177306                  BIC     #HYBYTE,RTEMP2
016124  005267  177300                          INC     RTEMP1
016130  005367  177300                          DEC     CNTFLG
016134  000472                                  BR      50027$
 (3)    016136                          50026$
                                        ;
                                        ;       COUNTER FLAG NOT GREATER THAN ZERO, IS IT ZERO?
                                        ;
016136  005767  177272                          TST     CNTFLG
 (9)    016142  001041                          BNE     50030$
                                        ;
                                        ;       IF ZERO, BYTE IS ALSO SECOND BYTE OF COUNT (6 HIGH ORDER BITS)
                                        ;       GET IT, MASK IT AND PUT IT INTO RTEMP2 (EXPECTED COUNT)
                                        ;
016144  010146                                  MOV     R1,-(SP)
016146  117701  177256                          MOVB    @RTEMP1,R1
016152  005267  177252                          INC     RTEMP1
016156  000301                                  SWAB    R1
016160  042701  140377                          BIC     #140377,R1
016164  060167  177242                          ADD     R1,RTEMP2
016170  012601                                  MOV     (SP)+,R1
                                        ;
                                        ;       PUT THE COMPLETE SIZE IN THE QUEUE (EXPECTED COUNT +
                                        ;       12 (OCTAL) BYTES OF HEADER AND CRC)
                                        ;
016172  016777  177234  177240                  MOV     RTEMP2,@RQUE
016200  062777  000012  177232                  ADD     #12,@RQUE
                                        ;
                                        ;       ADJUST EXPECTED COUNT TO INCLUDE THE REMAINDER OF THE HEADER AND
                                        ;       THE DATA CRC
                                        ;
016206  062767  000007  177216                  ADD     #7,RTEMP2
                                        ;
                                        ;       UPDATE THE QUEUE AND DECREMENT THE COUNTER FLAG (COUNT COMPLETE)
                                        ;
016214  062767  000002  177216                  ADD     #2,RQUE
016222  026727  177212  015454                  CMP     RQUE,#RQUEB
 (9)    016230  003403                          BLE     50031$
016232  012767  015442  177200                  MOV     #RQUET,RQUE
016240                                  50031$
016240  005367  177170                          DEC     CNTFLG
                                        ;
```

```
                                        ;     GO DO AN RTI
                                        ;
016244  000426                                  BR      50032$
(3)  016246                             50030$
                                        ;
                                        ;     COUNTER FLAG IS LESS THAN ZERO    THIS IS A DATA BYTE
                                        ;      UPDATE POINTER AND COUNT
                                        ;
016246  005267  177156                          INC     RTEMP1
016252  005367  177154                          DEC     RTEMP2
                                        ;
                                        ;     IF COUNT HAS REACHED ZERO, END OF MESSAGE   SET DONE FLAG
016256  005767  177150                          TST     RTEMP2
(9)  016262  001017                             BNE     50033$
016264  005267  176450                          INC     RDONE
                                        ;
                                        ;     FORCE DUP TO RESTART RECEPTION
                                        ;
016270  016701  177112                          MOV     RCVADD,R1
(6)  016274  162701  000002                     SUB     #2,R1
016300  042761  000020  000000                  BIC     #RCENBL,RCVCSR(R1)
016306  052761  000020  000000                  BIS     #RCENBL,RCVCSR(R1)
                                        ;
                                        ;     ABORT ANY WAITS AND RESET NEW MESSAGE FLAG
                                        ;
016314  104021                                  EMT     C$ABRT
016316  005067  177104                          CLR     NMSG
016322                                  50033$
016322                                  50032$
016322                                  50027$
016322                                  50025$
                                        ;
                                        ;DO RTI
                                        ;
016322  012601                                  MOV     (SP)+,R1
016324                                  L10030
(2)  016324  000002                             RTI


                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                        ;                                                                     +
                                        ;             ROUTINE TO GET AN INPUT BUFFER FROM THE INPUT QUEUES    +
                                        ;                                                                     +
                                        ;             WILL RETURN AN ERROR IF TERMINAL DOES NOT RESPOND       +
                                        ;                                                                     +
                                        ;             ARGUEMENT (IF SET) INHIBITS ERROR PRINTOUT  THIS ALLOWS +
                                        ;             THE PROGRAM TO RETRY THE STRT-STACK SEQUENCE WITHOUT HAVING +
                                        ;             AN ERROR MESSAGE EACH TIME                              +
                                        ;                                                                     +
                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
016326                                  RCVE
                                        ;
                                        ;RESET TIMER
                                        ;
```

```
016326  016767  161446  177054          MOV     0,TIMER
                                ;
                                ;GO TO SUPERVISOR FOR AWHILE AND INCREMENT TIMER
                                ;
016334                          50002$:
016334  012700  000100                  MOV     #100,R0
016340  104027                          EMT     C$WTU
016342  005267  177042                  INC     TIMER
                                ;
                                ;LOOP UNTIL TIME LIMIT EXCEEDED OR RECEIVER DONE FLAG IS SET
                                ;
016346  026767  177036  163534          CMP     TIMER,TIMLIM
016354  003003                          BGT     50003$
016356  005767  176356                  TST     RDONE
016362  001764                          BEQ     50002$
016364                          50003$:
                                ;
                                ;RESET THE NEW MESSAGE FLAG
                                ;
016364  005067  177036                  CLR     NMSG
                                ;
                                ;IF THE TIME LIMIT WASN'T EXCEEDED
                                ;
016370  026767  177014  163512          CMP     TIMER,TIMLIM
016376  002044                          BGE     50004$
                                ;
                                ;RESET THE TIMER
                                ;CHECK FOR ERROR (RDONE LESS THAN ZERO)
                                ;
016400  005067  177004                  CLR     TIMER
016404  005767  176330                  TST     RDONE
016410  003426                          BLE     50005$
                                ;
                                ;   NOT ERRROR, GET BUFFER ADDRESS AND SIZE FROM QUEUE AND UPDATE QUEUE
                                ;
016412  017767  177020  163526          MOV     @RQUEUE,INBUF
016420  062767  000002  177010          ADD     #2,RQUEUE
016426  017767  177004  163514          MOV     @RQUEUE,INSIZ
016434  062767  000002  176774          ADD     #2,RQUEUE
016442  026727  176770  015454          CMP     RQUEUE,#RQUEB
016450  003403                          BLE     50006$
016452  012767  015442  176756          MOV     #RQUET,RQUEUE
016460                          50006$:
                                ;
                                ;RESET DONE FLAG AND LEAVE
                                ;
016460  005067  176254                  CLR     RDONE
016464  000410                          BR      50007$
016466                          50005$:
                                ;
                                ;ERROR DID OCCUR.. RESET THE DONE FLAG, REPORT ERROR, RETURN WITH ERROR
                                ;
016466  005067  176246                  CLR     RDONE
016472  104463                          TRAP    T$ERCODE
016474  000032                           WORD   26
016476  003062                           WORD   MSG26
```

```
  (5)  016500  004264                          WORD    AREA26
 7920  016502  000261                          SEC
  (4)  016504  000443                          BR      50001$
 7921  016506                          50007$:
 7922  016506  000441                          BR      50010$
  (3)  016510                          50004$:
 7923                                  ;
 7924                                  ;TIME LIMIT EXCEEDED...RESET TIMER, GET REGISTER CONTENTS
 7925                                  ;
 7926  016510  005067  176674                   CLR     TIMER
 7927  016514  010146                          MOV     R1,-(SP)
 7928  016516  016701  176664                   MOV     RCVADD,R1
  (6)  016522  162701  000002                   SUB     #2,R1
 7929  016526  010246                          MOV     R2,-(SP)
 7931  016530  012702  002224                   MOV     #DUPREG,R2
 7936  016534  016162  000000  000000           MOV     RCVCSR(R1),RCVCSR(R2)
 7937  016542  016162  000002  000002           MOV     RCVBUF(R1),RCVBUF(R2)
 7938  016550  016162  000004  000004           MOV     XMTCSR(R1),XMTCSR(R2)
 7939  016556  016162  000006  000006           MOV     XMTBUF(R1),XMTBUF(R2)
 7940  016564  012602                          MOV     (SP)+,R2
 7941  016566  012601                          MOV     (SP)+,R1
 7942                                  ;
 7943                                  ;IF NOT INHIBITED, PRINT ERROR MESSAGE
 7944                                  ;
 7945  016570  005765  000000                   TST     DDSTRT(R5)
  (9)  016574  001004                          BNE     50011$
 7946  016576  104463                          TRAP    T$ERCODE
  (5)  016600  000037                           WORD   31
  (5)  016602  003147                          .WORD   MSG31
  (5)  016604  004266                           WORD   AREA31
 7947  016606                          50011$:
 7948                                  ;
 7949                                  ;RETURN WITH ERROR IN ANY CASE
 7950                                  ;
 7951  016606  000261                          SEC
  (4)  016610  000401                          BR      50001$
 7952  016612                          50010$:
 7953  016612                          50000$:
  (2)  016612  000241                          CLC
  (3)  016614                          50001$:
  (2)  016614  000207                          RTS     PC
 7954
 7955
 7956
 7957
 7958
 7960                                  ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 7961                                  ;                                                                  +
 7962                                  ;       DUMMY ROUTINE FOR COMPATIBILITY                            +
 7963                                  ;                                                                  +
 7964                                  ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 7965  016616                          INSERR:
 7966  016616                          50000$:
  (3)  016616                          50001$:
  (2)  016616  000207                          RTS     PC
 8029
```

```
                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                        ;                                                                     +
                                        ;       THIS ROUTINE COMPUTES CRC-16 AND REQUIRES THREE ARGUMENTS     +
                                        ;       DATA ADDRESS, DATA SIZE (IN BYTES) AND A WORD LOCATION        +
                                        ;       FOR THE COMPUTED CRC                                          +
                                        ;        3 2 1 2.1, 3.2.1 2 4, 3 2.20 1 2, 3 2 20 3 1,                +
                                        ;       3 2.20 6 2, 3.2.20 6 4 AND 3 2 20 10.1                        +
                                        ;                                                                     +
                                        ;       THIS IS A STANDARD ROUTINE FROM DEC'S NETWORK GROUP           +
                                        ;                                                                     +
                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
016620                                  CRC
016620  010446                                  MOV     R4,-(SP)
016622  010604                                  MOV     SP,R4
016624  162706  000004                          SUB     #4,SP
016630  010146                                  MOV     R1,-(SP)
016632  010246                                  MOV     R2,-(SP)
016634  010346                                  MOV     R3,-(SP)
016636  016501  000000                          MOV     A(R5),R1
016642  016502  000002                          MOV     B(R5),R2
016646  005003                                  CLR     R3
016650                                  50002$
016650  005702                                  TST     R2
016652  003500                                  BLE     50003$
016654  112164  177776                          MOVB    (R1)+,D(R4)
016660  042764  177400  177776                  BIC     #HYBYTE,D(R4)
016666  005302                                  DEC     R2
016670  010346                                  MOV     R3,-(SP)
016672  046416  177776                          BIC     D(R4),(SP)
016676  040364  177776                          BIC     R3,D(R4)
016702  052664  177776                          BIS     (SP)+,D(R4)
016706  016464  177776  177774                  MOV     D(R4),E(R4)
016714  042764  177760  177776                  BIC     #177760,D(R4)
016722  006364  177776                          ASL     D(R4)
016726  062764  017074  177776                  ADD     #CRCTAB,D(R4)
016734  017464  177776  177776                  MOV     @D(R4),D(R4)
016742  042764  177417  177774                  BIC     #177417,E(R4)
016750  006264  177774                          ASR     E(R4)
016754  006264  177774                          ASR     E(R4)
016760  006264  177774                          ASR     E(R4)
016764  062764  017074  177774                  ADD     #CRCTAB,E(R4)
016772  062764  000040  177774                  ADD     #32 ,E(R4)
017000  017464  177774  177774                  MOV     @E(R4),E(R4)
017006  016446  177774                          MOV     E(R4),-(SP)
017012  046416  177776                          BIC     D(R4),(SP)
017016  046464  177774  177776                  BIC     E(R4),D(R4)
017024  052664  177776                          BIS     (SP)+,D(R4)
017030  042703  000377                          BIC     #LOBYTE,R3
017034  000303                                  SWAB    R3
017036  016446  177776                          MOV     D(R4),-(SP)
017042  040316                                  BIC     R3,(SP)
```

```
  (6)  017044  046403  177776                  BIC     D(R4),R3
  (6)  017050  052603                          BIS     (SP)+,R3
 8075  017052  000676                          BR      50002$
  (3)  017054                          50003$
 8076  017054  010365  000004                  MOV     R3,C(R5)
 8077  017060  012603                          MOV     (SP)+,R3
 8078  017062  012602                          MOV     (SP)+,R2
 8079  017064  012601                          MOV     (SP)+,R1
 8080  017066                          50000$:
  (3)  017066                          50001$.
  (3)  017066  010406                          MOV     R4,SP
  (3)  017070  012604                          MOV     (SP)+,R4
  (2)  017072  000207                          RTS     PC
 8081
 8082  017074  000000                  CRCTAB: 000000
 8083  017076  140301                          140301
 8084  017100  140601                          140601
 8085  017102  000500                          000500
 8086  017104  141401                          141401
 8087  017106  001700                          001700
 8088  017110  001200                          001200
 8089  017112  141101                          141101
 8090  017114  143001                          143001
 8091  017116  003300                          003300
 8092  017120  003600                          003600
 8093  017122  143501                          143501
 8094  017124  002400                          002400
 8095  017126  142701                          142701
 8096  017130  142201                          142201
 8097  017132  002100                          002100
 8098  017134  000000                          000000
 8099  017136  146001                          146001
 8100  017140  154001                          154001
 8101  017142  012000                          012000
 8102  017144  170001                          170001
 8103  017146  036000                          036000
 8104  017150  024000                          024000
 8105  017152  162001                          162001
 8106  017154  120001                          120001
 8107  017156  066000                          066000
 8108  017160  074000                          074000
 8109  017162  132001                          132001
 8110  017164  050000                          050000
 8111  017166  116001                          116001
 8112  017170  104001                          104001
 8113  017172  042000                          042000
```

```
                                        ;
                                        ;       HARDWARE TESTS  3 2
                                        ;
                                        ;<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
                                        ;                                                                          <>
                                        ;       SELF TEST  3.2C                                                    <>
                                        ;                                                                          <>
                                        ;<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
                                        ;
                                        ;FORCE INIT OF COMM DEVICE AND LINK
                                        ;
017174  005067  176216                          CLR     INIT
                                        ;
                                        ;GET MESSAGE ADDRESS AND SIZE
                                        ;
017200  016767  000744, 162724                  MOV     MDMSG,OUTMSG
 (6)    017206  042767  100001  162716          BIC     #100001,OUTMSG
017214  016767  000732  162712                  MOV     MDMSG+2,OUTMSG+2
                                        ;
                                        ;SEND THE MESSAGE
                                        ;
017222  004767  173032                          JSR     PC,TEROUT
                                        ;
                                        ;IF ERROR
017226  103015                                  BCC     50004$
                                        ;
                                        ;       INSERT ERROR INTO STATISTICS TABLE (MANUFACTURING VERSIONS ONLY),
                                        ;       REPORT THE ERROR, REQUEST SUPERVISOR TO LOOP
                                        ;       IF LOOPING ENABLED (EXIT TEST IF NO LOOPING)
                                        ;
017230  010546                                  MOV     R5,-(SP)
 (4)    017232  012745  000000                  MOV     #0,-(R5)
 (3)    017236  004767  177354                  JSR     PC,INSERR
 (3)    017242  012605                          MOV     (SP)+,R5
017244  104463                                  TRAP    T$ERCODE
 (5)    017246  000062                           WORD   50
 (5)    017250  003271                           WORD   MSG50
 (5)    017252  004072                           WORD   AREA40
017254  104006                                  EMT     C$CLP1
017256  104032                                  EMT     C$EXIT
 (3)    017260  000364                           WORD   L10031-
017262                                  50004$
                                        ;
                                        ;LOOK FOR ACK
                                        ;
017262  162705  000002                          SUB     #1*2,R5
 (3)    017266  004767  174122                  JSR     PC,TERIN
 (4)    017272  012567  000650                  MOV     (R5)+,MCOMP
                                        ;
                                        ;IF ERROR   INSERT ERROR INTO TABLE (MANUFACTURING TEST ONLY)
                                        ;       REPORT THE ERROR, REQUEST LOOP IF ENABLED (EXIT TEST
                                        ;       IF NO LOOPING)
                                        ;
017276  103015                                  BCC     50005$
```

```
 8165  017300  010546                          MOV     R5,-(SP)
  (4)  017302  012745  000000                  MOV     #0,-(R5)
  (3)  017306  004767  177304                  JSR     PC,INSERR
  (3)  017312  012605                          MOV     (SP)+,R5
 8166  017314  104463                          TRAP    T$ERCODE
  (5)  017316  000062                          .WORD   50
  (5)  017320  003271                          .WORD   MSG50
  (5)  017322  004072                          .WORD   AREA40
 8167  017324  104006                          EMT     C$CLP1
 8168  017326  104032                          EMT     C$EXIT
  (3)  017330  000314                          .WORD   L10031-
 8169  017332                            50005$
 8170                                  ;
 8171                                  ;IF MESSAGE WAS NOT AN ACK ..ERROR
 8172                                  ;       REPORT ERROR, LOOP IF ENABLED, EXIT TEST IF NOT
 8173                                  ;
 8174  017332  005767  000610                   TST     MCOMP
  (9)  017336  001407                          BEQ     50006$
 8175  017340  104463                          TRAP    T$ERCODE
  (5)  017342  000062                           WORD   50
  (5)  017344  003271                          .WORD   MSG50
  (5)  017346  004072                           WORD   AREA40
 8176  017350  104006                          EMT     C$CLP1
 8177  017352  104032                          EMT     C$EXIT
  (3)  017354  000270                          .WORD   L10031-.
 8178  017356                            50006$
 8179                                  ;
 8180                                  ;FORCE A RESTART ON THE LINK AND WAIT FOR SELFTEST TO COMPLETE
 8181                                  ;
 8182  017356  005067  176034                   CLR     INIT
 8183  017362  012700  000400                   MOV     #400,R0
  (3)  017366  104026                          EMT     C$WTM
 8184                                  ;
 8185                                  ;SET UP MESSAGE ASKING FOR RESULTS AND SEND IT
 8186                                  ;
 8187  017370  016767  000564  162534           MOV     MDMSG1,OUTMSG
  (6)  017376  042767  100001  162526           BIC     #100001,OUTMSG
 8188  017404  016767  000552  162522           MOV     MDMSG1+2,OUTMSG+2
 8189  017412  004767  172642                   JSR     PC,TEROUT
 8190                                  ;
 8191                                  ;IF ERROR    INSERT ERROR IN TABLE (MANUF  TESTS ONLY), REPORT ERROR,
 8192                                  ;       LOOP IF ENABLED, EXIT TEST IF NO LOOPING
 8193
 8194  017416  103015                          BCC     50007$
 8195  017420  010546                          MOV     R5,-(SP)
  (4)  017422  012745  000000                  MOV     #0,-(R5)
  (3)  017426  004767  177164                  JSR     PC,INSERR
  (3)  017432  012605                          MOV     (SP)+,R5
 8196  017434  104463                          TRAP    T$ERCODE
  (5)  017436  000062                           WORD   50
  (5)  017440  003271                           WORD   MSG50
  (5)  017442  004072                           WORD   AREA40
 8197  017444  104006                          EMT     C$CLP1
 8198  017446  104032                          EMT     C$EXIT
  (3)  017450  000174                           WORD   L10031-
 8199  017452                            50007$
```

```
                                        ;
                                        ;ACK TERMINAL AND GET ITS RESPONSE
                                        ;
017452                                  50010$
017452  004767  000170                          JSR     PC,OUTREP
                                        ;
                                        ;       HANDLE ERROR IF ONE OCCURED WHILE ACK'ING
                                        ;
017456  103023                                  BCC     50011$
017460  010546                                  MOV     R5,-(SP)
(4)     017462  012745  000000                  MOV     #0,-(R5)
(3)     017466  004767  177124                  JSR     PC,INSERR
(3)     017472  012605                          MOV     (SP)+,R5
017474  010546                                  MOV     R5,-(SP)
(4)     017476  012745  000001                  MOV     #1,-(R5)
(3)     017502  004767  177110                  JSR     PC,INSERR
(3)     017506  012605                          MOV     (SP)+,R5
017510  104463                                  TRAP    T$ERCODE
(5)     017512  000062                          .WORD   50
(5)     017514  003271                           WORD   MSG50
(5)     017516  004072                           WORD   AREA40
017520  104006                                  EMT     C$CLP1
017522  104032                                  EMT     C$EXIT
(3)     017524  000120                           WORD   L10031-
017526                                  50011$
                                        ;
                                        ;GET THE RESPONSE
                                        ;
017526  162705  000002                          SUB     #1*2,R5
(3)     017532  004767  173656                  JSR     PC,TERIN
(4)     017536  012567  000404                  MOV     (R5)+,MCOMP

                                                HANDLE ERROR IF ONE OCCURED

                                        ;
017542  103015                                  BCC     50012$
017544  010546                                  MOV     R5,-(SP)
(4)     017546  012745  000000                  MOV     #0,-(R5)
(3)     017552  004767  177040                  JSR     PC,INSERR
(3)     017556  012605                          MOV     (SP)+,R5
017560  104463                                  TRAP    T$ERCODE
(5)     017562  000062                           WORD   50
(5)     017564  003271                           WORD   MSG50
(5)     017566  004072                           WORD   AREA40
017570  104006                                  EMT     C$CLP1
017572  104032                                  EMT     C$EXIT
(3)     017574  000050                           WORD   L10031-
017576                                  50012$
                                        ;
                                        ;REPEAT THE LOOP UNTIL A MESSAGE IS RECEIVED
                                        ;
017576  005767  000344                          TST     MCOMP
(6)     017602  001723                          BEQ     50010$

                                        ;PROCESS THE MESSAGE RECEIVED

017604  004767  000146                          JSR     PC,STINPT
```

```
                                        ;
                                        ;IF IN ERROR...INSERT ERROR INTO TABLE (MANUF TESTS ONLY, REPORT
                                        ;ERROR, LOOP IF ENABLED, EXIT TEST IF NOT
                                        ;
017610  103015                                  BCC     50013$
017612  010546                                  MOV     R5,-(SP)
(4)  017614  012745  000001                     MOV     #1,-(R5)
(3)  017620  004767  176772                     JSR     PC,INSERR
(3)  017624  012605                             MOV     (SP)+,R5
017626  104463                                  TRAP    T$ERCODE
(5)  017630  000062                              WORD   50
(5)  017632  003271                              WORD   MSG50
(5)  017634  004072                              WORD   AREA40
017636  104006                                  EMT     C$CLP1
017640  104032                                  EMT     C$EXIT
(3)  017642  000002                              WORD   L10031-
017644                                  50013$
                                        ;
                                        ;END OF TEST
                                        ;
017644                                  L10031
(3)  017644  104001                             EMT     C$ETST


                                         ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                        ;                                                               +
                                        ;       ROUTINE TO SEND AND ACK IN ORDER TO POLL A TERMINAL     +
                                        ;                                                               +
                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
017646                                  OUTREP
                                        ;
                                        ;INSERT MESSAGE NUMBER (LAST MESSAGE SENT) AND RESPONSE NUMBER
                                        ;(LAST MESSAGE RECEIVED)    INSERT STATION ADDRESS
                                        ;
017646  116767  162312  000320                  MOVB    MSGNO,MSGN
017654  116767  162306  000311                  MOVB    RMSGNO,REPN
017662  116767  163617  000305                  MOVB    STRTSA,REPA
                                        ;
                                        ;COMPUTE AND INSERT CRC
                                        ;
017670  162705  000002                          SUB     #1*2,R5
(3)  017674  010546                             MOV     R5,-(SP)
(5)  017676  012745  000006                     MOV     #6,-(R5)
(4)  017702  012745  020170                     MOV     #REPM,-(R5)
(3)  017706  004767  176706                     JSR     PC,CRC
(3)  017712  012605                             MOV     (SP)+,R5
(4)  017714  012567  000256                     MOV     (R5)+,REPCRC

                                         SET UP BUFFER POINTER AND SIZE AND SEND THE MESSAGE
                                        ;
017720  012767  020170  175474                  MOV     #REPM,XBUFFER
017726  012767  000010  175470                  MOV     #10,XSIZE
017734  004767  174466                          JSR     PC,XMIT
```

```
                                        ;
                                        ;IF AN ERROR OCCURRED, RETURN W'TH ERROR. OTHERWISE RETURN NORMALLY
                                        ;
017740  103004                                  BCC     50002$
017742  005267  162152                          INC     OUTERR
017746  000261                                  SEC
 (4)    017750  000401                          BR      50001$
017752                                  50002$
017752                                  50000$
 (2)    017752  000241                          CLC
 (3)    017754                          50001$
 (2)    017754  000207                          RTS     PC


                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                                        ;                                                               +
                                        ;       ROUTINE TO PROCESS TERMINAL SELFTEST PESULTS            +
                                        ;                                                               +
                                        ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
017756                                  STINPT
017756  010146                                  MOV     R1 -(SP)
017760  010246                                  MOV     R2,-(SP)
017762  010346                                  MOV     R3,-(SP)
                                        ;
                                        ;SKIP RESPONSE HEADER AND CHECK FIPST DATA BYTE
017764  016701  162156                          MOV     INBUF,R1
 (6)    017770  062701  000004                  ADD     #4,R1
                                        ;IF IT IS "A", GOOD RESPONSE  OTHEPWISE
                                        ;
017774  121127  000101                          CMPB    (R1),#101
 (9)    020000  001455                          BEQ     50002$
                                        ;
                                        ;   GET THE BYTE AND CHECK TO SEE  F  T 'E "M" (MEMORY ERROR)
020002  112103                                  MOVB    (R1)+,R3
020004  120327  000115                          CMPB    R3,#115
 (9)    020010  001026                          BNE     50003$
                                        ;
                                        ;    MEMORY ERROR   GET THE TYPE AND BANK NUMBER
                                        ;        INSERT INTO ERROR MESSAGE, REPORT THE ERROR AND
                                        ;        RETURN WITH ERROR
                                        ;        (NOTE: ERROR REPORTING PROTOCOL DELIMITERS ARE
                                        ;        IGNORED  THIS PROGRAM WILL NOT ACCEPT INPUT FPOM
                                        ;        ANY OTHER INTELLIGENT TERMINAL PROPERLY )
020012  062701  000002                          ADD     #2,R1
020016  012702  003446                          MOV     #MTY,R2
020022  112122                                  MOVB    (R1)+,(R2)+
020024  112122                                  MOVB    (R1)+,(R2)+
020026  012702  003462                          MOV     #BNO,R2
020032  062701  000003                          ADD     #3,R1
```

```
 8327  020036  111103                          MOVB    (R1),R3
 8328  020040  062703  000060                  ADD     #60,R3
 8329  020044  110312                          MOVB    R3,(R2)
 8330  020046  104423                          TRAP    T$ERCODE
  (5)  020050  000107                           WORD   71
  (5)  020052  003422                           WORD   MSG71
 8331  020054  012603                          MOV     (SP)+,R3
 8332  020056  012602                          MOV     (SP)+,R2
 8333  020060  012601                          MOV     (SP)+,R1
 8334  020062  000261                          SEC
  (4)  020064  000427                          BR      50001$
 8335  020066                          50003$
 8336                                  ;
 8337                                  ;     NOT A MEMORY ERROR OR PASS   CHECK FOR "P" (PROCESSOR ERROR)
 8338                                  ;
 8339  020066  120327  000120                  CMPB    R3,#120
  (9)  020072  001010                          BNE     50005$
 8340                                  ;
 8341                                  ;     PROCESSOR ERROR   REPORT IT AND EXIT WITH ERROR
 8342                                  ;
 8343  020074  104423                          TRAP    T$ERCODE
  (5)  020076  000110                           WORD   72
  (5)  020100  003464                           WORD   MSG72
 8344  020102  012603                          MOV     (SP)+,R3
 8345  020104  012602                          MOV     (SP)+,R2
 8346  020106  012601                          MOV     (SP)+,R1
 8347  020110  000261                          SEC
  (4)  020112  000414                          BR      50001$
 8348  020114                          50005$
 8349
 8350                                   RESPONSE WAS NOT LEGAL   REPORT IT AND RETURN WITH ERROR
 8351
 8352  020114  104423                          TRAP    T$ERCODE
  (5)  020116  000106                           WORD   70
  (5)  020120  003367                           WORD   MSG70
 8353  020122  012603                          MOV     (SP)+,R3
 8354  020124  012602                          MOV     (SP)+,R2
 8355  020126  012601                          MOV     (SP)+,R1
 8356  020130  000261                          SEC
  (4)  020132  000404                          BR      50001$
 8357  020134                          50006$
 8358  020134                          50004$
 8359  020134                          50002$
 8360  020134  012603                          MOV     (SP)+,R3
 8361  020136  012602                          MOV     (SP)+,R2
 8362  020140  012601                          MOV     (SP)+,R1
 8363  020142                          50000$
  (2)  020142  000241                          CLC
  (3)  020144                          50001$
  (2)  020144  000207                          RTS     PC
 8364
 8365
 8366                                   MESSAGE AREA FOR SELFTEST
 8367                                  ;
 8368  020146  000000                  MCOMP   0
 8369  020150  020154                  MDMSG    +4
```

```
020152  000004                          EMDMG1-2-.
020154     040     002     050           BYTE 40,2,50,0
020157     000
020160                          EMDMG1
                                          EVEN
020160  120165                  MDMSG1  .+100005
020162  000004                          EMDMG2-2-.
020164     040     002     051           BYTE 40,2,51,0
020167     000
020170                          EMDMG2:
                                          EVEN
                                ;
                                ;ACK MESSAGE AREA
                                ;
020170     005     001     200  REPM    .BYTE ENQ,ACK,200
020173     000                  REPN    .BYTE 0
020174     000                  MSGN     BYTE 0
020175     000                  REPA.    BYTE 0
020176  000000                  REPCRC. .WORD 0
                                ;<><><><><><><>;<><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
                                ;                                                                      <>
                                ;                LOOPBACK TEST                                         <>
                                ;                                                                      <>
                                ;><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
                                ;
                                ;CLEAR ALL FLAGS
                                ;
020200  016767  157574  161700          MOV     C.ABORT
020206  016767  157566  161710          MOV     O.EOMSG
020214  016767  157560  161706          MOV     O.EXPINP
                                ;
                                ;START THE LINK (INIT DEVICE, DO DDCMP STRT)
                                ;
020222  004767  172266                  JSR     PC.XSTRT
                                ;
                                ;IF ERROR DURING START-UP   INSERT INTO TABLE (MANUF  TESTS ONLY, REPORT
                                ;IT. LOOP IF ENABLED--EXIT IF NOT
                                ;
020226  103015                          BCC     50007$
020230  010546                          MOV     R5,-(SP)
(4) 020232  012745  000000              MOV     #0,-(R5)
(3) 020236  004767  176354              JSR     PC,INSERP
(3) 020242  012605                      MOV     (SP)+,R5
020244  104463                          TRAP    T$ERCODE
(5) 020246  000063                       WORD   51
(5) 020250  003326                       WORD   MSG51
(5) 020252  004072                       WORD   AREA40
020254  104006                          EMT     C$CLP1
020256  104032                          EMT     C$EXIT
(3) 020260  000446                       WORD   L10032-
020262                          50007$
                                ;FILL MESSAGE AREA WITH 256 BYTES OF SELECTED PATTERN
020262  010146                          MOV     R1,-(SP)
```

```
020264  012701  021066                          MOV     #LOOPB,R1
020270  005767  171150                          TST     PATTERN
(9)  020274  001403                             BEQ     50010$
020276  005067  161600                          CLR     SCRTCH
020302  000403                                  BR      50011$
(3)  020304                          50010$
020304  012767  125125  161570                  MOV     #125125,SCRTCH
020312                               50011$.
020312                               50012$.
020312  016721  161564                          MOV     SCRTCH,(R1)+
020316  020127  021466                          CMP     R1,#ELOOPB
(6)  020322  001373                             BNE     50012$
                                     ;
                                     ;PUT A 1 AT THE END TO ASSURE THAT THE FINAL BYTE IS NOT NULL (0)
                                     ;
020324  012701  021466                          MOV     #ELOOPB,R1
(6)  020330  005301                             DEC     R1
020332  112711  000001                          MOVB    #1,(R1)
                                     ;
                                     ;GET THE STATION ADDRESS AND INSERT IT IN THE HEADER
                                     ;
020336  012701  002170                          MOV     #PTABLE,R1
(6)  020342  062701  000005                     ADD     #5,R1
020346  111167  000511                          MOVB    (R1),LOOSTA
020352  012601                                  MOV     (SP)+,R1
                                     ;
                                     ;INSERT LOOP DATA TYPE
                                     ;
020354  112767  000024  000504                  MOVB    #OSOP,LOOPB
                                     ;
                                     ;SET MESSAGE NUMBER
                                     ;
020362  112767  000001  000472                  MOVB    #1,LOOPN
                                     ;
                                     ;COMPUTE AND INSERT HEADER CRC
                                     ;
020370  162705  000002                          SUB     #1*2,R5
(3)  020374  010546                             MOV     R5,-(SP)
(5)  020376  012745  000006                     MOV     #6,-(R5)
(4)  020402  012745  021056                     MOV     #LOOPH,-(R5)
(3)  020406  004767  176206                     JSR     PC,CRC
(3)  020412  012605                             MOV     (SP)+,R5
(4)  020414  012567  000444                     MOV     (R5)+,LCRCH
                                     ;
                                     ;COMPUTE AND INSERT DATA CRC
                                     ;
020420  162705  000002                          SUB     #1*2,R5
(3)  020424  010546                             MOV     R5,-(SP)
(5)  020426  012745  000400                     MOV     #256.,-(R5)
(4)  020432  012745  021066                     MOV     #LOOPB,-(R5)
(3)  020436  004767  176156                     JSR     PC,CRC
(3)  020442  012605                             MOV     (SP)+,R5
(4)  020444  012567  001016                     MOV     (R5)+,LCRCD
                                     ;
                                     ;SET UP BUFFER POINTER AND SIZE    SEND MESSAGE
                                     ;
```

```
020450  012767  021056  174744          MOV     #LOOPH,XBUFFER
020456  012767  000412  174740          MOV     #266.,XSIZE
020464  004767  173736                  JSR     PC,XMIT
                                ;
                                ;IF ERROR OCCURRED...INSERT THE ERROR INTO TABLE (MANUF  TESTS ONLY),
                                ;REPORT THE ERROR, LOOP IF ENABLED
                                ;
020470  103012                          BCC     50013$
020472  010546                          MOV     R5,-(SP)
(4) 020474  012745  000000              MOV     #0,-(R5)
(3) 020500  004767  176112              JSR     PC,INSERR
(3) 020504  012605                      MOV     (SP)+,R5
020506  104463                          TRAP    T$ERCODE
(5) 020510  000063                      .WORD   51
(5) 020512  003326                       WORD   MSG51
(5) 020514  004072                       WORD   AREA40
020516                          50013$:
020516  104006                          EMT     C$CLP1
                                ;
                                ;SEND ACK'S TO TERMINAL UNTIL IT RESPONDS
                                ;
020520                          50014$:
020520  004767  177122                  JSR     PC,OUTREP
                                ;
                                ;IF AN ERROR OCCURS   INSERT IN TABLE (MAN  ONLY), REPORT IT, LOOP
                                ;IF ENABLED - EXIT TEST IF NOT
                                ;
020524  103023                          BCC     50015$
020526  010546                          MOV     R5,-(SP)
(4) 020530  012745  000000              MOV     #0,-(R5)
(3) 020534  004767  176056              JSR     PC,INSERR
(3) 020540  012605                      MOV     (SP)+,R5
020542  010546                          MOV     R5,-(SP)
(4) 020544  012745  000002              MOV     #2,-(R5)
(3) 020550  004767  176042              JSR     PC,INSERR
(3) 020554  012605                      MOV     (SP)+,R5
020556  104463                          TRAP    T$ERCODE
(5) 020560  000063                      .WORD   51
(5) 020562  003326                      .WORD   MSG51
(5) 020564  004072                      .WORD   AREA40
020566  104006                          EMT     C$CLP1
020570  104032                          EMT     C$EXIT
(3) 020572  000134                      .WORD   L10032-
020574                          50015$
                                ;
                                ;GET RESPONSE
                                ;
020574  162705  000002                  SUB     #1*2,R5
(3) 020600  004767  172610              JSR     PC,TERIN
(4) 020604  012567  000240              MOV     (R5)+,LCOMP
                                ;
                                ;IF ERROR OCCURS...INSERT IN TABLE (MAN  ONLY), REPORT IT, LOOP IF
                                ;ENABLED, EXIT TEST IF NOT
                                ;
020610  103012                          BCC     50016$
020612  010546                          MOV     R5,-(SP)
```

```
  (4)  020614  012745  000000                MOV     #0,-(R5)
  (3)  020620  004767  175772                JSR     PC,INSERR
  (3)  020624  012605                        MOV     (SP)+,R5
 8522  020626  104463                        TRAP    T$ERCODE
  (5)  020630  000063                        .WORD   51
  (5)  020632  003326                        .WORD   MSG51
  (5)  020634  004072                        .WORD   AREA40
 8523  020636                        50016$:
 8524  020636  104006                        EMT     C$CLP1
 8525                                ;
 8526                                ;REPEAT LOOP UNTIL TERMINAL SENDS LOOPBACK DATA
 8527                                ;
 8528  020640  005767  000204                TST     LCOMP
  (6)  020644  001725                        BEQ     50014$
 8529                                ;
 8530                                ;PROCESS THE LOOPBACK DATA
 8531                                ;
 8532  020646  004767  000056                JSR     PC,PROLOP
 8533  020652  104004                        EMT     C$BSEG
 8534                                ;
 8535                                ;REINIT THE LINK
 8536                                ;
 8537  020654  004767  171634                JSR     PC,XSTRT
 8538                                ;
 8539                                ;IF ERROR OCCURS, INSERT ERROR INTO TABLE (MANUF  ONLY), REPORT ERROR,
 8540                                ;LOOP ON REINIT IF ERROR LOOPING ENABLED   EXIT TEST IF NO LOOPING
 8541                                ;
 8542  020660  103020                        BCC     50017$
 8543  020662  010546                        MOV     R5,-(SP)
  (4)  020664  012745  000000                MOV     #0,-(R5)
  (3)  020670  004767  175722                JSR     PC,INSERR
  (3)  020674  012605                        MOV     (SP)+,R5
 8544  020676  010546                        MOV     R5,-(SP)
  (4)  020700  012745  000002                MOV     #2,-(R5)
  (3)  020704  004767  175706                JSR     PC,INSERR
  (3)  020710  012605                        MOV     (SP)+,R5
 8545  020712  104463                        TRAP    T$ERCODE
  (5)  020714  000063                         WORD   51
  (5)  020716  003326                         WORD   MSG51
  (5)  020720  004610                        .WORD   AREA61
 8546  020722                        50017$
 8547  020722  104006                        EMT     C$CLP1
 8548  020724                        10000$:
  (3)  020724  104005                        EMT     C$ESEG
 8549                                ;
 8550                                ;END OF TEST
 8551                                ;
 8552  020726                        L10032
  (3)  020726  104001                        EMT     C$ETST
 8553
 8554
 8555
 8556
 8557
 8558                                ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 8559                                                                                              +
```

```
                                ;       ROUTINE TO PROCESS LOOPBACK DATA                        +
                                ;                                                               +
                                ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
020730                          PROLOP.
020730  010146                          MOV     R1,-(SP)
020732  010246                          MOV     R2,-(SP)
020734  010346                          MOV     R3,-(SP)
                                ;
                                ;R1 = POINTER TO EXPECTED DATA
                                ;R2 = COUNT OF DATA EXPECTED
                                ;R3 = POINTER TO DATA FROM TERMINAL
                                ;
020736  012701  021066                  MOV     #LOOPB,R1
020742  012702  000200                  MOV     #128 ,R2
020746  016703  161174                  MOV     INBUF,R3
                                ;
                                ;CHECK DATA BYTE-BY-BYTE
                                ;
020752                          50002$.
                                ;
                                ;IF ERROR.  INSERT ERROR INTO TABLE (MANUF  ONLY), REPORT ERROR
                                ;(INCLUDING RECEIVED/EXPECTED BYTES), RETURN WITH ERROR
                                ;
020752  121113                          CMPB    (R1),(R3)
 (9)    020754  001423                  BEQ     50003$
020756  011167  164054                  MOV     (R1),A52A
020762  011367  164052                  MOV     (R3),A52B
020766  010546                          MOV     R5,-(SP)
 (4)    020770  012745  000002          MOV     #2,-(R5)
 (3)    020774  004767  175616          JSR     PC,INSERR
 (3)    021000  012605                  MOV     (SP)+,R5
021002  104464                          TRAP    T$ERCODE
 (5)    021004  000064                   WORD   52
 (5)    021006  003354                   WORD   MSG52
 (5)    021010  004740                   WORD   AREA52
021012  012603                          MOV     (SP)+,R3
021014  012602                          MOV     (SP)+,R2
021016  012601                          MOV     (SP)+,R1
021020  000261                          SEC
 (4)    021022  000411                  BR      50001$
021024                          50003$
                                ;
                                ;UPDATE COUNT AND POINTERS
                                ;
021024  005201                          INC     R1
021026  005302                          DEC     R2
021030  005203                          INC     R3

                                ;REPEAT UNTIL ALL DATA HAS BEEN CHECKED

021032  005702                          TST     R2
 (6)    021034  001346                  BNE     50002$
021036  012603                          MOV     (SP)+,R3
021040  012602                          MOV     (SP)+,R2
021042  012601                          MOV     (SP)+,R1
021044                          50000$
```

```
  (2)  021044  000241                          CLC
  (3)  021046                          50001$:
  (2)  021046  000207                          RTS     PC
 8607
 8608                                  ;
 8609                                  ;LOOPBACK MESSAGE AREA
 8610                                  ;
 8611  021050  000000                  LCOMP:  0
 8612  021052  121057                  LOOPMG:  .+100005
 8613  021054  000410                          ELOOPB-2-.
 8614  021056     220     000     201  LOOPH    BYTE DLE,0,201,0
       021061     000
 8615  021062     000                  LOOPN    .BYTE 0
 8616  021063     000                  LOOSTA   BYTE   0
 8617  021064  000000                  LCRCH   0
 8618  021066  000200                  LOOPB    BLKW   128
 8619  021466                          ELOOPB
 8620  021466  000000                  LCRCD   0
 8622
```

```
                                        ;
                                        ;       HARDWARE PARAMETER CODING
                                        ;
        021470  000022                          .WORD L10033-L$HARD/2
        021472  000031                           WORD   T$CODE
         (4)    021474  021536                  .WORD   DEVICE
         (4)    021476  000000                  .WORD   T$LOLIM
         (4)    021500  177776                   WORD   T$HILIM
        021502  001031                           WORD   T$CODE
         (4)    021504  021554                  .WORD   VECTOR
         (4)    021506  000000                  .WORD   T$LOLIM
         (4)    021510  000400                  .WORD   T$HILIM
        021512  002032                           WORD   T$CODE
         (4)    021514  021571                   WORD   PRIOR
         (4)    021516  000340                   WORD   340
         (4)    021520  000000                   WORD   T$LOLIM
         (4)    021522  000007                   WORD   T$HILIM
        021524  002032                           WORD   T$CODE
         (4)    021526  021614                  .WORD   STATION
         (4)    021530  177400                   WORD   177400
         (4)    021532  000000                   WORD   T$LOLIM
         (4)    021534  000377                   WORD   T$HILIM
                                                 EVEN
         (3)    021536                  L10033
                021536  052504  020120  042101  DEVICE  .ASCIZ /DUP ADDRESS  /
                021544  051104  051505  035123
                021552  000040
                021554  052504  020120  042526  VECTOR   ASCIZ /DUP VECTOR  /
                021562  052103  051117  020072
                021570     000
                021571     104  050125  041040  PRIOR    ASCIZ /DUP BUS PRIORITY. /
                021576  051525  050040  044522
                021604  051117  052111  035131
                021612  000040
                021614  042524  046522  047111  STATION  ASCIZ /TERMINAL ADDRESS  /
                021622  046101  040440  042104
                021630  042522  051523  020072
                021636     000
                        021640                          .EVEN
                                        ;
                                        ;       SOFTWARE PARAMETER CODING
                                        ;
        021640  000010                           WORD L10034-L$SOFT/2
        021642  000032                           WORD   T$CODE
         (4)    021644  021662                   WORD   ALIMIT
         (4)    021646  177777                  .WORD   177777
         (4)    021650  000000                  .WORD   T$LOLIM
         (4)    021652  077777                  .WORD   T$HILIM
        021654  001130                           WORD   T$CODE
         (4)    021656  021731                   WORD   PATTRN
         (4)    021660  177777                   WORD   177777
                                                 EVEN
         (3)    021662                  L10034
        021662  047510  020127  040515  ALIMIT   ASCIZ /HOW MANY INITIATE TRIES BEFORE ABORT
```

```
        021670  054516  044440  044516
        021676  044524  052101  020105
        021704  051124  042511  020123
        021712  042502  047506  042522
        021720  040440  047502  052122
        021726  020077     000
  8968  021731     125  042523  040440  PATTRN  .ASCIZ /USE ALL ZERO'S FOR LOOPBACK DATA? /
        021736  046114  055040  051105
        021744  023517  020123  047506
        021752  020122  047514  050117
        021760  040502  045503  042040
        021766  052101  037501  000040
  8972                                          EVEN
  8974                                         .EVEN
  (3)   021774                          L$LAST
  8975                                          REQ DOCTOR P11 400,1066
  8976                                          TITLE  PDP-11 DIAGNOSTIC SUPERVISOR
 20272          052674                         END.SUPV=.+2
 20274          071776                          =71776
 20275  071776  000000                          WORD   0
 20277          072000                         X1X1=
 20278          000200                          END 200
```

```
A       = 000000
ABOFLA  022246 G
ABOPAS  022160 G
ABORT   002106
ABO FM  025122
ACK     = 000001
ALIMIT  021662
ALLOC   042712
AREA00  003510 G
AREA01  003570 G
AREA02  003572 G
AREA03  003674 G
AREA04  003726 G
AREA11  004030 G
AREA21  004066 G
AREA22  004070 G
AREA23  004072 G
AREA24  004072
AREA26  004264 G
AREA30  004266 G
AREA31  004266
AREA40  004072
AREA41  003726
AREA42  003674
AREA43  003726
AREA52  004740 G
AREA60  004532 G
AREA61  004610 G
AREA70  005042 G
AREA71  005044 G
A$AAW   027032
A$AAX   027046
A$AAY   027054
A$AAZ   027070
A$ABA   027100
A02A    003624
A02B    003626
A04EX   003762
A04RC   003760
A52A    005036
A52B    005040
B       = 000002
BAPNT   002160
BASIZE  002162
BGN SU= 021774
BINMSG  041222
BIT0    = 000001 G
BIT00   = 000001 G
BIT01   = 000002 G
BIT02   = 000004 G
BIT03   = 000010 G
BIT04   = 000020 G
BIT05   = 000040 G
BIT06   = 000100 G
BIT07   = 000200 G
BIT08   = 000400 G
BIT09   = 001000 G
BIT1    = 000002 G
BIT10   = 002000 G
BIT11   = 004000 G
BIT12   = 010000 G
BIT13   = 020000 G
BIT14   = 040000 G
BIT15   = 100000 G
BIT2    = 000004 G
BIT3    = 000010 G
BIT4    = 000020 G
BIT5    = 000040 G
BIT6    = 000100 G
BIT7    = 000200 G
BIT8    = 000400 G
BIT9    = 001000 G
BLD HW  027712
BLOCK   045046
BNC     003462
B$AAB   031314
B$AAF   031226
C       = 000004
CALLPC= 000022
CALLPS= 000024
CALLSP= 000026
CALLTC= 000030
CAL CL  047430
CAL TI  047466 G
CHKLUP  031330
CHKREP  014036
CHKSTR  043254
CHKTTY  041342
CHK FO  023540
CHK MA  027470
CHK.PC  034556
CHK SW  023262
CHRCNT  042574
CH FLA  027172
CH PAS  027214
CLEAR   030612
CLKACC  022156 G
CLKBFR  047432
CLKCNT  022154 G
CLKRES  051030 G
CLKSER  051330 G
CLKSON  022220 G
CLK SE  027272
CLR MA  027546
CNTFLG  015434
CNVT    045506
CODE    011454 G
COMMTA  045326
CONTCL  051110 G
CRC     016620
CRCTAB  017074
CRLF    041424
CTS     = 020000
CURR.T  022174 G
C$AAD   034530
C$AAE   034542
C$AAK   035360
C$AAL   035470
C$ABRT= 000021
C$ADR = 000020
C$AU  = 000054
C$BRK = 000022
C$BSEG= 000004
C$BSUB= 000002
C$BUFF= 000030
C$CEFG= 000046
C$CLEA= 000012
C$CLP1= 000006
C$CVEC= 000036
C$DCLN= 000044
C$DODU= 000053
C$DRPT= 000024
C$DU  = 000055
C$EDIT= 000006
C$ERDF= 000002
C$ERHR= 000003
C$ERSF= 000001
C$ERSO= 000004
C$ESCA= 000010
C$ESEG= 000005
C$ESUB= 000003
C$ETST= 000001
C$EXIT= 000032
C$GMAN= 000043
C$GPHR= 000042
C$GPRI= 000040
C$GTIM= 000052
C$INIT= 000011
C$INLP= 000020
C$KWOF= 000035
C$KWON= 000034
C$LOOP= 000100
C$MANI= 000051
C$MSG = 000023
C$PNTB= 000014
C$PNTF= 000017
C$PNTS= 000016
C$PNTX= 000015
C$POIN= 000040
C$QIO = 000377
C$RDBU= 000007
C$REFG= 000050
C$REQT= 000045
C$RESE= 000033
C$REVI= 000001
C$RPT = 000025
C$SEFG= 000047
C$SPRI= 000041
C$SVEC= 000037
C$TPRI= 000013
C$UNBU= 000031
C$WTM = 000026
C$WTU = 000027
D       = 177776
DDSTRT= 000000
DDSTRY  012772
DECMSG  041236
DEVICE  021536
DEVINI  014742
DEVPRI  002200
DIAG.T  022254 G
DLE     = 000220
DPDVD   052040 G
DPMUL   051726 G
DTR     = 000002
DUNIT.  022164 G
DUPMSK  002220 G
DUPREG  002224 G
DUP$    = 000001
DVC FT  035330
D$AAG   036200
D$AAH   036216
D$AAI   040770
D$AAJ   040774
D$AAK   041012
D$AAL   041030
D$AAM   041040
E       = 177774
EF CON= 000036 G
EF NEW= 000035 G
EF PWR= 000034 G
EF RES= 000037 G
EF STA= 000040 G
EF01    = 000001 G
EF02    = 000002 G
EF03    = 000003 G
EF04    = 000004 G
EF05    = 000005 G
EF06    = 000006 G
EF07    = 000007 G
EF08    = 000010 G
EF09    = 000011 G
EF10    = 000012 G
EF11    = 000013 G
EF12    = 000014 G
EF13    = 000015 G
EF14    = 000016 G
EF15    = 000017 G
EF16    = 000020 G
ELOOPB  021466
EMDMG1  020160
EMDMG2  020170
EMT.TR  022252 G
END OF  030600
END SU= 052674
ENQ     = 000005
EOMSG   002124
EOP CH  051352 G
EOP.FM  025136
EOP IN  027210
ERRFOR  035546
ERRHAN  034562
ERR HR  035340
ERR SF  035344
ERR1FO  035632
ESC PC  034554
EXPINP  002130
FIL     015420
FILL    042072
FILL C  000204 G
FLAGS   022216 G
FLAGTA  045244
FLAG.I  027256
FLA.SE  045212
FLG MA  027216
FLLCNT  015414
FORM.T  035642
FORM00  003536
FORM02  003630
FORM04  003764
FORM23  004166
FORM30  004420
FORM40  004206
FORM52  004772
FORM60  004554
FORM61  004702
FREE    043150
FRM30A  004442
FRM30B  004465
FRM30C  004507
FS$     = 000001
FS$AU   = 000015
FS$BGN  = 000040
FS$CLEA= 000007
FS$DU   = 000016
```

```
F$END = 000041
F$HARD= 000004
F$HW  = 000013
F$INIT= 000006
F$JMP = 000050
F$MOD = 000000
F$MSG = 000011
F$PWR = 000017
F$RPT = 000012
F$SEG = 000003
F$SOFT= 000005
F$SRV = 000010
F$SUB = 000002
F$SW  = 000014
F$TEST= 000001
GARBAG  042576
GETCHR  041302
GETCMN  044666
GETOB   012774
GETPAR  036360
GETSWI  043662
GET.TW  043432
GLOB    002100 G
GSUB    012052 G
G$EXCP= 000400
G$HILI= 000002
G$LOLI= 000001
G$NO  = 000000
G$OFFS= 000400
G$OFSI= 000376
G$PRMA= 000001
G$PRMD= 000002
G$PRML= 000000
G$RADA= 000140
G$RADB= 000000
G$RADD= 000040
G$RADF= 000200
G$RADL= 000120
G$RADO= 000020
G$RADT= 000100
G$XFER= 000004
G$YES = 000010
HCORED  026754
HCOREQ  026634
HCORET  022206 G
HEAD    002000 G
HERTZ   026574
HOLDSP= 000020
HYBYTE= 177400
H$AAB   046034
ILIMIT  002114
INBUF   002146
INERR   002116
ININIT  022176 G
INIT    015416
INITIA  041252
INITMA  002154
INITMS  002156
INIT.M  027614
INIT.R  022010 G
INMSG   002136
INPUTA  042200
INSERR  016616
INSIZ   002150
INTFOR  035476
INVAL.  026722
INVINT  035370
INV SW  023216
IN SUF  030564
I$AU  = 000041
I$CLN = 000041
I$DU  = 000041
I$HRD = 000041
I$INIT= 000041
I$MOD = 000041
I$MSG = 000041
I$PWR = 000041
 $RPT = 000041
I$SEG = 000041
 $SFT = 000041
I$SRV = 000041
I$SUB = 000041
I$TST = 000041
J$JMP = 000167
KBPTR   022026 G
KBUF    022030 G
KEX     004064
KRC     004062
LCOMP   021050
LCRCD   021466
LCRCH   021064
LINE F  022250 G
LOAD F  027212
LOBYTE= 000377
LOGDEV  002126
LOGMSG  041244
LOOPB   021066
LOOPH   021056
LOOPMG  021052
LOOPN   021062
LOOSTA  021063
LPBFR   022024 G
LPCNTR  022022 G
LPT AD  026612
LPT RE  026606
LST RE  026602
LUP     047334
LUP.AD  034560
L$AU    011456 G
L$AUT   002070 G
L$CCP   002044 G
L$CLEA  012026 G
L$DEPO  002011 G
L$DEVP  002052 G
L$DISP  011450 G
L$DR    002220 G
L$DRCT  002062 G
L$DRS   002064 G
L$DRST  002224 G
L$DTP   002040 G
L$DU    011460 G
L$DUT   002072 G
L$DVTY  002202 G
L$EF    002024 G
L$EXP1  002032 G
L$EXP2  002034 G
L$EXP3  002036 G
L$HARD  021472 G
L$HPCP  002046 G
L$HPTP  002056 G
L$HW    011430 G
L$ICP   002042 G
L$INIT  011462 G
L$LADP  002076 G
L$LAST  021774 G
L$MREV  002012 G
L$NAME  002000 G
L$REPP  002054 G
L$REV   002010 G
L$RPT   011454 G
L$SOFT  021642 G
L$SPC   002030 G
L$SPCP  002050 G
L$SPTP  002060 G
L$STA   002066 G
L$SW    011442 G
L$TIML  002022 G
L$TIMU  002020 G
L$TIM1  002016 G
L$TSTI  002074 G
L$UNIT  002014 G
L CLK   026560
L10000  003534
L10001  003570
L10002  003622
L10003  003724
L10004  003756
L10005  004060
L10006  004066
L10007  004070
L10010  004162
L10011  004264
L10012  004416
L10013  004552
L10014  004700
L10015  004770
L10016  005042
L10017  005044
L10020  011440
L10021  011446
L10022  011454
L10023  011456
L10024  011460
L10025  011734
L10026  012050
L10027  015632
L10030  016324
L10031  017644
L10032  020726
L10033  021536
L10034  021662
MAJ IN  022000 G
MAJ LO  047434
MAJ US  022002 G
MAN TI  024302
MAP16   052276 G
MASCLR= 000400
MASK B  031326
MASK W  031324
MCOMP   020146
MDMSG   020150
MDMSG1  020160
MEM SI  026622
MIN IN  021774 G
MIN US  021776 G
MODR    051640 G
MSG     012052
MSGAP   002152
MSGN    020174
MSGNO   002164
MSG00   002234
MSG01   002273
MSG02   002320
MSG03   002345
MSG04   002411
MSG10   002436
MSG11   002516
MSG12   002551
MSG20   002620
MSG21   002637
MSG22   002657
MSG23   002722
MSG24   002762
MSG25   003025
MSG26   003062
MSG30   003117
MSG31   003147
MSG40   002637
MSG41   002411
MSG42   002345
MSG43   003224
MSG45   003243
MSG50   003271
MSG51   003326
MSG52   003354
MSG70   003367
MSG71   003422
MSG72   003464
MTY     003446
MUL     051574 G
NAK   = 000002
NEWPRI  051320 G
NEXTAR  045430
NMSG    015426
NO.CLK  024332
NO FLA  045224
NO LPT  042542
NO PTA  027022
NR    = 000001
NUMBIN  035666
NUM.LA  036034
NUM UN  022372
NUNITS  031302
NXTFOR  045500
OCTMSG  041230
OSOP  = 000024
OUT     002152
OUTBUF  002142
OUTERR  002120
OUTMSG  002132
OUTREP  017646
OUTSIZ  002144
O$APTS= 000001
O$AU  = 000001
O$BGNR= 000001
O$BGNS= 000001
O$DU  = 000001
O$GNSW= 000001
O$POIN= 000001
O$PWR = 000001
PARAM = 101226
PARITY= 000200
PARSES  044740
PAR LA  040732
PASCNT  002122
```

```
PATTER  011444
PATTRN  021731
PRINTC  042552
PRINTF  046054
PRIOR   021571
PRI00 = 000000 G
PRI01 = 000040 G
PRI02 = 000100 G
PRI03 = 000140 G
PRI04 = 000200 G
PRI05 = 000240 G
PRI06 = 000300 G
PRI07 = 000340 G
PRNTST  042442
PROBUF  013122
PROLOP  020730
PRO CM  027164
PTABLE  002170
PTAB S  022204 G
PTCODE  021470 G
PTEND   002200
PUTCHR  041256
PWR FA  052532 G
PWR FL  022006 G
PWR MS  052660
PWR SA  052654
PWR UP  052656
P CLK   026566
RBQUE   015456
RBQUEB  015464
RBQUET  015460
RCENBL= 000020
RCVADD  015406
RCVBUF= 000002
RCVCSR= 000000
RCVE    016326
RDONE   014740
READ P  047436 G
REGBAC  052262 G
REGSAV  052246 G
REP   = 000003
REPA    020175
REPCRC  020176
REPM    020170
REPN    020173
REQN P  027174
REQN T  027166
RE SET  023364
RINT    015634 G
RMSGNO  002166
RQUE    015440
RQUEB   015454
RQUET   015442
RQUEUE  015436
RSTACK  051522 G
RSX.FL  027206
RTEMP1  015430
RTEMP2  015432
RTS   = 000004
RXINTE= 000120
RSSTCK  002100
SCRTCH  002102
SEARCH  043400
SEGSTA  022222 G
SENDFD= 000020
SET MA  027402
SHIFT   052360 G
SIZE.C  051236 G
SIZE M  051154 G
SIZ TR  051314
SOH   = 000201
SPEC U  027112
SPV SE  023614
SSTACK  011426
STACK = 000007
STARTC  051104 G
STATIO  021614
STIME   002112
STINPT  017756
STRCHR  042132
STREQ   026734
STRT  = 000006
STRTB   003500
STRTC   003506
STRTSA  003505
STRT T  027170
ST REQ  027110
ST SET  023434
SUNIT   027176
SUPERV  025154
SUPFLA  022162 G
SUPV T  022340 G
SUP PR  023206
SVCCNT= 177777
SVCGBL= 000000
SVCHAN  031506
SVCINS= 000000
SVCSTK= 177777
SVCSUB= 177777
SVCTAG= 000000
SVCTST= 177777
SWCHAN  027014
SWITCH  045404
SW PTA  027000
SYNC  = 000226
SYNCNT  015412
SYS FT  035320
S$LSYM= 010000
TABLE   011426 G
TEMBUF  005046
TEMP    002104
TEMP2   004164
TERIN   013414
TERMI   047424
TERMLI  045232
TERMTA  041214
TEROUT  012260
TESTS   017174 G
TEST.M  027124
TIMER   015410
TIMFLG  022152 G
TIMLIM  002110
TIM CO  022004 G
TIM OP  035640
TMPBUF  010172
TOO MA  041174
TSOM  = 000400
TSTS  = 000002
TST AB  031440
TST TO  023244
TXINTE= 000100
TYPE    012512
TYPEC   041570
TYPEPC  035464
TYPFLA  045106
TYPLIN  041466
TYPNUM  041054
TYPSTR  041506
TYP ER  035350
TY UNI  030604
T$ARGC= 000003
T$CODE= 001130
T$ERCO= 000064
T$ERRN= 000064
T$EXCP= 000000
T$FLAG= 000040
T$HILI= 077777
T$LOLI= 000000
T$LSYM= 010000
T$MCAL= 177777
T$NEST= 177777
T$NSK0= 000000
T$NSK1= 000005
T$NSK2= 000003
T$SAVL= 177777
T$SEGL= 177777
T$SEK0= 010000
T$SUBN= 000000
T$TAGL= 177777
T$TAGN= 010035
T$TEMP= 000000
T$TEST= 000002
T$TSTM= 177777
T$TSTS= 000001
T$SAU = 010023
T$SCLE= 010026
T$SDU = 010024
T$SHAR= 010033
T$SHW = 010020
T$SINI= 010025
T$SMSG= 010017
T$SRPT= 010022
T$SSEG= 010000
T$SSOF= 010034
T$SSRV= 010030
T$SSW = 010021
T$STES= 010032
T1      017174 G
T2      020200 G
UNI MA  027114
USER P  022200 G
USER T  022202 G
VALID.  022442
VAL LA  023172
VAL SW  027230
VECTOR  021554
VMSG    011736
WIDTH   036234
XBUFFE  015422
XDONE   014736
XEQDIA  051406 G
XEQSUB  051374 G
XEQ CL  031244
XEQ CM  026552
XEQ IN  030726
XEQ LA  025110
XEQ OP  031020
XEQ PP  024342
XEQ TE  031064
XINT    015466 G
XMIT    014426
XMTADD  015404
XMTBUF= 000006
XMTCSR= 000004
XSIZE   015424
XSTRT   012514
XTIME   050114 G
XTIMEN  050740
XTIMST  050136
XXDP D  026762
X$ALWA= 000000
X$FALS= 000040
X$OFFS= 000400
X$TRUE= 000020
X1X1  = 072000
$BGNLE= 177777
$BREG   027270
$ENDAD  051360 G
$ERFLG= 000400
$F$AND= 000310
$F$BAD= 000401
$F$BLA= 000170
$F$CAS= 000150
$F$DEC= 000220
$F$DO = 000340
$F$FAL= 000405
$F$GOO= 000400
$F$IF = 000110
$F$INC= 000210
$F$LOO= 000200
$F$NAM= 000160
$F$NO = 000403
$F$OR = 000320
$F$RTN= 000300
$F$SEL= 000140
$F$THE= 000330
$F$TRU= 000404
$F$UNT= 000130
$F$WHI= 000120
$F$YES= 000402
$IFLEV= 177777
$ISK0 = 000001
$ISK1 = 000001
$ISK2 = 000001
$ISK3 = 000001
$LOCTA= 177777
$LSTCN= 177777
$LSTIN= 000000
$LSTST= 177777
$LSTTA= 000000
$MCALL= 000000
$NESTL= 177777
$NSK0 = 000300
$NSK1 = 000130
$NSK2 = 000110
$NSK3 = 000110
$NSK4 = 000110
$SAVLE= 177777
$SAV2   052424 G
$SAV3   052440 G
$SAV4   052456 G
$SAV5   052476 G
$SSK0 = 050003
$TAGLE= 177777
$TAGNU= 050004
```

```
$TEMP = 000300          $$BYTE= 000402          $$FROM= 000000          $$RTN2= 050001          $$$TAG= 050000
$TSK0 = 050002          $$CASE= 000000          $$LOC = 021034          $$SRC = 000064                  = 072000
$TSK1 = 050003          $$DST = 000003          $$LOCN= 000000          $$TGSV= 000000
$TSK2 = 050006          $$ELOC= 000402          $$REG = 177777          $$TGS1= 000000
$TSK3 = 050033          $$ERFL= 000000          $$RETU= 000001          $$TGS2= 000000
$$ARGC= 000000          $$FLAG= 000001          $$RTN1= 050000          $$T0  = 000001

  ABS   072000     000


 ERRORS DETECTED   0

 DIDPGA=DIDPGA
 RUN-TIME  93 95  9 SECONDS
 RUN-TIME RATIO  1402 190=7 3
  PE  EFC   75K  169 PAGES)
```