```
     TITLE DCMVA        07/28/78  1733.9 edt Fri        HRS ASSEMBLER 5.05     PAGE  0001
000001                                  TITLE     DCMVA,'78072500'    MLCP LINE ADAPTER TEST
000002                          *
000003                          *
000004                          * DESCRIPTION:
000005                          *
000006                          *   THIS T&V PROGRAM VERIFIES PROPER OPERATION OF THE LEVEL 6 COMM LINE
000007                          *   ADAPTERS, WHICH ARE ATTACHMENTS TO THE MULTILINE COMMUNICATION PTOCESSOR.
000008                          *
000009                          *   THE SUBSYSTEM ITEMS SUPPORTED BY THIS PROGRAM ARE:
000010                          *
000011                          *       MLC9101 MULTILINE COMMUNICATION PROCESSOR
000012                          *       MLC9102 MULTILINE COMMUNICATION PROCESSOR
000013                          *       MLC9103 MULTILINE COMMUNICATION PROCESSOR
000014                          *
000015                          *       DCM9101 DUAL ASYCHRONOUS ADAPTER
000016                          *       DCM9102 SINGLE ASYCHRONOUS ADAPTER
000017                          *       DCM9103 DUAL SYNCHRONOUS ADAPTER
000018                          *       DCM9104 SINGLE SYNCHRONOUS ADAPTER
000019                          *       DCM9105 SINGLE WIDEBAND CURRENT MODE SYNCHRONOUS ADAPTER
000020                          *       DCM9108 SINGLE WIDEBAND BALANCED LINE SYNCHRONOUS ADAPTER
000021                          *       DCM9109 SINGLE MIL 188C SYNCHRONOUS ADAPTER
000022                          *       DCM9111 SINGLE CURRENT LOOP ASYNCHRONOUS ADAPTER
000023                          *       DCM9114 DUAL CURRENT LOOP ASYNCHRONOUS ADAPTER
000024                          *
000025                          * REVISION HISTORY:
000026                          *
000027                          *       REV     DATE
000028                          *        A      JULY 76    ORIGINAL RELEASE
000029                          *        B      AUG  76
000030                          *        C      SEPT 76
000031                          *        D      DEC  76
000032                          *        E      MAR  77
000033                          *        F      MAY  77
000034                          *        G      JULY 77
000035                          *        H      OCT  77
000036                          *#       VA     AUG  78
000037                          *
000038                          ********************************************************************************
000039                          *    THIS  DOCUMENT  AND  THE   INFORMATION   CONTAINED
000040                          *    THEREIN IS CONFIDENTIAL AND PROPRIETARY TO AND THE
000041                          *    EXCLUSIVE   PROPERTY   OF    HONEYWELL INFORMATION
000042                          *    SYSTEMS INC.  IT  IS MADE AVAILABLE ONLY TO HONEY-
000043                          *    WELL AUTHORIZED RECIPIENTS FOR THEIR USE SOLELY IN
000044                          *    THE  MAINTENANCE  AND   OPERATION   OF   HONEYWELL
000045                          *    PRODUCTS.  THIS  DOCUMENT  AND INFORMATION MUST BE
```

```
000046                          *    MAINTAINED IN STRICTEST CONFIDENCE;  IT  MUST  NOT
000047                          *    BE  REPRODUCED  IN  WHOLE OR IN PART; AND IT SHALL
000048                          *    NOT BE DISCLOSED TO ANY OTHER  PARTY  WITHOUT  THE
000049                          *    PRIOR WRITTEN CONSENT OF HONEYWELL.
000050                          *
000051                          * PROGRAM PREPARATION:
000052                          *
000053                          *
000054                          *    THE ROOT SOURCE OF THIS PROGRAM, AFTER THE ADDITION OF APPROPRIATE
000055                          *    TITLE AND END STATEMENTS, WAS PROCESSED BY THE HOST RESIDENT ASSEMBLER
000056                          *    TO CREATE EITHER SHORT OR LONG ADDRESS FORM (SAF OR LAF) OBJECT TEXT
000057                          *    AND LISTING. THE OBJECT TEXT WAS FURTHER PROCESSED BY THE HOST RESIDENT
000058                          *    LINKER USING THE APPROPRIATE CONSOLE ZV$LIB LIBRARY TO CREATE A PUNCH
000059                          *    SEGMENT CONTAINING AN EXECUTABLE MODULE. THE ASSEMBLY LISTING WAS
000060                          *    AUGMENTED WITH CROSS REFERENCE DATA PLUS THE LOAD MAP FROM THE LINKER
000061                          *    TO CREATE A LIST SEGMENT.
000062                          *
000063                          *                ROOT           SAF            LAF
000064                          *                ----           ---            ---
000065                          *        NAME    DCMX1          DCMS1          DCML1
000066                          *        DOCUMENT  ***SPECIAL  ***SPECIAL  ***SPECIAL
000067                          *
000068                          *
000069                          * PROGRAM DISTRIBUTION:
000070                          *
000071                          *
000072                          *    THE ELEMENTARY ITEMS SUBMITTED TO THE T&V PROGRAM DISTRIBUTION CENTER
000073                          *    WERE THE EXECUTABLE PUNCHED CARD DECKS OF DCMS1 AND DCML1, AND MAGNETIC
000074                          *    TAPE IMAGES OF THE AUGMENTED LISTINGS.
000075                          *
000076                          *    REPRODUCTIONS OF THE EXECUTABLE CARD DECKS MAY BE AS DUPLICATE CARD DECKS
000077                          *    OR AS MEMBER "VL" OF A MULTIPLE MEMBER FILE. IN THE MOST FREQUENT CASE
000078                          *    IT WILL BE FOUND AS MEMBER VL WITHIN FILE PROGFILE OF A DISKETTE VOLUME
000079                          *    ENTITLED DIAGS.
000080                          *
000081                          *    DISTRIBUTION OF THE LISTINGS, WHICH SHOULD BE AVAILABLE IF ANY COMPLEX
000082                          *    MAINTENANCE OR REPAIR IS TO BE PERFORMED, IS NORMALLY MADE AS A
000083                          *    PRINTED COPY.
000084                          *
000085                          *
000086                          * ROUTINE DEMONSTRATION:
000087                          *
000088                          *
000089                          *    A MINIMUM SATISFACTORY TEST FOR NORMAL OPERATION MAY BE OBTAINED BY
000090                          *    ANSWERING TH QUESTION 'NEXT' WITH AN  'A' AND COMPLETING ONE PASS.
000091                          *
000092                          *
000093                          * MAIN MEMORY REQUIREMENT:
000094                          *
```

```
000095                          *
000096                          *    THIS PROGRAM REQUIRES 8K WORDS OF MAIN MEMORY AND WILL USE
000097                          *    ALL OF AVAILABLE MEMORY THROUGH 8K WORDS.
000098                          *
000099                          ********************************************************************************
000100                          *
000101                          *
000102                          * TEST PROCEEDURE
000103                          *
000104                          *
000105                          *   BEFORE RUNNING THE CLA TEST, THE MOTHER BOARD TEST (MLCS1,MLCL1)
000106                          *   SHOULD BE RUN TO MAKE SURE THE MLCP IS OPERATING CORRECTLY.
000107                          *
000108                          *   THE PROGRAM AUTOMATICALLY TEST ALL CLA'S WHICH ARE LISTED ABOVE
000109                          *   UNDER DESCRIPTION.  IT PROCEEDS TO EXERCISE ALL THE LINE ADAPTER
000110                          *   FUNCTIONAL TESTS IN LOOP BACK MODE.
000111                          *
000112                          *   THE TESTS ARE SET UP SO THAT ALL THE CLA'S ARE TESTED ACCORDING TO A
000113                          *   PARTICULAR FUNCTION.  SOME OF THE FUNCTIONAL TESTS INCLUDE CHARACTER
000114                          *   SIZE TEST, LINE SPEED TEST, PARITY AND CRC TEST, AND UNDERRUN AND
000115                          *   OVERRRN TEST.
000116                          *
000117                          *
000118                          *
000119                          * OPERATING INSTRUCTIONS
000120                          *
000121                          *
000122                          *     LOAD AND START (OR RESTART) THE PROGRAM. THE PROGRAM IDENTIFICATION WILL
000123                          *     BE DISPLAYED ON THE CONSOLE.  THE INITIAL START WILL ALSO DISPLAY:
000124                          *
000125                          *       THE ZV$LIB REVISION NUMBER
000126                          *       THE ADDRESS FORM (SAF OR LAF)
000127                          *       I/O  EQUIPMENT DETECTED IN THE SYSTEM
000128                          *       MEMORY SIZE
000129                          *
000130                          *     THIS DISPLAY MUST BE VERIFIED BY THE OPERATOR.   THIS DISPLAY IS OMITTED
000131                          *     ON RESTARTS.
000132                          *
000133                          *     THE CONSOLE SEARCH RULES ARE:  FIND THE CONSOLE WITH THE LOWEST  CHANNEL
000134                          *     NUMBER CONNECTED THRU AN  MDC  CONTROLLER.  IF THERE IS NO CONSOLE ON AN
000135                          *     MDC, THEN SEARCH FOR A TERMINAL WITH THE HIGHEST CHANNEL NUMBER ASSIGNED
000136                          *     TO AN ACLA ADAPTER ON AN MLC CONTROLLER.  IF NO ASYNC ADAPTER  IS FOUND,
000137                          *     THEN GO TO THE FULL CONTROL PANEL.
000138                          *
000139                          *     THERE ARE THREE CONSOLE CHANNEL OPTIONS DETERMINED BY THE VALUE  OF  LO-
000140                          *     CATION "ZV$TTY".
000141                          *
000142                          *     IF ZV$TTY EQUALS (0000), SEARCH FOR A CONSOLE.
000143                          *     IF ZV$TTY EQUALS (FFFF), ASSUME THERE IS NO CONSOLE.
```

```
000144                  *     IF ZV$TTY EQUALS NEITHER (0000), NOR (FFFF), THEN IT IS THE CONSOLE CHAN-
000145                  *     NEL NUMBER.   NOTE:  DEFAULT IS TO SEARCH FOR A CONSOLE.
000146                  *
000147                  *     ALL CONSOLE I/O IS EVEN PARITY.  IF CONSOLE IS ON MLC,  IT MUST BE ASYNC
000148                  *     AND THE BAUD RATE SET AT 1200 TO MATCH THE PROGRAM SUPPLIED RATE.  IF IT
000149                  *     IS NECESSARY TO CHANGE THE PROGRAM BAUD RATE,   THEN THE  NEW  BAUD RATE
000150                  *     CODE SHOULD BE PUT INTO LOCATION "ZV$BUD" IN HEX. THE TERMINAL BAUD RATE
000151                  *     MUST BE  SET  TO MATCH THIS NEW BAUD RATE.  THE CORRECT HEX VALUE MAY BE
000152                  *     OBTAINED FROM THE FOLLOWING TABLE.
000153                  *
000154                  *-----------------------------------------*
000155                  *                                         *
000156                  *        BAUD  RATE  TABLE                *
000157                  *                                         *
000158                  *-----------------------------------------*
000159                  * ACLA I.D.          (2118)(2110)        (2108)
000160                  * BAUD-RATE
000161                  *    50                  0                  1
000162                  *    75                  1                  2
000163                  *   110                  2                  3
000164                  *   134                  3                  4
000165                  *   150                  4                  5
000166                  *   200                  5                 ---
000167                  *   300                  6                  6
000168                  *   600                  7                  7
000169                  *   900                 ---                 8
000170                  *  1050                  8                 ---
000171                  *  1200                  9                  9
000172                  *  1800                 10 (A)             10 (A)
000173                  *  2000                 11 (B)             ---
000174                  *  2400                 12 (C)             11 (B)
000175                  *  3600                 ---                12 (C)
000176                  *  4800                 13 (D)             13 (D)
000177                  *  7200                 ---                14 (E)
000178                  *  9600                 14 (E)             15 (F)
000179                  * 19200                 15 (F)             ---
000180                  *
000181                  *     TO MAKE ANY OF THE ABOVE CHANGES,  LOAD AND HALT THE PROGRAM BEFORE EX-
000182                  *     ECUTION.  INSERT CHANGE THEN EXECUTE.  MEMORY LOCATIONS OF "ZV$TTY" AND
000183                  *     "ZV$BUD" MAY BE FOUND IN MAP AT END OF LISTING.
000184                  *     CONSULT LEVEL-6 T&V MANUAL "AW94" FOR DETAILS ON HOW TO LOAD THE TESTS.
000185                  *
000186                  *     THE FOLLOWING IS A TYPICAL RESULT OF LOADING  AND  STARTING TO  RUN THE
000187                  *     PROGRAM.
000188                  *
000189                  *     CLA TEST <PGM NAME> <PGM DATE> <PGM REV>
000190                  *     ZV$LIB REV. 6.0
000191                  *     ZV$AF= 1  <2>
000192                  *     WDT
```

```
000193                          *      CHAN  DEVC  ID
000194                          *      0400  DSKT  2010
000195                          *      0480  DSKT  2010
000196                          *      0580  CDR   2008
000197                          *      1200  DISC  2330
000198                          *      1280  DISC  2330
000199                          *      1300  LPT   2000
000200                          *      1380  CONS  2019
000201                          *      MEMORY LOW  00002B2D
000202                          *      MEMORY HIGH 00003FFF  16K
000203                          *
000204                          *  THE PROGRAM WILL THEN ASK:
000205                          *
000206                          *              POWER LINE FREQ. ?:
000207                          *
000208                          *   THE OPERATOR NOW ENTERS THE POWER LINE FREQUENCY IN DECIMAL
000209                          *   FOLLOWED BY A CARRIAGE RETURN.  THE DEFAULT IS 60 HERTZ,
000210                          *   IF THIS IS DESIRED A CARRIAGE RETURN IS ENTERED.
000211                          *
000212                          *   NEXT THE PROGRAM WILL DISPLAY:
000213                          *
000214                          *             CLA CHANNEL(S) ?:
000215                          *
000216                          *   AT THIS TIME THE CHANNEL(S) OF THE LINE(S) YOU WANT
000217                          *   TO TEST SHOULD BE ENTERED,FOLLOWED BY A CARRIAGE RETURN.
000218                          *   THE CLA CHANNEL(S) WILL NORMALLY BE DISPLAYED AS AN ITEM(S) IN THE
000219                          *   CONFIGURATION PRINTOUT.
000220                          *
000221                          *   THE PROGRAM WILL THEN RESPOND WITH:
000222                          *
000223                          *                         NEXT ?:
000224                          *
000225                          *   THE OPERATOR HAS TO ENTER ONE OF THE FOLLOWING LETTERS:
000226                          *
000227                          *           "A" - RUN LINE(S) WITH INTERNAL LOOP TESTS
000228                          *
000229                          *           "C" - RUN LINE(S) WITH CONNECTOR LOOP TESTS
000230                          *
000231                          *#         "D" - RUN LINE(S) WITH DIRRECT CONNECT LOOP TESTS
000232                          *#
000233                          *#         "M" - RUN LINE(S) WITH MODEM LOOPBACK TESTS; USE EXTERNAL CLOCK
000234                          *#
000235                          *#         "L" - RUN ECHO BACK CCP
000236                          *#
000237                          *   IN ORDER TO RUN WITH "C" SELECTED, AN EXTERNAL LOOP CONNECTOR
000238                          *   MUST BE ATTACHED TO THE LINE ADAPTER(S) EITHER AT THE
000239                          *   LINE ADAPTER CONNECTOR OF THE END OF THE CABLE.  THE REQUIRED
000240                          *   EXTERNAL LOOP CONNECTOR AND INSTRUCTIONS FOR ITS USE ARE
000241                          *   DESCRIBED IN EACH LINE ADAPTER'S PRODUCT MANUAL.
```

```
000242                          *
000243                          *     TO TEST A DIRECT CONNECT LINK, FUNCTION "D" MUST BE ENTERED
000244                          *     ON THE SYSTEM THAT SUPPLIES THE CLOCK FOR THE LINK. PRIOR TO THAT
000245                          *     TIME, "L" MUST BE ENTERED ON THE OTHER SYSTEM.
000246                          *
000247                          *     "M" MODE OF OPERATION IS DESIGNED  TO WORK CORRECTLY WITH
000248                          *   PARADYNE LSI 96 MODEMS. THESE MODEMS MUST HAVE STRUPS C & D INSERTED
000249                          *     PRIOR TO EXECUTING THIS TEST, THE LOCAL OR REMOTE MODEM MUST BE
000250                          *     PLACED INTO LOOPBACK MODE.
000251                          *
000252                          *
000253                          *   IF THERE  IS NO CONSOLE PRESENT REFER TO "SERIES 60
000254                          *   LEVEL 6 MODEL 34/36 T & V OPERATING INSTRUCTION
000255                          *   MANUAL" ORDER NO. FL39 FOR INSTRUCTIONS ON ENTERING DATA
000256                          *   AND INTERPRETING PROGRAM MESSAGES.
000257                          *
000258                          *   THE PROGRAM WILL RUN APPROXOMATELY 1 MINUTE PER ASYNCHRONOUS LINE AND
000259                          *   30 SECONDS PER SYNCHRONOUS LINE.
000260                          *
000261                          *   DURING THE LINE SPEED TEST THE PROGRAM WILL PRINT OUT THE SPEED AT WHICH
000262                          *   THE SYNCHRONOUS LINE(S) ARE TESTED AT WITHIN 3% OF THE ACTUAL SPEED.
000263                          *   ASYNCHRONOUS LINE(S) ARE TESTED AT ALL SELECTABLE BAUD RATES.
000264                          *
000265                          *   IF THERE ARE NO FAULTS. IT WILL THEN DISPLAY:
000266                          *
000267                          *        PASS
000268                          *
000269                          *   THE PROGRAM WILL CONTINUE AND TYPE "PASS" WITH NO HALTS BETWEEN PASSES.
000270                          *   TO RESTART: START THE PROGRAM AT HEX 100
000271                          *
000272                          *
000273                          * ERROR REPORTS
000274                          *
000275                          *   ERRORS WILL CAUSE THE PROGRAM TO HALT. AN ERROR MESSAGE WILL
000276                          *   BE DISPLAYED IF A CONSOLE IS PRESENT.
000277                          *
000278                          *   ERROR DISPLAYS ARE AS FOLLOWS:
000279                          *
000280                          *            ERR XXYP @ ZZZZ
000281                          *
000282                          *            B7  B6  B5  B4  B3  B2  B1  I
000283                          *            R7  R6  R5  R4  R3  R2  R1  M
000284                          *
000285                          *        XX   = MB(MOTHER BOARD) OR DB(DAUGHTER BOARD)
000286                          *        Y    = TEST TYPE
000287                          *        P    = LINE NUMBER
000288                          *        ZZZZ = ERROR LOCATION IN LISTING. HAS COMMENT
000289                          *               GIVING FAILING FUNTION.
000290                          *
```

```
000291                          *         B1-B7, I, R1-R7, M ARE  CONTENTS OF REGISTERS.
000292                          *         INTERPERTATION OF THESE IS FOR SPECIALIST USAGE.
000293                          *
000294                          *         IN ALL CASES:
000295                          *
000296                          *              R3 = CHANNEL NUMBER
000297                          *
000298                          *         IN GENERAL
000299                          *
000300                          *              R6 = SHOULD BE DATA
000301                          *              R5 = ACTUAL DATA
000302                          *              R7 = WORD NUMBER IN BLOCK TRANSFER
000303                          *
000304                          *
000305                          ********************************************************************************
```

```
000306                                  /
000307          0000                    ZERO    EQU     $
000308                                          XLOC    ZHNTSA,ZHTH15,ZV$HR
000309                                          XLOC    ZHISAZ,ZHIAFB
000310                                          XLOC    ZHWDTC,ZHRTCC,ZV$TTY
000311                                          XLOC    ZHRTCI,ZHRTCL
000312                                          XLOC    ZHPFR,ZHCOMM
000313  00FF                                    ORG     ZERO+X'FF'
000314  00FF    0000                    STOP    HLT                             NO ERROR FOUND IN MLCC
000315                                  *
000316                                  *               OPERATOR MUST ENTER CLA ADDRESS
000317                                  *
000318          0100                    STRT    EQU     $
000319  0100    0F8C                            B       >STP                    BRANCH TO NORMAL RUN
000320  0105                                    ORG     ZERO+X'105'
000321                                          CALL    ZV$RD,ZV$IZ             INITIALIZE FOR NON-CONSOLE USE
        0105    FBF0 0001
        0107    D380 0000       X
000322  0109    8700 0000       X               CL      <ZV$TTY                 CLEAR CONSOLE CHANNEL
000323  010B    0F87                            B       >NTYR
000324                                  STP     CALL    ZV$RD,MESG
        010C    FBC0 0003
        010E    D380 0000       X
        0110    0F80
        0111    1349
000325  0112    8980 115D               NTYR    CMZ     <FWFG
000326  0114    09AC                            BNE     >EDLINT
000327                                  *
000328                                  *           ASK FOR POWER LINE FREQ.
000329                                  *
000330  0115    8C51                            STS     =$R1                    ASK ONLY ON 6/36 AND 3/34
000331  0116    82D1                            LB      =$R1,=Z'2000'
        0117    2000
000332  0118    050D                            BBT     >CP40
000333                                          CALL    ZV$QC,MESG9
        0119    FBC0 0003
        011B    D380 0000       X
        011D    0F80
        011E    137E
000334                                          CALL    ZV$ID,HRTZ
        011F    FBC0 0003
        0121    D380 0000       X
        0123    0F80
        0124    115F
000335                                  *
000336                                  *           FIND NUMBER OF MILSEC. IN RTC TICK
000337                                  *
000338  0125    8756                    CP40    CL      =$R6
```

```
000339  0126  F870 C350                    LDR   $R7,=Z'C350'
000340  0128  F300 115F                    DIV   $R7,<HRTZ
000341  012A  EF00 116B                    STR   $R6,<TLOC
000342  012C  8756                         CL    =$R6
000343  012D  F370 0064                    DIV   $R7,=100
000344  012F  FF00 1157                    STR   $R7,<DIV1
000345  0131  70D0                         DOR   $R7,16
000346  0132  F370 000A                    DIV   $R7,=10
000347  0134  FF00 1158                    STR   $R7,<DIV2
000348  0136  EF00 1159                    STR   $R6,<DIV3
000349  0138  8756                         CL    =$R6
000350  0139  F800 116B                    LDR   $R7,<TLOC
000351  013B  7F0A                         MLV   $R7,=10
000352  013C  F300 115F                    DIV   $R7,<HRTZ
000353  013E  FF00 115A                    STR   $R7,<DIV4
000354                              *
000355                              *
000356                              *
000357                              *
000358                              *           INITIALIZE  THE ACTIVE LINE TABLE
000359                              *              INITIALIZE THE LINE ADDRESS TABLE
000360                              *
000361                              EDLINT CALL   ZV$F,ATLT,CPFLG+3,C16
        0140  FBC0 0003
        0142  D380 0000        X
        0144  0F80
        0145  11CB
        0146  11DF
        0147  12A5
000362  0148  FE00 114F                    SWR   $R7,<CHANL
000363                              *       CALL  ZV$QC,MESG7
        014A  FBC0 0003
        014C  D380 0000        X
        014E  0F80
        014F  1373
000364                              *       CALL  ZV$IH,ADDRS,C8
        0150  FBC0 0003
        0152  D380 0000        X
        0154  0F80
        0155  11D3
        0156  12A4
000365  0157  8752                         CL    =$R2                CLEAR INDEX
000366  0158  AB80 11D3                    LAB   $B2,<ADDRS          LOAD $B2 WITH ADDRESS OF ACTIVE LINE TABLE
000367  015A  CB80 12A8                    LAB   $B4,<TSA1           SET UP FOR TRAP ON
000368  015C  CF80 0000        X           STB   $B4,<ZHNTSA         UNAVAILIABLE RESOURCE
000369  015E  CB80 0192                    LAB   $B4,<NODEVF
000370  0160  CF80 0000        X           STB   $B4,<ZHTH15
000371  0162  986E                   SALT  LDR   $R1,$B2,+$R2        LOAD ADDRESS FROM TABLE
000372  0163  1931                         BEZ   $R1,>CON3
```

```
000373  0164  9570 FFC0                AND     $R1,=Z'FFC0'        STRIP CHANNEL WITH HEX FFC0
000374  0166  9970 03C0                CMR     $R1,=X'3C0'         COMPARE CHANNEL WITH HEX 3C0
000375  0168  0302                     BG      >RCK1               BRANCH IF GREATER
000376  0169  0F96                     B       >DEVNF              CHANNEL NUMBER TO SMALL
000377  016A  1E26                RCK1 ADV     $R1,=X'26'          PUT FUNCTION IN
000378  016B  8055                     IO      =$R5,=$R1           INPUT DEVICE NUMBER
        016C  0051
000379  016D  C855                     LDR     $R4,=$R5
000380  016E  D570 FF00                AND     $R5,=Z'FF00'
000381  0170  D970 2100                CMR     $R5,=Z'2100'        COMPARE DEVICE NUMBER RESPONSE
000382  0172  098D                     BNE     >DEVNF
000383  0173  B851                     LDR     $R3,=$R1            SET $R3 FOR INDEX OF LINE NUMBER
000384  0174  B570 03C0                AND     $R3,=X'3C0'
000385  0176  3047                     SOR     $R3,7
000386  0177  CF30 11CB                STR     $R4,<ATLT.$R3       STORE ID INTO ACTIVE LINE TABLE
000387  0179  9570 FC00                AND     $R1,=Z'FC00'
000388  017B  9F00 114F                STR     $R1,<CHANL
000389  017D  0F80 0162                B       <SALT
000390                             *
000391                             DEVNF CALL    ZV$TC,ERMG1         MLCC NOT FOUND ON THIS ADDRESS
        017F  FBC0 0003
        0181  D380 0000        X
        0183  0F80
        0184  13AB
000392  0185  88D2                     DEC     =$R2
000393  0186  9820 11D3                LDR     $R1,<ADDRS.$R2      STORE ADDRESS THAT FAILED
000394  0188  9F00 116B            PDRS STR     $R1,<TLOC
000395                                 CALL    ZV$TH,TLOC          PRINT ADDRESS THAT FAIL
        018A  FBC0 0003
        018C  D380 0000        X
        018E  0F80
        018F  116B
000396  0190  0F80 0140                B       <EDLINT
000397                             *
000398  0192  8755                NODEVF CL      =$R5                TRAP HANDLER ROUTINE
000399  0193  0003                     RTT
000400                             *
000401                             *
000402                             *
000403                             *
000404                             *       CHECK TO SEE IF THIS IS A CLA I CAN TEST
000405                             *
000406  0194  8751                CON3 CL      =$R1                CLEAR INDEX
000407  0195  AB80 11CB                LAB     $B2,<ATLT
000408  0197  9B80 11C4                LAB     $B1,<BITO
000409  0199  2CF6                CAR  LDV     $R2,=-10
000410  019A  C85E                RON  LDR     $R4,$B2.+$R1
000411  019B  1D09                     CMV     $R1,=9
000412  019C  0910                     BE      >CON3A
```

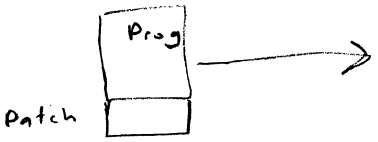
```
000413  019D  497D                        BEZ     $R4,>RON
000414  019E  C921                  CYN   CMR     $R4,$B1.$R2
000415  019F  097A                        BE      >CAR
000416  01A0  27FE                        BINC    $R2,>CYN
000417                                    CALL    ZV$TC,ERMG2
        01A1  FBC0 0003
        01A3  D380 0000       X
        01A5  0F80
        01A6  13B8
000418  01A7  88D1                        DEC     =$R1
000419  01A8  1007                        SOL     $R1,7
000420  01A9  9400 114F                   OR      $R1,<CHANL
000421  01AB  0FDD                        B       >PDRS
000422                              *
000423  01AC  2CF4                  CON3A LDV     $R2,=-12          NUMBER OF I/O TO BE CHANGED
000424  01AD  9800 11D3                   LDR     $R1,<ADDRS        LOAD $R1 WITH ADDRESS OF CLA ACTIVE
000425  01AF  DB80 1149                   LAB     $B5,<FID+1        PUT ADDRESS OF CONTROL TABLE IN $B5
000426  01B1  C870 003F             ALL   LDR     $R4,=X'3F'        LOAD MASK TO CLEAR CHANNEL
000427  01B3  9570 FC00                   AND     $R1,=Z'FC00'      CLEAR SUBCH. & FUNCTION
000428  01B5  C525                        AND     $R4,$B5.$R2       CLEAR CHANNEL
000429  01B6  9454                        OR      $R1,=$R4          PUT CHANNEL NUMBER IN
000430  01B7  9F25                        STR     $R1,$B5.$R2       STORE IT BACK IN CONTROL TABLE
000431  01B8  27F9                        BINC    $R2,>ALL
000432  01B9  8700 1165                   CL      <PSFG             CLEAR PRINT SYNC. SPEED FLAG
000433  01BB  8700 115E                   CL      <HGSPD            CLEAR HIGH SPEED FLAG
000434  01BD  F900 114F                   CMR     $R7,<CHANL        CHECK TO ONLY PRINT ONCE
000435  01BF  091C                        BE      >INVD
000436                              *
000437                              *           PRINT FIRMWARE REV.
000438                              *
000439  01C0  3001                        SOL     $R3,1             SHIFT $R3 TO ALIGN WITH CHANNEL NUMBER
000440  01C1  C380 0D0A                   LNJ     $B4,<GENITZ       INITIALIZE CONTROLLER
000441  01C3  E380 0D75                   LNJ     $B6,<RDLCT        READ LOCATION 1 IN LCT TO PRINT REV.
000442  01C5  0000                        DC      0                 ADDRESS
000443  01C6  0002                        DC      2                 RANGE
000444                              *
000445  01C7  8780 1173                   CLH     <LCTV
000446  01C9  C800 1173                   LDR     $R4,<LCTV
000447  01CB  CF00 1226                   STR     $R4,<FRIMRV       STORE FIRMWARE REV NUMBER
000448                                    CALL    ZV$TC,FREV
        01CD  FBC0 0003
        01CF  D380 0000       X
        01D1  0F80
        01D2  13A1
000449                                    CALL    ZV$TD,LCTV
        01D3  FBC0 0003
        01D5  D380 0000       X
        01D7  0F80
        01D8  1173
```

```
000450  01D9  8A80 115D                    INC    <FWFG              SET FLAG TO INHIBIT PRINT OUT
000451                              *
000452                              *            ASK WHAT MODE OF OPERATION
000453                              *
000454                              INVD   CALL   ZV$QC,MESG6
        01DB  FBC0 0003
        01DD  D380 0000       X
        01DF  0F80
        01E0  1370
000455                                     CALL   ZV$IA,TP2,TLOC
        01E1  FBC0 0003
        01E3  D380 0000       X
        01E5  0F80
        01E6  116D
        01E7  116B
000456  01E8  7C08                        LDV    $R7,=8
000457  01E9  D380 0D63                    LNJ    $B5,<TDLAY         DELAY FOR MLC CONSOLE
000458  01EB  C800 116B                    LDR    $R4,<TLOC
000459  01ED  4048                        SOR    $R4,8              SHIFT MODE TYPE ANSWER OVER
000460                              *##
000461                              *#
000462  01EE  CF00 1149                    STR    $R4,<TEST          SAVE COMMAND CODE
000463                              *#
000464                              *##
000465  01F0  4D53                        CMV    $R4,=X'53'          IF 'S' DO SHORT TEST ONLY
000466  01F1  0900 0254                    BE     <SHTST
000467  01F3  4D43                        CMV    $R4,=X'43'          IF 'C' DO CABLE LOOP TEST ONLY
000468  01F4  0900 0265                    BE     <CABLP
000469                              *##
000470                              *#
000471  01F6  4D4D                        CMV    $R4,=X'4D'          IF 'M' DO MODEM LOOPBACK TEST
000472  01F7  0900 0265                    BE     <CABLP
000473  01F9  4D44                        CMV    $R4,X'44'           IF 'D' DO DIRRECT CONNECT LOOP TEST
000474  01FA  0900 0265                    BE     <CABLP
000475  01FC  4D4C                        CMV    $R4,X'4C'           IF 'L' EXECUTE ECHO BACK CCP
000476  01FD  0900 0228                    BE     <LOOPBK
000477  01FF  4D50                        CMV    $R4,=X'50'          IF 'P' GO TO PATCH ROUTINE
000478                              *#
000479                              *##
000480  0200  0900 0222                    BE     <PCH
000481  0202  4D41                        CMV    $R4,=X'41'          IF 'A' DO INTERNAL LOOP TEST ONLY
000482  0203  09D8                        BNE    >INVD
000483                              *
000484  0204  8700 1152                    CL     <COUNT
000485  0206  8700 116F                    CL     <SHORT
000486  0208  C870 C714              RTST  LDR    $R4,=Z'C714'       STORE LINE CONTROL FOR INTERNAL LOOP
000487  020A  CF00 12D4                    STR    $R4,<LC1LCR
000488  020C  A380 02A6                    LNJ    $B2,<LPTA          TEST A:  GO TO LOOP TEST
000489  020E  A380 034A                    LNJ    $B2,<SPSYTC        TEST B:  GO TO LINE SPEED TEST
```

```
000490  0210  A380 04A7                    LNJ    $B2,<ADRTST          TEST C:  GO TO BART AND BARF TEST
000491  0212  A380 04E4                    LNJ    $B2,<CRPTB           TEST D:  GO TO CRC AND PARITY TEST
000492  0214  A380 05F8                    LNJ    $B2,<CHSZTD          TEST E:  GO TO CHARACTER SIZE TEST
000493  0216  A380 067C                    LNJ    $B2,<STBITE          TEST F:  GO TO STOP BIT TEST
000494  0218  A380 0705                    LNJ    $B2,<PARERF          TEST G:  GO TO PARITY ERROR TEST
000495  021A  A380 0796                    LNJ    $B2,<STIOG           TEST H:  GO TO STOP I/O TEST
000496  021C  A380 0891                    LNJ    $B2,<UORNTH          TEST J:  GO TO OVERRUN AND UNDERRUN TEST
000497  021E  A380 0E04                    LNJ    $B2,<EDPCK
000498  0220  0F80 0208                    B      <RTST
000499                              *
000500                              PCH     CALL   ZV$PCH
        0222  FBF0 0001
        0224  D380 0000        X
000501  0226  0F80 01DB                    B      <INVD
000502                              *##
000503                              *#
000504  0228  8F00 119A            LOOPBK  SAVE   <SAV3,=Z'0112'       EXECUTE ECHO BACK CCP
        022A  0112
000505  022B  8753                        CL     =$R3
000506  022C  D380 0D4F                    LNJ    $B5,<FLN             FIND LINE
000507  022E  11BD                        DC     <SYNC
000508  022F  0FA3                        B      >LOOPER              NO SYNC LINE RETURN
000509  0230  C380 0D0A                    LNJ    $B4,<GENITZ          GENERAL INITIALIZATION
000510  0232  C570 FF80                    AND    $R4,=Z'FF80'         $R4 < CHANNEL NUMBER
000511  0234  C470 0040                    OR     $R4,=X'40'           OR IN DIRECTION BIT
000512  0236  CF00 1151                    STR    $R4,<CHNZ
000513                                     CALL   ZV$MLW,LOPCCP,LOPSZE,HX200,CHNZ LOAD CCP INTO MLCP
        0238  FBC0 0003
        023A  D380 0000        X
        023C  0F80
        023D  110F
        023E  1133
        023F  12A6
        0240  1151
000514  0241  D380 0D3B                    LNJ    $B5,<SETLCT          SET UP LCT'S
000515  0243  1134                        DC     <LOPLCT
000516  0244  C800 113F                    LDR    $R4,<FOCC            START I/O ON RECEIVE CHANNEL
000517  0246  C380 0D95                    LNJ    $B4,<CGSCH
000518  0248  8000 11C5                    IO     <BIT1,=$R4
        024A  0054
000519  024B  0703                        BIOT   >LOOPB1
000520  024C  F380 0DBA                    LNJ    $B7,<ERMB            ERROR:  I/O WAS NAK'D
000521  024E  0000                  LOOPB1  HLT
000522                                     XLOC   STRTD2
000523  024F  D380 0000        X           LNJ    $B5,<STRTD2          START GOLDMP
000524  0251  0FFD                        B      >LOOPB1
000525  0252  F380 0DBA            LOOPER  LNJ    $B7,<ERMB            ERROR:  NOT SYNC LINE
000526                              *#
000527                              *##
```

```
000528                                  *
000529  0254  C870 C714                  SHTST   LDR     $R4,=Z'C714'
000530  0256  CF00 12D4                          STR     $R4,<LC1LCR             STORE LINE CONTROL FOR INTERNAL LOOP
000531  0258  CF00 116F                          STR     $R4,<SHORT              SET FLAG
000532  025A  A380 02A6                          LNJ     $B2,<LPTA               TEST A:  GO TO LOOP TEST
000533  025C  A380 034A                          LNJ     $B2,<SPSYTC             TEST B: LINE SPEED TEST TO SET HIGH SPEED FLA
000534  025E  A380 04E4                          LNJ     $B2,<CRPTB              TEST D: CRC AND PARITY TEST
000535  0260  A380 0705                          LNJ     $B2,<PARERF             TEST G:  PARITY ERROR TEST
000536  0262  A380 0E04                          LNJ     $B2,<EDPCK              PRINT PASS MESSAGE
000537  0264  0FF0                              B       >SHTST
000538                                  *
000539                                  *##
000540                                  *#
000541  0265  8700 114A                  CABLP   CL      <MLXCT                  CLEAR MODEM LOOPBACK XMIT CTRS
000542  0267  8700 114B                          CL      <MLECT
000543                                  *#
000544                                  *##
000545  0269  4C01                              LDV     $R4,=1
000546  026A  CF00 1152                          STR     $R4,<COUNT
000547  026C  8700 116F                          CL      <SHORT
000548  026E  C870 C314                  RTST1   LDR     $R4,=Z'C314'            STORE LINE CONTROL FOR CONNECTOR LOOP
000549  0270  CF00 12D4                          STR     $R4,<LC1LCR
000550                                  *
000551                                  *##
000552                                  *#
000553  0272  A380 0956                          LNJ     $B2,<LPCBLK             TEST K:  GO TO LOOP DSS IN DSC
000554  0274  4C32                              LDV     $R4,50
000555  0275  CF00 114C                          STR     $R4,<COUNT1
000556  0277  A380 02A6                  RTST3   LNJ     $B2,<LPTA               TEST A:  GO TO LOOP DATA TEST
000557  0279  8880 114C                          DEC     <COUNT1
000558  027B  8980 114C                          CMZ     <COUNT1
000559  027D  09FA                              BNE     >RTST3
000560                                  *#
000561                                  *##
000562  027E  A380 0A1C                          LNJ     $B2,<XMMSL              TEST L:  GO TO XMIT MARK & SPACE TEST
000563  0280  8880 1152                          DEC     <COUNT
000564  0282  8980 1152                          CMZ     <COUNT
000565  0284  09EA                              BNE     >RTST1
000566                                  *##
000567                                  *#
000568  0285  C800 1149                          LDR     $R4,<TEST
000569  0287  4D4D                              CMV     $R4,X'4D'
000570  0288  099B                              BNE     >RTST2                  BRANCH IF NOT MODEM LOOPBACK TEST
000571                                          CALL    ZV$TH,ZV$TD,MLXCT,MLCTRL PRINT ERROR SUMMARY
        0289  FBC0 0003
        028B  D380 0000           X
        028D  0F80
        028E  114A
        028F  13DC
```

```
000572                                      CALL   ZV$T,MLMSG1
        0290  FBC0 0003
        0292  D380 0000      X
        0294  0F80
        0295  13CB
000573                                      CALL   ZV$TH.ZV$TD,MLECT
        0296  FBC0 0003
        0298  D380 0000      X
        029A  0F80
        029B  114B
000574                                      CALL   ZV$T,MLMSG2
        029C  FBC0 0003
        029E  D380 0000      X
        02A0  0F80
        02A1  13D8
000575  02A2  0FC3                         B      >CABLP
000576                              *#
000577                              *##
000578  02A3  A380 0E04            RTST2  LNJ    $B2,<EDPCK
000579  02A5  0FC0                         B      >CABLP
000580                              *
000581                              *
000582                              *          DATA LOOP TEST
000583                              *
000584  02A6  C870 4120            LPTA   LDR    $R4,=A'A '
000585  02A8  CF00 13A9                    STR    $R4,<ERMG+1
000586  02AA  C870 0F18                    LDR    $R4,=X'F18'          STORE FAST LINE SPEED
000587  02AC  CF00 12D3                    STR    $R4,<LCT1S
000588  02AE  B380 0BA6                    LNJ    $B3,<CLFG            CLEAR ALL TEST FLAGS
000589  02B0  8700 1160                    CL     <MASK                SET MASK FOR ZV$C
000590  02B2  8700 12D5                    CL     <LC1SYF              CLEAR SYNC. FLAG FOR ASYNC. LINES
000591  02B4  7C06                         LDV    $R7,=6               NUMBER OF 30 MS DELAY
000592                              *
000593                              *            TEST ASYNCHRONOUS LINES WITH OLD BAUD RATES
000594                              *               AT 9600 BAUD, WITH 2 STOP BITS
000595                              *
000596  02B5  9B80 11BA                    LAB    $B1,<ASYN            STORE ADDRESS OF LINE TYPE
000597  02B7  B380 0A8C                    LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
000598  02B9  0F85                         B      >T1
000599  02BA  131D                         DC     <WCP1                ADDRESS TABLE FOR BLOCK WRITE
000600  02BB  12CF                         DC     <LCT1                LCT TABLE
000601  02BC  003F                         DC     (DFLT-DOUT)          RANGE IN WORDS
000602  02BD  11E3                         DC     <DOUT                TRANSMIT BUFFER
000603                              *
000604                              *            TEST ASYNCHRONOUS LINES WITH NEW BAUD RATES
000605                              *               AT 19200 BAUD, WITH 2 STOP BITS
000606                              *
000607  02BE  9B80 11BB            T1     LAB    $B1,<ASYN1           LINE TYPE
000608  02C0  B380 0A8C            T1A    LNJ    $B3,<DTALOP
```



Delete DCMVA from
Copy DCMVA from
RUN prog DGH2

```
000609  02C2  0F85                    B       >T2AA
000610  02C3  131D                    DC      <WCP1
000611  02C4  12CF                    DC      <LCT1
000612  02C5  003F                    DC      (DFLT-DOUT)          RANGE IN WORDS
000613  02C6  11E3                    DC      <DOUT                TRANSMIT BUFFER ADDRESS
000614                          *
000615                          *           TEST CURRENT LOOP ASYNCHRONOUS
000616                          *
000617  02C7  8980 1152         T2AA   CMZ     <COUNT               IF CONNECTOR  LOOP
000618  02C9  0900 02CF                 BE      <T2                  CHANGE SPEED TO TEST AT 9600 BAUD
000619  02CB  C870 0E18                 LDR     $R4,=X'E18'
000620  02CD  CF00 12D3                 STR     $R4,<LCT1S
000621  02CF  C871               T2    LDR     $R4,+$B1             LOAD ID TYPE INTO $R4
000622  02D0  C900 11BC                 CMR     $R4,<ASYNC           CHECK TO SEE IF ALREADY DONE
000623  02D2  0902                      BE      >T2A
000624  02D3  0FED                      B       >T1A
000625        02D4               T2A    EQU     $
000626  02D4  8980 116F                 CMZ     <SHORT
000627  02D6  0980                      BNE     >NCBL
000628  02D7  C870 0118                 LDR     $R4,=X'118'          CHANGE BAUD RATE TO 50
000629  02D9  CF00 12D7                 STR     $R4,<LCT1S
000630  02DB  7C78                      LDV     $R7,=120             NUMBER OF 30 MS DELAY FOR DATA TRANSFER
000631                          *
000632                          *           TEST ASYNCHRONOUS LINES WITH OLD BAUD RATES
000633                          *              AT 50 BAUD, WITH 2 STOP BITS
000634                          *
000635  02DC  9B80 11BA                 LAB     $B1,<ASYN            LOAD LINE TYPE
000636  02DE  B380 0A8C                 LNJ     $B3,<DTALOP          LOOP TEST SUB-ROUTINE
000637  02E0  0F85                      B       >T3
000638  02E1  131D                      DC      <WCP1                ADDRESS TABLE FOR BLOCK WRITE
000639  02E2  12CF                      DC      <LCT1                LCT TABLE
000640  02E3  0007                      DC      7                    RANGE IN WORDS
000641  02E4  11F3                      DC      <DOUT1
000642                          *
000643                          *           TEST ASYNCHRONOUS LINES WITH NEW BAUD RATE
000644                          *              AT 50 BAUD, WITH  2 STOP BITS
000645                          *
000646  02E5  C870 0018          T3     LDR     $R4,=X'18'
000647  02E7  CF00 12D3                 STR     $R4,<LCT1S
000648  02E9  9B80 11BB                 LAB     $B1,<ASYN1           LOAD LINE TYPE
000649  02EB  B380 0A8C          T3A    LNJ     $B3,<DTALOP          LOOP SUB-ROUTINE.
000650  02ED  0F85                      B       >T4
000651  02EE  131D                      DC      <WCP1
000652  02EF  12CF                      DC      <LCT1
000653  02F0  0007                      DC      7
000654  02F1  11F3                      DC      <DOUT1
000655                          *
000656                          *           TEST CURRENT LOOP ASYNCHRONOUS
000657                          *
```

```
000658  02F2  C871           T4      LDR     $R4,+$B1            LOAD ID TYPE INTO $R4
000659  02F3  C900 11BC              CMR     $R4,<ASYNC          CHECK TO SEE IF ALREADY DONE
000660  02F5  0902                   BE      >T4A
000661  02F6  0FF5                   B       >T3A
000662        02F7           T4A     EQU     $
000663  02F7  C800 12D4              LDR     $R4,<LC1LCR         CHECK LINE CONTROL FOR CONNECTOR LOOP
000664  02F9  82D4                   LB      =$R4,=Z'0400'       IF CONNECTOR LOOP SET DIRECT CONNECT BIT
        02FA  0400
000665  02FB  050B                   BBT     >NCBL
000666                       *##
000667                       *#
000668  02FC  C800 1149              LDR     $R4,<TEST
000669  02FE  4D4D                   CMV     $R4,X'4D'
000670  02FF  C870 C314              LDR     $R4,=Z'C314'
000671  0301  0903                   BE      >NCBL1              DON'T SET DIRECT CONNECT IF MODEM LOOP
000672                       *#
000673                       *##
000674  0302  C870 CB14              LDR     $R4,=Z'CB14'
000675  0304  CF00 12D4      NCBL1   STR     $R4,<LC1LCR
000676  0306  C870 3A18      NCBL    LDR     $R4,=X'3A18'        PUT IN SYNC. CHARACTER TO TEST
000677  0308  CF00 12D3              STR     $R4,<LCT1S          SYNC. LINE TYPES
000678  030A  C870 021C              LDR     $R4,=X'21C'         SET LCT SYNC. FLAG
000679  030C  CF00 12D5              STR     $R4,<LC1SYF
000680  030E  7C2D                   LDV     $R7,=45             NUMBER OF 30 MS DELAY FOR 800 BAUD
000681                       *##
000682                       *#
000683  030F  C800 1149              LDR     $R4,<TEST           REDUCE DELAY IF 'D' OR 'M' TEST
000684  0311  4D44                   CMV     $R4,X'44'
000685  0312  0903                   BE      >NCBL2
000686  0313  4D4D                   CMV     $R4,X'4D'
000687  0314  0982                   BNE     >NCBL3
000688  0315  7C04           NCBL2   LDV     $R7,=4
000689        0316           NCBL3   EQU     $
000690                       *#
000691                       *##
000692                       *
000693                       *           TEST SYNCHRONOUS (3A AS SYNC. CHAR.)
000694                       *
000695  0316  9B80 11BD              LAB     $B1,<SYNC           LOAD LINE TYPE
000696  0318  B380 0A8C      QA1     LNJ     $B3,<DTALOP         LOOP SUB-ROUTINE
000697  031A  0F85                   B       >T5
000698  031B  131D                   DC      <WCP1
000699  031C  12CF                   DC      <LCT1
000700  031D  002F                   DC      (DFLT-DOUT1)
000701  031E  11F3                   DC      <DOUT1
000702                       *
000703                       *           TEST BI-SYNCHRONOUS LINES (3A AS SYNC. CHAR.)
000704                       *
000705  031F  C871           T5      LDR     $R4,+$B1            LOAD ID TYPE INTO $R4
```

```
000706  0320  C900 11BE                CMR    $R4,<BISYNC          CHECK TO SEE IF ALREADY DONE
000707  0322  0902                     BE     >T6
000708  0323  0FF5                     B      >QA1
000709                           *
000710                           *           TEST MIL 188 SYNCHRONOUS (3A AS SYNC. CHAR.)
000711                           *
000712  0324  9B80 11BF           T6    LAB    $B1,<SYN188          LOAD LINE TYPE
000713  0326  B380 0A8C                LNJ    $B3,<DTALOP          LOOP SUB-ROUTINE
000714  0328  0F85                     B      >T7
000715  0329  131D                     DC     <WCP1
000716  032A  12CF                     DC     <LCT1
000717  032B  002F                     DC     (DFLT-DOUT1)
000718  032C  11F3                     DC     <DOUT1
000719                           *
000720                           *
000721                           *           TEST WIDEBAND 301/303 (3A AS SYNC. CHARACTER)
000722                           *
000723  032D  9B80 11C0           T7    LAB    $B1,<WDB3            LOAD ADDRESS OF LINE TYPE
000724  032F  B380 0A8C           T7A   LNJ    $B3,<DTALOP          LOOP SUB-ROUTINE
000725  0331  0F85                     B      >T29A
000726  0332  131D                     DC     <WCP1
000727  0333  12CF                     DC     <LCT1
000728  0334  003F                     DC     (DFLT-DOUT)
000729  0335  11E3                     DC     <DOUT
000730                           *
000731                           *           TEST BI-SYNC WIDEBAND 301/303 (3A AS SYNC. CHARACTER)
000732                           *
000733  0336  C871                T29A  LDR    $R4,+$B1             LOAD ID TYPE INTO $R4
000734  0337  C900 11C1                CMR    $R4,<WDB3B           CHECK TO SEE IF ALREADY DONE
000735  0339  0902                     BE     >T29
000736  033A  0FF5                     B      >T7A
000737                           *
000738                           *           TEST WIDEBAND V35 (3A AS SYNC. CHAR.)
000739                           *
000740  033B  9B80 11C2           T29   LAB    $B1,<WDBV35          LOAD ADDRESS OF LINE TYPE
000741  033D  B380 0A8C           T29Z  LNJ    $B3,<DTALOP
000742  033F  0F85                     B      >T30A
000743  0340  131D                     DC     <WCP1
000744  0341  12CF                     DC     <LCT1
000745  0342  003F                     DC     (DFLT-DOUT)
000746  0343  11E3                     DC     <DOUT
000747                           *
000748                           *           TEST BI-SYNC WIDEBAND V35 (3A AS SYNC. CHARACTER)
000749                           *
000750  0344  C871                T30A  LDR    $R4,+$B1             LOAD ID TYPE INTO $R4
000751  0345  C900 11C3                CMR    $R4,<WDBV5B          CHECK TO SEE IF ALREADY DONE
000752  0347  0902                     BE     >T30
000753  0348  0FF5                     B      >T29Z
000754                           *
```

```
0349  8382                T30    JMP    $B2                  JUMP OUT OF LOOP TEST
                          *
                          * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                          *
                          *          LINE SPEED TEST
                          *
034A  C870 4220           SPSYTC LDR    $R4,=A'B '
034C  CF00 13A9                  STR    $R4,<ERMG+1
034E  B380 0BA6                  LNJ    $B3,<CLFG            CLEAR ALL TEST FLAGS
0350  C800 115F                  LDR    $R4,<HRTZ            CALCULATE THE LOW AND HIGH RANGE OF RTC TICKS
0352  4002                       SOL    $R4,2
0353  4EF8                       ADV    $R4,=-8
0354  CF00 1224                  STR    $R4,<SECL
0356  4E10                       ADV    $R4,=16
0357  CF00 1225                  STR    $R4,<SECH
0359  C800 115F                  LDR    $R4,<HRTZ            CALCULATE NUMBER OF RTC TICKS IN 5 SEC.
035B  4F0A                       MLV    $R4,=10
035C  CF00 1223                  STR    $R4,<FIVSEC
035E  CB80 0368                  LAB    $B4,<SPSYT1          SET UP INTERRUPT VECTORS AND
0360  CF80 12CA                  STB    $B4,<ISA4P           POINTERS TO CHANGE RUNNING LEVEL TO 4
0362  CB80 12C7                  LAB    $B4,<ISA4+$AF
0364  CF80 0004      X           STB    $B4,<ZHISAZ+4*$AF
0366  8E70 8004                  LEV    =Z'8004'             CHANGE RUNNING LEVEL TO 4
0368  4C01                SPSYT1 LDV    $R4,=1               SET RTC LEVEL TO 1
0369  CF00 0000      X           STR    $R4,<ZHRTCL
                          *
036B  CB80 048C                  LAB    $B4,<NOSPD           SET RUPT POINTER FOR RTC INTERRUPT
036D  CF80 12C4                  STB    $B4,<ISA1P
                          *
036F  CB80 12C1                  LAB    $B4,<ISA1+$AF        SET RUPT VECTOR FOR LEVEL 1 RUPT
0371  CF80 0001      X           STB    $B4,<ZHISAZ+$AF
                          *
0373  8980 116F                  CMZ    <SHORT
0375  099E                       BNE    >STSP1
0376  4C0A                       LDV    $R4,=10              STORE CHAR. SIZE
0377  CF00 1154                  STR    $R4,<CSB
0379  8700 12E5                  CL     <LC3SYF              CLEAR SYNC. FLAG FOR ASYNC. LINES
037B  8700 114E                  CL     <BYTP                SET BYTE POSITION TO ZERO
                          *
                          *     TEST LINE SPEEDS FOR ASYCHRONOUS OLD BAUD RATE
                          *
037D  C870 0118                  LDR    $R4,=X'118'          SET $R4 WITH FIRST SPEED TO TEST (ASYNC.)
037F  9B80 11BA                  LAB    $B1,<ASYN            LOAD LINE TYPE
0381  EB80 1247                  LAB    $B6,<OSPTBL          LOAD ADDRESS OF CHAR. NUMBER TABLE
0383  B380 03CE                  LNJ    $B3,<SPDTLP          GO TO LINE SPEED TEST
                          *
                          *     TEST LINE SPEEDS FOR ASYCHRONOUS NEW BAUD RATE
                          *
0385  4C18                       LDV    $R4,=X'18'           SET $R4 WITH FIRST SPEED TO TEST (ASYNC.)
```

```
000804  0386  9B80 11BB              LAB     $B1,<ASYN1          LOAD LINE TYPE
000805  0388  EB80 1237              LAB     $B6,<NSPTBL         LOAD ADDRESS OF CHAR. NUMBER TABLE
000806  038A  B380 03CE              LNJ     $B3,<SPDTLP
000807                        *
000808                        *       TEST LINE SPEEDS FOR ASYCHRONOUS CURRENT LOOP
000809                        *
000810  038C  4C18                    LDV     $R4,=X'18'
000811  038D  9B80 11BC              LAB     $B1,<ASYNC
000812  038F  EB80 1237              LAB     $B6,<NSPTBL
000813  0391  B380 03CE              LNJ     $B3,<SPDTLP
000814                        *
000815  0393  C870 0F00      STSP1   LDR     $R4,=3840           SET RANGE FOR SYNC. LINES
000816  0395  CF00 1166              STR     $R4,<RANGE
000817  0397  EB80 1166              LAB     $B6,<RANGE          LOAD ADDRESS OF RANGE FOR SPEED TEST SUB-ROUT
000818                        *
000819  0399  CB80 0496              LAB     $B4,<IRRG           SET RUPT POINTER FOR RTC INTERRUPT (SYNC.)
000820  039B  CF80 12C4              STB     $B4,<ISA1P
000821  039D  8A80 114E              INC     <BYTP               SET BYTE POSITION TO ONE
000822                        *
000823  039F  C870 021C              LDR     $R4,=X'21C'         SET LCT SYNC. LINE FLAG
000824  03A1  CF00 12E5              STR     $R4,<LC3SYF
000825  03A3  4C08                    LDV     $R4,=8              SET CHAR. SIZE FOR SPEED CALCULATION
000826  03A4  CF00 1154              STR     $R4,<CSB
000827  03A6  C870 C118              LDR     $R4,=Z'C118'        LOAD $R4 WITH SYNC CHARACTER TO BE USED
000828  03A8  8980 116F              CMZ     <SHORT
000829  03AA  098D                    BNE     >STSP2
000830                        *
000831                        *       MEASURE AND PRINT LINE SPEED FOR BI-SYCHRONOUS
000832                        *
000833  03AB  9B80 11BE              LAB     $B1,<BISYNC         LOAD ADDRESS OF LINE TYPE
000834  03AD  B380 03CE              LNJ     $B3,<SPDTLP         GO TO LINE SPEED TEST
000835                        *
000836                        *       MEASURE AND PRINT LINE SPEED FOR SYCHRONOUS
000837                        *
000838  03AF  9B80 11BD              LAB     $B1,<SYNC           LOAD ADDRESS OF LINE TYPE
000839  03B1  B380 03CE              LNJ     $B3,<SPDTLP         GO TO LINE SPEED TEST
000840                        *
000841                        *       MEASURE AND PRINT LINE SPEED FOR MIL 188
000842                        *
000843  03B3  9B80 11BF              LAB     $B1,<SYN188
000844  03B5  B380 03CE              LNJ     $B3,<SPDTLP         GO TO LINE SPEED TEST
000845                        *
000846                        *           MEASURE AND PRINT LINE SPEED FOR WIDEBAND 301/303
000847                        *
000848  03B7  9B80 11C0      STSP2   LAB     $B1,<WDB3
000849  03B9  B380 03CE              LNJ     $B3,<SPDTLP
000850                        *
000851                        *           MEASURE AND PRINT LINE SPEED FOR BI SYNC WIDEBAND 301/303
000852                        *
```

```
000853  03BB  9B80 11C1                LAB    $B1,<WDB3B
000854  03BD  B380 03CE                LNJ    $B3,<SPDTLP
000855                          *
000856                          *           MEASURE AND PRINT LINE SPEED FOR WIDEBAND V35
000857                          *
000858  03BF  9B80 11C2                LAB    $B1,<WDBV35
000859  03C1  B380 03CE                LNJ    $B3,<SPDTLP
000860                          *
000861                          *           MEASURE AND PRINT LINE SPEED FOR BI-SYNC WIDEBAND V35
000862                          *
000863  03C3  9B80 11C3                LAB    $B1,<WDBV5B
000864  03C5  B380 03CE                LNJ    $B3,<SPDTLP
000865                          *
000866  03C7  8A80 1165                INC    <PSFG                 SET FLAG TO INHIBIT PRINTOUT
000867                                 CALL   ZV$RD                 RE-INITIALIZE INTERRUPT VECTOR AND POINTERS &
        03C9  FBF0 0001
        03CB  D380 0000      X
000868                          *                                       SET LEVEL TO ZERO
000869  03CD  8382                     JMP    $B2                   JUMP OUT OF TEST
000870                          *
000871                          * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
000872                          *
000873                          *           LINE SPEED LOOP
000874                          *
000875  03CE  8F00 119A          SPDTLP SAVE   <SAV3,=Z'0810'        SAVE REGS.
        03D0  0810
000876  03D1  8753                     CL     =$R3                  "   $R3
000877  03D2  9F80 03D6                STB    $B1,<ST1              STORE LINE TYPE FOR FIND LINE ON
000878                          *
000879  03D4  D380 0D4F          JPT3   LNJ    $B5,<FLN              FIND LINE ON OF TYPE BELOW
000880  03D6  0000               ST1    RESV   $AF,0                 LINE TYPE FOUND AND STORED HERE
000881  03D7  8383                     JMP    $B3                   NO MORE LINE THIS TYPE END TEST
000882                          *
000883  03D8  8751                     CL     =$R1                  CLEAR $R1
000884  03D9  8752                     CL     =$R2                  CLEAR INDEX ON NEW LINE
000885  03DA  CF00 12E3          NXSPD  STR    $R4,<LCT3S            STORE LINE SPEED IN LCT TABLE
000886                          *
000887  03DC  C808 03D6          NXSPD1 LDR    $R4,*<ST1
000888  03DE  C900 11BC                 CMR    $R4,<ASYNC
000889  03E0  0983                     BNE    >NXSPD2               B IF NOT CURRENT LOOP ADAPT.
000890  03E1  C380 0C18                 LNJ    $B4,<SCLS             SET CURRENT LOOP ADAPTER  SPEED
000891  03E3  C380 0D0A          NXSPD2 LNJ    $B4,<GENITZ           GENERAL INITIALIZE
000892  03E5  82D4                     LB     =$R4,=X'80'           SEE IF SECOND LINE OF CLA
        03E6  0080
000893  03E7  0583                     BBF    >ST2                  IF SECOND LINEOF CLA
000894  03E8  C380 0C11                 LNJ    $B4,<STMD             SET TEST MODE FOR THE BOARD
000895                          *
000896  03EA  C570 FF80          ST2    AND    $R4,=Z'FF80'          PREPARE  TO WRITE CHANNEL PROG
000897  03EC  C470 0040                 OR     $R4,=X'40'
```

```
000898  03EE  CF00 1151                    STR   $R4,<CHNZ
000899                                     CALL  ZV$MLW,CHLP1,RAG1,HX200,CHNZ
        03F0  FBC0 0003
        03F2  D380 0000       X
        03F4  0F80
        03F5  0E17
        03F6  131F
        03F7  12A6
        03F8  1151
000900                                     CALL  ZV$MLW,CHLP1A,RAG2,HX400,CHNZ
        03F9  FBC0 0003
        03FB  D380 0000       X
        03FD  0F80
        03FE  0E2F
        03FF  1320
        0400  12A7
        0401  1151
000901                        *
000902  0402  D380 0D3B               LNJ   $B5,<SETLCT          SEND OUT LCT TABLE
000903  0404  12DF                    DC    <LCT3
000904                        *
000905  0405  C826                    LDR   $R4,$B6.$R2          LOAD $R4 WITH RANGE IN BYTES
000906  0406  8AD3                    INC   =$R3
000907  0407  D380 0D99               LNJ   $B5,<MCCB            FORM CCB FOR XMIT
000908  0409  11E3                    DC    <DOUT
000909  040A  0060                    DC    X'60'
000910  040B  88D3                    DEC   =$R3                 CHANGE TO RECV. SIDE
000911  040C  4900 047A               BEZ   $R4,<TSNL
000912  040E  CF55                    STR   $R4,=$R5
000913  040F  C970 03C0        MLCCB  CMR   $R4,=X'3C0'          MAKE MULTIPLE CCB IF LARGE RANGE
000914  0411  0A80 041D               BALE  <SGCCB
000915  0413  C870 03C0               LDR   $R4,=X'3C0'
000916  0415  D380 0D99               LNJ   $B5,<MCCB
000917  0417  1C00                    DC    <DIN
000918  0418  0040                    DC    X'40'
000919  0419  88D1                    DEC   =$R1
000920  041A  D254                    SUB   $R5,=$R4
000921  041B  DF54                    STR   $R5,=$R4
000922  041C  0FF3                    B     >MLCCB
000923                        *
000924  041D  D380 0D99        SGCCB  LNJ   $B5,<MCCB            FORM CCB FO RECV.
000925  041F  1C00                    DC    <DIN
000926  0420  0060                    DC    X'60'
000927                        *
000928                        *
000929                        *      START I/O XMIT FIRST THEN RECV.
000930                        *
000931  0421  7C01                    LDV   $R7,=1
000932  0422  8AD3                    INC   =$R3                 CHANGE TO XMIT SIDE
```

```
000933  0423  C800 113F                LDR    $R4,<FOCC
000934  0425  C380 0D95                LNJ    $B4,<CGSCH
000935  0427  8000 11C5                IO     <BIT1,=$R4
        0429  0054
000936  042A  0703                     BIOT   >MELC01
000937  042B  F380 0DBA                LNJ    $B7,<ERMB
000938  042D  D380 0D63          MELC01 LNJ    $B5,<TDLAY
000939  042F  88D3                     DEC    =$R3
000940                           *
000941  0430  88D1                     DEC    =$R1
000942  0431  C800 113F                LDR    $R4,<FOCC
000943  0433  C380 0D95                LNJ    $B4,<CGSCH
000944  0435  F800 1223                LDR    $R7,<FIVSEC      SYNC. UP CLOCK
000945  0437  FF00 0000       X        STR    $R7,<ZHRTCC      STORE RTCC WITH 5 SEC OF TICKS
000946  0439  0004                     RTCN                    TURN  ON REAL TIME CLOCK
000947  043A  F900 0000       X  SYCT CMR    $R7,<ZHRTCC
000948  043C  027E                     BL     >SYCT
000949  043D  8040 0D87                IO     BIT1,=$R4        START I/O
        043F  0054
000950  0440  0703                     BIOT   >JPT1
000951  0441  F380 0DBA                LNJ    $B7,<ERMB        ERROR:  I/O WAS NAK'ED
000952  0443  C380 0D25          JPT1  LNJ    $B4,<INXT        INPUT NEXT STATUS
000953  0445  C800 1144                LDR    $R4,<FIS         LOOP ON INPUT STATUS UNTIL STATUS COMPLETE
000954  0447  C380 0D95                LNJ    $B4,<CGSCH
000955  0449  8055                ST3_   IO     =$R5,=$R4        INPUT STATUS
        044A  0054
000956  044B  0703                     BIOT   >JPT2
000957  044C  F380 0DBA                LNJ    $B7,<ERMB        ERROR:  I/O WAS NAK'ED
000958  044E  82D5                JPT2  LB     =$R5,=Z'1000'    SEE IF STATUS COMPLETE
        044F  1000
000959  0450  05F9                     BBF    >ST3             LOOP UNTIL STATUS COMPLETE
000960  0451  17F2                     BINC   $R1,>JPT1        LOOP TO FINAL CCB
000961                           *
000962  0452  0005                     RTCF                    TURN OFF REAL TIME CLOCK
000963  0453  C800 1223                LDR    $R4,<FIVSEC      CALCULATE ELAPSED TIME
000964  0455  C200 0000       X        SUB    $R4,<ZHRTCC
000965  0457  8755                     CL     =$R5
000966  0458  8980 116E                CMZ    <SBTFG           CHECK FLAG FOR STOP BIT TIME TEST
000967  045A  0980 06C6                BNE    <SBTE
000968                           *
000969  045C  8980 12E5          JPT4  CMZ    <LC3SYF          SEE IF SYNC. OR ASYNC. LINE
000970  045E  09A2                     BNE    >SYNSP
000971  045F  C900 1224                CMR    $R4,<SECL        SEE IF NUMBER OF TICKS WITHIN
000972  0461  0A84                     BALE   >TRYH            ACCEPTABLE RANGE
000973  0462  C900 1225                CMR    $R4,<SECH
000974  0464  080E                     BAL    >TYXL
000975  0465  F800 115B          TRYH  LDR    $R7,<ERF         CLEAR ERROR FLAG TO
000976  0467  8700 115B                CL     <ERF             SUPPRESS REGISTER DUMP
000977  0469  F380 0DC2                LNJ    $B7,<ERDB
```

```
000978  046B  C380 0CA8                LNJ     $B4,<PRTLSP          ERROR:  INCORRECT LINE SPEED
000979  046D  FF00 115B                STR     $R7,<ERF
000980  046F  4C0A                     LDV     $R4,=10
000981  0470  CF00 1154                STR     $R4,<CSB
000982  0472  8AD2             TYXL    INC     =$R2
000983  0473  C800 12E3                LDR     $R4,<LCT3S           SET LCT TO NEXT SPEED
000984  0475  CA70 0100                ADD     $R4,=X'100'          CHANGE TO NEXT SPEED
000985  0477  2D10                     CMV     $R2,=16              SEE IF ALL SPEED TESTED
000986  0478  0800 03DA                BAL     <NXSPD               BRANCH TO TEST NEXT SPEED
000987                           *
000988  047A  3E02             TSNL    ADV     $R3,=2               TEST NEXT LINE
000989  047B  8F80 119A                RSTR    <SAV3,=Z'0810'       RESTORE REGS.
        047D  0810
000990  047E  0F80 03D4                B       <JPT3
000991                           *
000992  0480  8980 1165         SYNSP   CMZ     <PSFG
000993  0482  0986                     BNE     >SYNSP1
000994  0483  8980 116F                CMZ     <SHORT
000995  0485  0983                     BNE     >SYNSP1
000996  0486  C380 0CA8                LNJ     $B4,<PRTLSP          PRINT SYNCHRONOUS LINE SPEED
000997  0488  4C08             SYNSP1  LDV     $R4,=8
000998  0489  CF00 1154                STR     $R4,<CSB
000999  048B  0FEF                     B       >TSNL
001000                           *
001001                           *
001002                           *         RTC INTERRUPT ROUTINE
001003                           *
001004  048C  0005             NOSPD   RTCF                         TURN OFF RTC
001005  048D  8700 1154                CL      <CSB
001006  048F  DB80 045C                LAB     $B5,<JPT4            SET RUPT POINTER TO RETURN TO LEVEL 4
001007  0491  DF80 12CA                STB     $B5,<ISA4P
001008  0493  8E70 8004                LEV     =Z'8004'
001009  0495  0FF7                     B       >NOSPD
001010                           *
001011                           *
001012                           * + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +
001013                           *         SYNC. RTC INTERRUPT ROUTINE
001014                           *
001015  0496  0005             IRRG    RTCF                         TURN OFF RTC
001016  0497  C380 0D2B                LNJ     $B4,<INRG            INPUT RANGE INTO $R5
001017  0499  8AD1                     INC     =$R1                 FORM TOTAL RESIDUAL RANGE
001018  049A  1904                     BEZ     $R1,>IRRG2           FROM OTHER CCB'S
001019  049B  DA70 03C0         IRRG1   ADD     $R5,=X'3C0'
001020  049D  17FE                     BINC    $R1,>IRRG1
001021  049E  C800 1223         IRRG2   LDR     $R4,<FIVSEC          PUT ELAPSED TIME IN $R4
001022  04A0  DB80 0480                LAB     $B5,<SYNSP           SET RUPT POINTER FOR RETURN TO LEVEL 4
001023  04A2  DF80 12CA                STB     $B5,<ISA4P
001024  04A4  8E70 8004                LEV     =Z'8004'
001025  04A6  0FF0                     B       >IRRG
```

```
001026                                  *
001027                                  * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
001028                                  *
001029                                  *           TEST ADAPTER READY BIT FOR BROADBAND LINES
001030                                  *
001031                                  *
001032  04A7  C870 4320                 ADRTST LDR    $R4,=A'C '
001033  04A9  CF00 13A9                        STR    $R4,<ERMG+1
001034  04AB  B380 0BA6                        LNJ    $B3,<CLFG              CLEAR ALL TEST FLAGS
001035  04AD  8700 1313                        CL     <LCT8FG
001036  04AF  8A80 114D                        INC    <ARFG                  SET ADAPTER READY TEST FLAG
001037  04B1  7C2D                             LDV    $R7,=45
001038                                  *
001039                                  *           TEST WIDEBAND 301/303 (C5 AS SYNC. CHARACTER)
001040                                  *
001041  04B2  9B80 11C0                        LAB    $B1,<WDB3              LOAD ADDRESS OF LINE TYPE
001042  04B4  B380 0A8C                 T46Y   LNJ    $B3,<DTALOP            LOOP SUB-ROUTINE
001043  04B6  0F85                             B      >T46A
001044  04B7  1333                             DC     <WCP6
001045  04B8  130B                             DC     <LCT8
001046  04B9  00C0                             DC     192
001047  04BA  11E3                             DC     <DOUT
001048                                  *
001049                                  *           TEST BI-SYNC. WIDEBAND 301/303 (C5 AS SYNC. CHAR.)
001050                                  *
001051  04BB  C871                      T46A   LDR    $R4,+$B1               LOAD ID TYPE INTO $R4
001052  04BC  C900 11C1                        CMR    $R4,<WDB3B             CHECK TO SEE IF ALREADY DONE
001053  04BE  0902                             BE     >T46
001054  04BF  0FF5                             B      >T46Y
001055                                  *
001056                                  *           TEST WIDEBAND V35 (C5 AS SYNC. CHARACTER)
001057                                  *
001058  04C0  9B80 11C2                 T46    LAB    $B1,<WDBV35            LOAD ADDRESS OF LINE TYPE
001059  04C2  B380 0A8C                 T46Z   LNJ    $B3,<DTALOP
001060  04C4  0F85                             B      >T47A
001061  04C5  1333                             DC     <WCP6
001062  04C6  130B                             DC     <LCT8
001063  04C7  00C0                             DC     192
001064  04C8  11E3                             DC     <DOUT
001065                                  *
001066                                  *           TEST BI-SYNC. WIDEBAND V35 (C5 AS SYNC. CHAR.)
001067                                  *
001068  04C9  C871                      T47A   LDR    $R4,+$B1               LOAD ID TYPE INTO $R4
001069  04CA  C900 11C3                        CMR    $R4,<WDBV5B            CHECK TO SEE IF ALREADY DONE
001070  04CC  0902                             BE     >T47
001071  04CD  0FF5                             B      >T46Z
001072                                  *
001073  04CE  8382                      T47    JMP    $B2                    JUMP OUT OF TEST
001074                                  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
001075                              *
001076                              *          ADAPTER READY CHECK ROUTINE
001077                              *
001078  04CF  E380 0BB7             ARCK   LNJ    $B6,<BASE          LOAD $R5 WITH LCT BASE ADDRESS
001079  04D1  5E3C                         ADV    $R5,=60
001080  04D2  DF00 04D6                     STR    $R5,<ARCK1
001081                              *
001082  04D4  E380 0D75                     LNJ    $B6,<RDLCT         READ LCT LOCATIONS 25-28
001083  04D6  0000                  ARCK1  DC     0
001084  04D7  0004                         DC     4
001085                              *
001086  04D8  8980 1173                     CMZ    <LCTV              SEE IF NO ERRORS IN ADAPTER READY BIT
001087  04DA  0903                          BE     >ARCK2
001088  04DB  F380 0DC2                     LNJ    $B7,<ERDB          ERROR: ADAPTER READY BIT SETTING INCORRECTLY
001089  04DD  8980 1174             ARCK2  CMZ    <LCTV+1            CHECK BUFFER EMPTY BIT ERROR
001090  04DF  0903                          BE     >ARCK3
001091  04E0  F380 0DC2                     LNJ    $B7,<ERDB          ERROR:  INCORRECT BUFFER EMPTY BIT
001092  04E2  0F80 0BA0             ARCK3  B      <JP5A
001093                              *
001094                              *
001095                              * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
001096                              *
001097                              *        CRC AND PARITY TEST
001098                              *
001099  04E4  C870 4420             CRPTB  LDR    $R4,=A'D '
001100  04E6  CF00 13A9                     STR    $R4,<ERMG+1
001101  04E8  B380 0BA6                     LNJ    $B3,<CLFG          CLEAR ALL TEST FLAGS
001102  04EA  8A80 1156                     INC    <CRCFG             SET FLAG FOR DTALOP SUB-ROUTINE
001103  04EC  8700 1160                     CL     <MASK              SET MASK FOR ZV$C
001104  04EE  C870 C802                     LDR    $R4,=Z'C802'       SET LCT CONFIG. FOR
001105  04F0  CF00 12D9                     STR    $R4,<LC2CFR        RECV., ODD PARITY, CRC16
001106  04F2  4E20                          ADV    $R4,=X'20'
001107  04F3  CF00 12DA                     STR    $R4,<LC2CFX        XMIT LCT CONFIG. ODD PARITY
001108  04F5  EB80 11DC                     LAB    $B6,<CPFLG         LOAD ADDRESS OF LCT FLAG TABLE
001109  04F7  C876                          LDR    $R4,+$B6           LOAD $R4 WITH LCT FLAG
001110  04F8  CF00 12DD             CRPTB1 STR    $R4,<LCT2FG        STORE INTO LCT TABLE
001111  04FA  F870 0F18                     LDR    $R7,=X'F18'        STORE INITIAL SPEED FOR ASYC. LINES
001112  04FC  FF00 12DB                     STR    $R7,<LCT2S
001113  04FE  7C06                          LDV    $R7,=6             NUMBER OF 30 MS TIME DELAY FOR TRANSFER
001114                              *
001115                              *          TEST ASYNCHRONOUS LINES WITH OLD BAUD RATES
001116                              *             9600 BAUD, WITH 2 STOP BITS
001117                              *
001118  04FF  9B80 11BA                     LAB    $B1,<ASYN          STORE ADDRESS OF LINE TYPE
001119  0501  B380 0A8C                     LNJ    $B3,<DTALOP        GO TO LOOP SUB-ROUTINE
001120  0503  0F85                          B      >T8
001121  0504  1322                          DC     <WCP2              ADDRESS TABLE FOR BLOCK WRITE
001122  0505  12D7                          DC     <LCT2              LCT TABLE
001123  0506  003F                          DC     (DFLT-DOUT)        RANGE IN WORDS
```

```
0507  11E3                DC     <DOUT
                    *
                    *
                    *             TEST ASYCHRONOUS LINES WITH NEW BAUD RATES
                    *                19200 BAUD, WITH 2 STOP BITS
0508  9B80 11BB     T8    LAB    $B1,<ASYN1           LOAD  LINE TYPE
050A  B380 0A8C     T8A   LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
050C  0F85                B      >T9
050D  1322                DC     <WCP2                ADDRESS TABLE FOR BLOCK WRITE
050E  12D7                DC     <LCT2                LCT TABLE
050F  003F                DC     (DFLT-DOUT)          RANGE IN WORDS
0510  11E3                DC     <DOUT
                    *
                    *             TEST CURRENT LOOP ASYNCHRONOUS
                    *
0511  C871          T9    LDR    $R4,+$B1             LOAD ID TYPE INTO $R4
0512  C900 11BC           CMR    $R4,<ASYNC           CHECK TO SEE IF ALREADY DONE
0514  0902                BE     >T9A
0515  0FF5                B      >T8A
      0516          T9A   EQU    $
0516  F870 3218           LDR    $R7,=X'3218'         STORE SYNC. CHAR. INTO LCT TABLE
0518  FF00 12DB           STR    $R7,<LCT2S
051A  7C28                LDV    $R7,=40              NUMBER OF 30 MS TIME DELAYS
                    *
                    *             TEST SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
                    *
051B  9B80 11BD           LAB    $B1,<SYNC            LOAD LINE TYPE
051D  B380 0A8C     QA2   LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
051F  0F85                B      >T10
0520  1322                DC     <WCP2                ADDRESS TABLE FOR BLOCK WRITE
0521  12D7                DC     <LCT2                LCT TABLE
0522  002F                DC     (DFLT-DOUT1)         RANGE IN WORDS
0523  11F3                DC     <DOUT1
                    *
                    *             TEST BI-SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
                    *
0524  C871          T10   LDR    $R4,+$B1             LOAD ID TYPE INTO $R4
0525  C900 11BE           CMR    $R4,<BISYNC          CHECK TO SEE IF ALREADY DONE
0527  0902                BE     >T12
0528  0FF5                B      >QA2
                    *
                    *             TEST MIL 188 SYNCHRONOUS (32 AS SYNC. CHAR.)
                    *
0529  9B80 11BF     T12   LAB    $B1,<SYN188          LOAD LINE TYPE
052B  B380 0A8C           LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
052D  0F85                B      >T13
052E  1322                DC     <WCP2                ADDRESS TABLE FOR BLOCK WRITE
052F  12D7                DC     <LCT2                LCT TABLE
0530  002F                DC     (DFLT-DOUT1)         RANGE IN WORDS
```

```
001173  0531  11F3                      DC      <DOUT1
001174                            *
001175                            *
001176                            *           TEST WIDEBAND 301/303 (3A AS SYNC. CHARACTER)
001177                            *
001178  0532  9B80 1322            T13     LAB     $B1,<WCP2           LOAD ADDRESS OF MEDIUM SPEED ADDRESS TABLE
001179  0534  8980 115E                    CMZ     <HGSPD              CHECK FOR HIGH SPEED LINE
001180  0536  0903                        BE      >T13X
001181  0537  9B80 133F                    LAB     $B1,<WCP8           LOAD ADDRESS OF HIGH SPEED ADDRESS TABLE
001182  0539  9F80 0550            T13X    STB     $B1,<T13Z           STORE INTO DTALOP SUB-ROUTINE
001183  053B  9F80 0542                    STB     $B1,<T13Y
001184  053D  9B80 11C0                    LAB     $B1,<WDB3           LOAD ADDRESS OF LINE TYPE
001185  053F  B380 0A8C            T13ZZ   LNJ     $B3,<DTALOP         LOOP SUB-ROUTINE
001186  0541  0F85                        B       >T37A
001187  0542  0000                T13Y    RESV    $AF,0
001188  0543  12D7                        DC      <LCT2
001189  0544  003F                        DC      (DFLT-DOUT)
001190  0545  11E3                        DC      <DOUT
001191                            *
001192                            *           TEST BI-SYNC. WIDEBAND 301/303 (3A AS SYNC. CHAR.)
001193                            *
001194  0546  C871                T37A    LDR     $R4,+$B1            LOAD ID TYPE INTO $R4
001195  0547  C900 11C1                    CMR     $R4,<WDB3B          CHECK TO SEE IF ALREADY DONE
001196  0549  0902                        BE      >T37
001197  054A  0FF5                        B       >T13ZZ
001198                            *
001199                            *           TEST WIDEBAND V35 (3A AS SYNC. CHAR.)
001200                            *
001201  054B  9B80 11C2            T37     LAB     $B1,<WDBV35         'LOAD ADDRESS OF LINE TYPE
001202  054D  B380 0A8C            T13YY   LNJ     $B3,<DTALOP
001203  054F  0F85                        B       >T38A
001204  0550  0000                T13Z    RESV    $AF,0
001205  0551  12D7                        DC      <LCT2
001206  0552  003F                        DC      (DFLT-DOUT)
001207  0553  11E3                        DC      <DOUT
001208                            *
001209                            *           TEST BI-SYNC. WIDEBAND V35 (3A AS SYNC. CHAR.)
001210                            *
001211  0554  C871                T38A    LDR     $R4,+$B1            LOAD ID TYPE INTO $R4
001212  0555  C900 11C3                    CMR     $R4,<WDBV5B         CHECK TO SEE IF ALREADY DONE
001213  0557  0902                        BE      >T38
001214  0558  0FF5                        B       >T13YY
001215                            *
001216  0559  C876                T38     LDR     $R4,+$B6            CHANGE LCT FLAG
001217  055A  4980 04F8                    BNEZ    $R4,<CRPTB1         SEE IF ALL FLAGS CHECKED
001218  055C  C800 12D9                    LDR     $R4,<LC2CFR         SEE IF ALL COMBINATIONS HAVE BEEN RUN
001219  055E  C970 D802                    CMR     $R4,=Z'D802'
001220  0560  090D                        BE      >CRPTB2             BRANCH IF ALL TESTED
001221  0561  C870 D802                    LDR     $R4,=Z'D802'        CHANGE LCT CONFIG. TO EVEN PARITY
```

```
001222  0563  CF00 12D9              STR   $R4,<LC2CFR          STORE RECV CONFIG.
001223  0565  4E20                   ADV   $R4,=X'20'           CHANGE ADDRESS FOR XMIT  LCT
001224  0566  CF00 12DA              STR   $R4,<LC2CFX          STORE XMIT CONFIG.
001225  0568  EB80 11DD              LAB   $B6,<CPFLG+1         SET TO TEST WITH PARITY
001226  056A  C876                   LDR   $R4,+$B6             LOAD $R4 WITH LCT FLAG
001227  056B  0F80 04F8              B     <CRPTB1              BRANCH TO TEST ALL LINES AGAIN WITH EVEN PARI
001228  056D  8700 1156     CRPTB2  CL    <CRCFG               CLEAR CRC AND PARITY TEST FLAG
001229  056F  8382                   JMP   $B2                  JUMP OUT OF TEST
001230                        *
001231                        * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
001232                        *
001233                        *          CRC AND PARITY CHECK ROUTINE
001234                        *
001235                        *
001236  0570  8F00 118A        CRPRT   SAVE  <SAV2,=Z'91E0'       SAVE REGS. $B1,B2,$R7
       0572  91E0
001237  0573  AC80 0ADF              LDB   $B2,<TG2             LOAD ADDRESS OF LCT TABLE INTO $B2
001238  0575  C842 0002              LDR   $R4,$B2.2            LOAD $R4 WITH CONFIG.
001239  0577  404E                   SOR   $R4,14               SHIFT TO CHAR. SIZE
001240  0578  4E05                   ADV   $R4,=5
001241  0579  CF00 1153              STR   $R4,<CS              STORE CHARACTER SIZE FROM CONFIG.
001242                        *
001243                        *          SET ZV$C MASK ACCORDING TO CHAR. SIZE
001244                        *
001245  057B  4D08                   CMV   $R4,=8
001246  057C  090A                   BE    >CRPR2
001247  057D  4D07                   CMV   $R4,=7
001248  057E  0984                   BNE   >CHRF1
001249  057F  C870 7F7F              LDR   $R4,=Z'7F7F'         SET MASK FOR 7 BIT CHAR.
001250  0581  0F83                   B     >CRPR1
001251  0582  C870 3F3F     CHRF1   LDR   $R4,=Z'3F3F'         SET MASK FOR 5 BIT CHAR.
001252  0584  CF00 1160     CRPR1   STR   $R4,<MASK
001253                        *
001254  0586  C842 0006     CRPR2   LDR   $R4,$B2.6            CHECK LCT FLAG TO SEE IF CRC AND/OR PARITY
001255  0588  4D17                   CMV   $R4,=X'17'
001256  0589  0A00 05BC              BAG   <CKCRC               BRANCH TO CHECK CRC ONLY
001257                        *
001258                        *
001259  058B  BC80 0B02              LDB   $B3,<TG3             LOAD ADDRESS OF DATA BUFFER
001260  058D  9B80 1D00              LAB   $B1,<DIN1            LOAD ADDRESS OF SHOULD BE WITH PARITY
001261  058F  F800 1166              LDR   $R7,<RANGE           LOAD RANGE OF TRANSFER IN WORDS
001262  0591  7001                   SOL   $R7,1                CHANGE RANGE TO BYTES
001263  0592  C842 0002              LDR   $R4,$B2.2            CHECK CONFIGURATION TO SEE IF ODD
001264  0594  C570 1000              AND   $R4,=X'1000'         OR EVEN PARITY
001265  0596  404C                   SOR   $R4,12               SHIFT TO LSB
001266  0597  8B54                   LBC   =$R4,=X'1'           COMPLEMENT THE  BIT TO WORK IN SUB-ROUTINE
       0598  0001
001267  0599  CF00 1164              STR   $R4,<PTTY            STORE IN PARITY TYPE LOCATION
001268                        *
```

```
001269  059B  C380 0BD9                LNJ   $B4,<CPAR            GO TO CALCULATE PARITY
001270                          *
001271                                CALL  ZV$C,DIN1,DIN,MASK,RANGE,ERAR
        059D  FBC0 0003
        059F  D380 0000       X
        05A1  0F80
        05A2  1D00
        05A3  1C00
        05A4  1160
        05A5  1166
        05A6  1176
001272  05A7  8980 1176                CMZ   <ERAR                SEE IF ANY ERRORS
001273  05A9  090D                      BE    >CKPTY2
001274  05AA  D800 1178                LDR   $R5,<ERAR+2          LOAD 'is' INTO $R5
001275  05AC  E800 1177                LDR   $R6,<ERAR+1          LOAD 'should be' INTO $R6
001276  05AE  F800 1176                LDR   $R7,<ERAR            LOAD $R7 WITH WORD COUNT
001277  05B0  F380 0DBA                LNJ   $B7,<ERMB            ERROR:  INCORRECT PARITY
001278                                CALL  ZV$CO                CONTINUE CHECKING ENTIRE BUFFER
        05B2  FBF0 0001
        05B4  D380 0000       X
001279  05B6  DB80 0B74          CKPTY2 LAB   $B5,<JP5             CHANGE $B5  TO POINT TO AFTER COMPARE
001280  05B8  89C2 0006                CMZ   $B2.6                CHECK FLAG IF PARITY ON TO
001281  05BA  0A80 05F4                BALE  <EDCK                BRANCH IF PARITY NOT ON
001282  05BC  E380 0BB7          CKCRC LNJ   $B6,<BASE            LOAD $R5 WITH LCT BASE ADDRESS
001283  05BE  5E03                      ADV   $R5,=3               FOR  BLOCK READ OF CRC RESIDUE
001284  05BF  DF00 05C6                STR   $R5,<RCRRL           STORE RECV. RESIDUE ADDRESS
001285  05C1  5E20                      ADV   $R5,=32
001286  05C2  DF00 05CF                STR   $R5,<RCRXL           STORE XMIT RESIDUE ADDRESS
001287                          *
001288  05C4  E380 0D75                LNJ   $B6,<RDLCT           READ RECV. CRC RESIDUE
001289  05C6  0000              RCRRL DC    0                    RAM ADDRESS
001290  05C7  0002                    DC    2                    RANGE
001291                          *
001292  05C8  C800 1173                LDR   $R4,<LCTV
001293  05CA  4018                      SCL   $R4,8                ROTATE THE ORDER
001294  05CB  CF00 1167                STR   $R4,<RCRC
001295                          *
001296  05CD  E380 0D75                LNJ   $B6,<RDLCT           READ XMIT CRC RESIDUE
001297  05CF  0000              RCRXL DC    0                    RAM ADDRESS
001298  05D0  0002                    DC    2                    RANGE
001299                          *
001300  05D1  C800 1173                LDR   $R4,<LCTV
001301  05D3  4018                      SCL   $R4,8
001302  05D4  CF00 1172                STR   $R4,<XCRC
001303                          *
001304  05D6  EB80 11CA                LAB   $B6,<CRC16           LOAD CRC TYPE
001305  05D8  E800 1153                LDR   $R6,<CS              LOAD CHAR. SIZE
001306  05DA  BC80 0B02                LDB   $B3,<TG3             LOAD ADDRESS OF OUTPUT BUFFER
001307  05DC  C842 0006                LDR   $R4,$B2.6            CHECK FLAG TO SEE IF OUTPUT BUFFER CORRECT
```

```
05DE   4D17                CMV    $R4,=X'17'
05DF   0983                BNE    >CKPCR
05E0   BB80 1D00           LAB    $B3,<DIN1
05E2   F800 1166    CKPCR  LDR    $R7,<RANGE          LOAD RANGE OF TRANSFER IN WORDS
                    *
05E4   C380 0BBB           LNJ    $B4,<CRC            CALCULATE CRC FOR THE TRANSFERS
                    *
05E6   E800 1155           LDR    $R6,<CRCAC          LOAD $R6 WITH SHOULD BE
05E8   D800 1167           LDR    $R5,<RCRC           LOAD CRC RECV. RESIDUE
05EA   E955                CMR    $R6,=$R5            COMPARE RECEIVE CRC
05EB   0903                BE     >RCRCG
05EC   F380 0DC2           LNJ    $B7,<ERDB           ERROR: RECEIVE CRC RESIDUE WRONG
05EE   D800 1172    RCRCG  LDR    $R5,<XCRC           LOAD XMIT CRC RESIDUE
05F0   E955                CMR    $R6,=$R5            COMPARE XMIT CRC
05F1   0903                BE     >EDCK
05F2   F380 0DC2           LNJ    $B7,<ERDB           ERROR:  XMIT RESIDUE WRONG
05F4   8F80 118A    EDCK   RSTR   <SAV2,=Z'91E0'      RESTORE REGS.
05F6   91E0
05F7   8385                JMP    $B5                 GO BACK TO DATLOP SUB-ROUTINE
                    *
                    *
                    * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                    *
                    *             CHARACTER SIZE TEST
                    *
05F8   C870 4520    CHSZTD LDR    $R4,=A'E '
05FA   CF00 13A9           STR    $R4,<ERMG+1
05FC   B380 0BA6           LNJ    $B3,<CLFG           CLEAR ALL TEST FLAGS
05FE   8A80 1150           INC    <CHSZFG             SET CHARACTER SIZE TEST FLAG
0600   4C02                LDV    $R4,=2              LOAD CONFIGURATION FOR 1 STOP BIT ODD
0601   CF00 12E9           STR    $R4,<LC4CFR         PARITY, 5 BIT CHAR. RECV.
0603   4E20                ADV    $R4,=X'20'
0604   CF00 12EA           STR    $R4,<LC4CFX         STORE XMIT CONF.
0606   C870 1F1F           LDR    $R4,=Z'1F1F'        SET MASK FOR FIVE BIT CHAR.
0608   CF00 1160           STR    $R4,<MASK
060A   7C06         CHSZT1 LDV    $R7,=6              NUMBER OF 30 MS TIME DELAY FOR TRANSFER
060B   C870 0F18           LDR    $R4,=X'F18'         SET LINE SPEED FOR ASYNC. LINE
060D   CF00 12EB           STR    $R4,<LCT4S
                    *
                    *             TEST ASYNCHRONOUS LINES WITH OLD BAUD RATES
                    *               9600 BAUD, WITH 1 STOP BIT
                    *
060F   9B80 11BA           LAB    $B1,<ASYN           STORE ADDRESS OF LINE TYPE
0611   B380 0A8C           LNJ    $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
0613   0F85                B      >T14
0614   1327                DC     <WCP3               ADDRESS TABLE FOR BLOCK WRITE
0615   12E7                DC     <LCT4               LCT TABLE
0616   003F                DC     (DFLT-DOUT)         RANGE IN WORDS
0617   11E3                DC     <DOUT
```

```
001356                                  *
001357                                  *           TEST ASYCHRONOUS LINES WITH NEW BAUD RATES
001358                                  *                19200 BAUD, WITH 1 STOP BIT
001359                                  *
001360  0618  9B80 11BB                T14     LAB     $B1,<ASYN1              LOAD  LINE TYPE
001361  061A  B380 0A8C                T14A    LNJ     $B3,<DTALOP             GO TO LOOP SUB-ROUTINE
001362  061C  0F85                             B       >T15
001363  061D  1327                             DC      <WCP3                   ADDRESS TABLE FOR BLOCK WRITE
001364  061E  12E7                             DC      <LCT4                   LCT TABLE
001365  061F  003F                             DC      (DFLT-DOUT)             RANGE IN WORDS
001366  0620  11E3                             DC      <DOUT
001367                                  *
001368                                  *           TEST CURRENT LOOP ASYNCHRONOUS
001369                                  *
001370  0621  C871                      T15     LDR     $R4,+$B1                LOAD ID TYPE INTO $R4
001371  0622  C900 11BC                         CMR     $R4,<ASYNC              CHECK TO SEE IF ALREADY DONE
001372  0624  0902                             BE      >T15A
001373  0625  0FF5                             B       >T14A
001374        0626                      T15A    EQU     $
001375  0626  F870 E618                         LDR     $R7,=Z'E618'            STORE SYNC. CHAR. INTO LCT TABLE
001376  0628  FF00 12EB                         STR     $R7,<LCT4S
001377  062A  7C28                             LDV     $R7,=40                 NUMBER OF 30 MS TIME DELAYS
001378                                  *
001379                                  *           TEST SYNCHRONOUS LINES (E6 AS SYNC. CHAR.)
001380                                  *
001381  062B  9B80 11BD                         LAB     $B1,<SYNC               LOAD LINE TYPE
001382  062D  B380 0A8C                QA3     LNJ     $B3,<DTALOP             GO TO LOOP SUB-ROUTINE
001383  062F  0F85                             B       >T16
001384  0630  1327                             DC      <WCP3                   ADDRESS TABLE FOR BLOCK WRITE
001385  0631  12E7                             DC      <LCT4                   LCT TABLE
001386  0632  002F                             DC      (DFLT-DOUT1)            RANGE IN WORDS
001387  0633  11F3                             DC      <DOUT1
001388                                  *
001389                                  *           TEST BI-SYNCHRONOUS LINES (E6 AS SYNC. CHAR.)
001390                                  *
001391  0634  C871                      T16     LDR     $R4,+$B1                LOAD ID TYPE INTO $R4
001392  0635  C900 11BE                         CMR     $R4,<BISYNC             CHECK TO SEE IF ALREADY DONE
001393  0637  0902                             BE      >T17
001394  0638  0FF5                             B       >QA3
001395                                  *
001396                                  *           TEST MIL 188 SYNCHRONOUS (E6 AS SYNC. CHAR.)
001397                                  *
001398  0639  9B80 11BF                T17     LAB     $B1,<SYN188             LOAD LINE TYPE
001399  063B  B380 0A8C                         LNJ     $B3,<DTALOP             GO TO LOOP SUB-ROUTINE
001400  063D  0F85                             B       >T18
001401  063E  1327                             DC      <WCP3                   ADDRESS TABLE FOR BLOCK WRITE
001402  063F  12E7                             DC      <LCT4                   LCT TABLE
001403  0640  002F                             DC      (DFLT-DOUT1)            RANGE IN WORDS
001404  0641  11F3                             DC      <DOUT1
```

```
001405                                  *
001406                                  *
001407                                  *           TEST WIDEBAND 301/303 (E6 AS SYNC. CHARACTER)
001408                                  *
001409  0642  9B80 1327                 T18     LAB     $B1,<WCP3           LOAD ADDRESS OF MEDIUM SPEED ADDRESS TABLE
001410  0644  8980 115E                         CMZ     <HGSPD              CHECK FOR HIGH SPEED LINE
001411  0646  0903                             BE      >T18X
001412  0647  9B80 133F                         LAB     $B1,<WCP8           LOAD ADDRESS OF HIGH SPEED ADDRESS TABLE
001413  0649  9F80 0660                 T18X    STB     $B1,<T18Z           STORE INTO DTALOP SUB-ROUTINE
001414  064B  9F80 0652                         STB     $B1,<T18Y
001415  064D  9B80 11C0                         LAB     $B1,<WDB3           LOAD ADDRESS OF LINE TYPE
001416  064F  B380 0A8C                 T18YY   LNJ     $B3,<DTALOP         LOOP SUB-ROUTINE
001417  0651  0F85                             B       >T35A
001418  0652  0000                     T18Y    RESV    $AF,0
001419  0653  12E7                             DC      <LCT4
001420  0654  003F                             DC      (DFLT-DOUT)
001421  0655  11E3                             DC      <DOUT
001422                                  *
001423                                  *           TEST BI-SYNC. WIDEBAND 301/303 (E6 AS SYNC. CHAR.)
001424                                  *
001425  0656  C871                     T35A    LDR     $R4,+$B1            LOAD ID TYPE INTO $R4
001426  0657  C900 11C1                         CMR     $R4,<WDB3B          CHECK TO SEE IF ALREADY DONE
001427  0659  0902                             BE      >T35
001428  065A  0FF5                             B       >T18YY
001429                                  *
001430                                  *           TEST WIDEBAND V35 (E6 AS SYNC. CHAR.)
001431                                  *
001432  065B  9B80 11C2                 T35     LAB     $B1,<WDBV35         LOAD ADDRESS OF LINE TYPE
001433  065D  B380 0A8C                 T18ZZ   LNJ     $B3,<DTALOP
001434  065F  0F85                             B       >T36A
001435  0660  0000                     T18Z    RESV    $AF,0
001436  0661  12E7                             DC      <LCT4
001437  0662  003F                             DC      (DFLT-DOUT)
001438  0663  11E3                             DC      <DOUT
001439                                  *
001440                                  *           TEST BI-SYNC. WIDEBAND V35 (E6 AS SYNC. CHAR.)
001441                                  *
001442  0664  C871                     T36A    LDR     $R4,+$B1            LOAD ID TYPE INTO $R4
001443  0665  C900 11C3                         CMR     $R4,<WDBV5B         CHECK TO SEE IF ALREADY DONE
001444  0667  0902                             BE      >T36
001445  0668  0FF5                             B       >T18ZZ
001446                                  *
001447  0669  8A80 1156                 T36     INC     <CRCFG
001448  066B  C800 12E9                         LDR     $R4,<LC4CFR         CHANGE CONFIGURATION TO NEXT CHAR. SIZE
001449  066D  CA70 4000                         ADD     $R4,=X'4000'
001450  066F  0608                             BCT     >CHSZDE             IF CARRY ALL CHAR. SIZES HAVE BEEN TESTED
001451  0670  CF00 12E9                         STR     $R4,<LC4CFR         STORE RECV CONFIG.
001452  0672  4E20                             ADV     $R4,=X'20'
001453  0673  CF00 12EA                         STR     $R4,<LC4CFX         STORE XMIT CONFIG.
```

```
0675   0F80 060A              B      <CHSZT1          DO NEXT CHAR.
                      *
0677   8700 1156      CHSZDE  CL     <CRCFG           CLEAR CRC AND PARITY FLAG
0679   8700 1150              CL     <CHSZFG          CLEAR CHARACTER SIZE TEST FLAG
067B   8382                   JMP    $B2              JUMP OUT OF TEST
                      *
                      * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                      *
                      *              STOP BIT TEST
                      *
067C   C870 4620      STBITE  LDR    $R4,=A'F '
067E   CF00 13A9              STR    $R4,<ERMG+1
0680   B380 08A6              LNJ    $B3,<CLFG        CLEAR ALL TEST FLAGS
0682   CB80 0695              LAB    $B4,<SBIT1       SET UP INTERRUPT VECTORS AND
0684   CF80 12CA              STB    $B4,<ISA4P       POINTERS TO CHANGE RUNNING LEVEL TO 4
0686   CB80 12C7              LAB    $B4,<ISA4+$AF
0688   CF80 0004      X       STB    $B4,<ZHISAZ+4
                      *
068A   C800 115F              LDR    $R4,<HRTZ        CALCULATE THE LOW AND THE
068C   4F06                   MLV    $R4,=6           HIGH  RANGE OF ACCEPTABLE CLOCK TICKS
068D   4EF7                   ADV    $R4,=-9
068E   CF00 1224              STR    $R4,<SECL
0690   4E12                   ADV    $R4,=18
0691   CF00 1225              STR    $R4,<SECH
0693   8E70 8004              LEV    =Z'8004'         CHANGE RUNNING LEVEL TO 4
0695   4C01           SBIT1   LDV    $R4,=1           SET RTC LEVEL TO 1
0696   CF00 0000      X       STR    $R4,<ZHRTCL
                      *
0698   CB80 06FB              LAB    $B4,<NBITT       SET RUPT POINTER FOR RTC INTERRUPT
069A   CF80 12C4              STB    $B4,<ISA1P
                      *
069C   CB80 12C1              LAB    $B4,<ISA1+$AF    SET RUPT VECTOR FOR LEVEL 1 RUPT
069E   CF80 0001      X       STB    $B4,<ZHISAZ+1
                      *
06A0   8700 12E5              CL     <LC3SYF          CLEAR SYNC. FLAG FOR ASYNC. LINE
06A2   8A80 116E              INC    <SBTFG           SET STOP BIT TIME TEST FLAG
06A4   C870 C002              LDR    $R4,=Z'C002'     LOAD AND STORE INITIAL LINE CONF.
06A6   CF00 12E1              STR    $R4,<LC3CFR      FOR RECV.
06A8   4E20                   ADV    $R4,=X'20'
06A9   CF00 12E2              STR    $R4,<LC3CFX      FOR XMIT
06AB   C870 0318              LDR    $R4,=X'318'      LOAD $R4 WITH SPEED FOR ASYNC. LINE
06AD   9B80 11BA              LAB    $B1,<ASYN        LOAD LINE TYPE
06AF   EB80 1276              LAB    $B6,<SBITBL      LOAD ADDRESS OF CHAR. NUMBER TABLE
06B1   B380 03CE              LNJ    $B3,<SPDTLP      GO TO SPEED TIME LOOP
                      *
06B3   C870 0218              LDR    $R4,=X'218'      LOAD $R4 WITH SPEED SELECT FOR NEW  ASYC.
06B5   9B80 11BB              LAB    $B1,<ASYN1       LOAD ADDRESS OF LINE TYPE
06B7   B380 03CE              LNJ    $B3,<SPDTLP      GO TO SPEED TIME LOOP
                      *
```

```
06B9   C870 0218              LDR    $R4,=X'218'
06BB   9B80 11BC              LAB    $B1,<ASYNC
06BD   B380 03CE              LNJ    $B3,<SPDTLP
                       *
                              CALL   ZV$RD               RE-INITIALIZE INTERRUPT POINTERS AND VECTORS
06BF   FBF0 0001
06C1   D380 0000     X
06C3   8700 116E              CL     <SBTFG              CLEAR STOP BIT TEST FLAG
06C5   8382                   JMP    $B2                 JUMP OUT OF TEST
                       *
                       * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                       *
                       *            STOP BIT TIME CHECKED HERE
                       *
06C6   C940 0B5E       SBTE   CMR    $R4,SECH            SEE IF CLOCK TICK WITHIN
06C8   0803                   BAL    >SBTE1              ACCEPTED RANGE
06C9   F380 0DC2              LNJ    $B7,<ERDB           ERROR:  STOP BITS ARE TOO LONG
06CB   C900 1224       SBTE1  CMR    $R4,<SECL
06CD   0A03                   BAG    >SBTE2
06CE   F380 0DC2              LNJ    $B7,<ERDB           ERROR:  STOP BITS ARE TOO SHORT
06D0   D800 12E1       SBTE2  LDR    $R5,<LC3CFR         LOAD LCT CONFIG.
06D2   8955                   LBT    =$R5,=Z'0800'       SEE IF BIT SET AND SET IT
06D3   0800
06D4   0509                   BBT    >SBTE3              IF TRUE TRY SLOW SPEED FOR 1.5 STOP BIT
06D5   DF00 12E1              STR    $R5,<LC3CFR         STORE LCT CONFIG. WITH 2 STOP BITS
06D7   5E20                   ADV    $R5,=X'20'
06D8   DF00 12E2              STR    $R5,<LC3CFX         STORE CONFIG. XMIT
06DA   8AD2                   INC    =$R2                INCREMENT INDEX
06DB   0F80 03DC              B      <NXSPD1
                       *
06DD   2D03            SBTE3  CMV    $R2,=3              SEE IF ALL STOP BIT S CHECKED
06DE   0900 06F2              BE     <SBTE4
06E0   4C02                   LDV    $R4,=2              CHANGE CONFIG. FOR 5 BIT CHAR.
06E1   CF00 12E1              STR    $R4,<LC3CFR         STORE RECV  CONFIG.
06E3   4E20                   ADV    $R4,=X'20'
06E4   CF00 12E2              STR    $R4,<LC3CFX         STORE XMIT CONFIG.
06E6   8AD2                   INC    =$R2                INCREMENT INDEX
06E7   C870 0218              LDR    $R4,=X'218'         LDR $R4 TO SET LINE SPEED TO 75 BAUD
06E9   D801                   LDR    $R5,$B1             IF NEW ASYC. RESET $R4 FOR 75 BAUD
06EA   D900 11BA              CMR    $R5,<ASYN
06EC   0900 03DA              BE     <NXSPD
06EE   C870 0118              LDR    $R4,=X'118'
06F0   0F80 03DA              B      <NXSPD
                       *
06F2   C870 C002       SBTE4  LDR    $R4,=Z'C002'        LOAD AND STORE BEGINNING CONF.
06F4   CF00 12E1              STR    $R4,<LC3CFR         FOR RECV.
06F6   4E20                   ADV    $R4,=X'20'
06F7   CF00 12E2              STR    $R4,<LC3CFX         FOR XMIT
06F9   0F80 047A              B      <TSNL
```

```
                         *
                         *
                         * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                         *
                         *              CLOOCK INTERRUPT ROUTINE
                         *
06FB  0005               NBITT  RTCF                          TURN OFF CLOCK
06FC  F380 0DC2                 LNJ    $B7,<ERDB              ERROR:  STATUS NOT COMPLETE AFTER 5 SEC.
06FE  DB80 03DC                 LAB    $B5,<NXSPD1            RETRY THE LINE
0700  DF80 12CA                 STB    $B5,<ISA4P             WITH SAME CONFIG.
0702  8E70 8004                 LEV    =Z'8004'
0704  0FF7                      B      >NBITT
                         *
                         *
                         *
                         * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                         *
                         *          PARITY ERROR TEST
                         *
0705  C870 4720          PARERF LDR    $R4,=A'G '
0707  CF00 13A9                 STR    $R4,<ERMG+1
0709  B380 0BA6                 LNJ    $B3,<CLFG              CLEAR TEST FLAGS
070B  8A80 1163                 INC    <PARFG
070D  C870 C802                 LDR    $R4,=Z'C802'           LOAD CONFIG. ON REC. FOR ODD PARITY
070F  CF00 12D9                 STR    $R4,<LC2CFR
0711  CA70 1020                 ADD    $R4,=X'1020'           SET CONFIG. ON XMIT TO EVEN PARITY
0713  CF00 12DA                 STR    $R4,<LC2CFX
0715  C870 F017                 LDR    $R4,=Z'F017'           SET CCP FLAG FOR PARITY ONLY
0717  CF00 12DD                 STR    $R4,<LCT2FG
0719  F870 0F18                 LDR    $R7,=X'F18'            STORE INITIAL SPEED FOR ASYC. LINES
071B  FF00 12DB                 STR    $R7,<LCT2S
071D  7C06                      LDV    $R7,=6                 NUMBER OF 30 MS TIME DELAY FOR TRANSFER .
                         *
                         *          TEST ASYNCHRONOUS LINES WITH OLD BAUD RATES
                         *             9600 BAUD, WITH 2 STOP BITS
                         *
071E  9B80 11BA                 LAB    $B1,<ASYN              STORE ADDRESS OF LINE TYPE
0720  B380 0A8C                 LNJ    $B3,<DTALOP            GO TO LOOP SUB-ROUTINE
0722  0F85                      B      >T39
0723  1322                      DC     <WCP2                  ADDRESS TABLE FOR BLOCK WRITE
0724  12D7                      DC     <LCT2                  LCT TABLE
0725  003F                      DC     (DFLT-DOUT)            RANGE IN WORDS
0726  11E3                      DC     <DOUT
                         *
                         *
                         *          TEST ASYCHRONOUS LINES WITH NEW BAUD RATES
                         *             19200 BAUD, WITH 2 STOP BITS
0727  9B80 11BB          T39    LAB    $B1,<ASYN1             LOAD  LINE TYPE
0729  B380 0A8C          T39A   LNJ    $B3,<DTALOP            GO TO LOOP SUB-ROUTINE
```

```
001598  072B  0F85                        B       >T40
001599  072C  1322                        DC      <WCP2                   ADDRESS TABLE FOR BLOCK WRITE
001600  072D  12D7                        DC      <LCT2                   LCT TABLE
001601  072E  003F                        DC      (DFLT-DOUT)             RANGE IN WORDS
001602  072F  11E3                        DC      <DOUT
001603                              *
001604                              *           TEST CURRENT LOOP ASYNCHRONOUS
001605                              *
001606  0730  C871                  T40   LDR     $R4,+$B1                LOAD ID TYPE INTO $R4
001607  0731  C900 11BC                    CMR     $R4,<ASYNC              CHECK TO SEE IF ALREADY DONE
001608  0733  0902                        BE      >T40A
001609  0734  0FF5                        B       >T39A
001610        0735                  T40A  EQU     $
001611  0735  F870 3218                    LDR     $R7,=X'3218'            STORE SYNC. CHAR. INTO LCT TABLE
001612  0737  FF00 12DB                    STR     $R7,<LCT2S
001613  0739  7C28                        LDV     $R7,=40                 NUMBER OF 30 MS TIME DELAYS
001614                              *
001615                              *           TEST SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
001616                              *
001617  073A  9B80 11BD                    LAB     $B1,<SYNC               LOAD LINE TYPE
001618  073C  B380 0A8C             QA4    LNJ     $B3,<DTALOP             GO TO LOOP SUB-ROUTINE
001619  073E  0F85                        B       >T41
001620  073F  1322                        DC      <WCP2                   ADDRESS TABLE FOR BLOCK WRITE
001621  0740  12D7                        DC      <LCT2                   LCT TABLE
001622  0741  002F                        DC      (DFLT-DOUT1)            RANGE IN WORDS
001623  0742  11F3                        DC      <DOUT1
001624                              *
001625                              *           TEST BI-SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
001626                              *
001627  0743  C871                  T41   LDR     $R4,+$B1                LOAD ID TYPE INTO $R4
001628  0744  C900 11BE                    CMR     $R4,<BISYNC             CHECK TO SEE IF ALREADY DONE
001629  0746  0902                        BE      >T42
001630  0747  0FF5                        B       >QA4
001631                              *
001632                              *           TEST MIL 188 SYNCHRONOUS (32 AS SYNC. CHAR.)
001633                              *
001634  0748  9B80 11BF             T42    LAB     $B1,<SYN188             LOAD LINE TYPE
001635  074A  B380 0A8C                    LNJ     $B3,<DTALOP             GO TO LOOP SUB-ROUTINE
001636  074C  0F85                        B       >T43
001637  074D  1322                        DC      <WCP2                   IORB ADDRESS FOR BLOCK WRITE
001638  074E  12D7                        DC      <LCT2                   LCT TABLE
001639  074F  002F                        DC      (DFLT-DOUT1)            RANGE IN WORDS
001640  0750  11F3                        DC      <DOUT1
001641                              *
001642                              *
001643                              *           TEST WIDEBAND 301/303 (3A AS SYNC. CHARACTER)
001644                              *
001645  0751  9B80 1322             T43    LAB     $B1,<WCP2               LOAD ADDRESS OF MEDIUM SPEED ADDRESS TABLE
001646  0753  8980 115E                    CMZ     <HGSPD                  CHECK FOR HIGH SPEED LINE
```

```
0755  0903                BE     >T43X
0756  9B80 133F           LAB    $B1,<WCP8          LOAD ADDRESS OF HIGH SPEED ADDRESS TABLE
0758  9F80 076F   T43X    STB    $B1,<T43Z          STORE INTO DTALOP SUB-ROUTINE
075A  9F80 0761           STB    $B1,<T43Y
075C  9B80 11C0           LAB    $B1,<WDB3          LOAD ADDRESS OF LINE TYPE
075E  B380 0A8C   T43YY   LNJ    $B3,<DTALOP        LOOP SUB-ROUTINE
0760  0F85                B      >T44A
0761  0000        T43Y    RESV   $AF,0
0762  12D7                DC     <LCT2
0763  003F                DC     (DFLT-DOUT)
0764  11E3                DC     <DOUT
                  *
                  *             TEST BI-SYNC. WIDEBAND 301/303 (3A AS SYNC. CHAR.)
                  *
0765  C871        T44A    LDR    $R4,+$B1           LOAD ID TYPE INTO $R4
0766  C900 11C1           CMR    $R4,<WDB3B         CHECK TO SEE IF ALREADY DONE
0768  0902                BE     >T44
0769  0FF5                B      >T43YY
                  *
                  *             TEST WIDEBAND V35 (3A AS SYNC. CHAR.)
                  *
076A  9B80 11C2   T44     LAB    $B1,<WDBV35        LOAD ADDRESS OF LINE TYPE
076C  B380 0A8C   T43ZZ   LNJ    $B3,<DTALOP
076E  0F85                B      >T45A
076F  0000        T43Z    RESV   $AF,0
0770  12D7                DC     <LCT2
0771  003F                DC     (DFLT-DOUT)
0772  11E3                DC     <DOUT
                  *
                  *             TEST BI-SYNC. WIDEBAND V35 (3A AS SYNC. CHAR.)
                  *
0773  C871        T45A    LDR    $R4,+$B1           LOAD ID TYPE INTO $R4
0774  C900 11C3           CMR    $R4,<WDBV5B        CHECK TO SEE IF ALREADY DONE
0776  0902                BE     >T45
0777  0FF5                B      >T43ZZ
                  *
0778  8700 1163   T45     CL     <PARFG
077A  8382                JMP    $B2
                  *
                  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                  *
                  *              CHECK PARITY ERROR BIT SET
                  *
077B  C380 0D25   PAR1    LNJ    $B4,<INXT          INPUT  NEXT STATUS
077D  82D5                LB     =$R5,=X'1000'
077E  1000
077F  0503                BBT    >PAR2
0780  F380 0DBA           LNJ    $B7,<ERMB          ERROR: STATUS COMPLETE BIT NOT SET
0782  82D5        PAR2    LB     =$R5,=X'40'        SEE IF DATA CHECK ERROR BIT SET
```

```
0783  0040
0784  0503                     BBT    >PAR3
0785  F380 0DBA                LNJ    $B7,<ERMB            ERROR:  DATA CHECK ERROR BIT NOT SET
0787  8AD3           PAR3      INC    =$R3
0788  C380 0D25                LNJ    $B4,<INXT            INPUT NEXT STATUS
078A  82D5                     LB     =$R5,=X'1000'
078B  1000
078C  0503                     BBT    >PAR4
078D  F380 0DBA                LNJ    $B7,<ERMB            ERROR:  STATUS NOT COMPLETE ON XMIT SIDE
078F  82D5           PAR4      LB     =$R5,=X'40'
0790  0040
0791  0583                     BBF    >PAR5
0792  F380 0DBA                LNJ    $B7,<ERMB            ERROR:  DATA CHECK ERROR BIT SET ON XMIT
0794  0F80 0BA0      PAR5      B      <JP5A
                     *
                     *
                     * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                     *
                     *             STOP I/O  TEST
                     *
0796  C870 4820      STIOG     LDR    $R4,=A'H '
0798  CF00 13A9                STR    $R4,<ERMG+1
079A  B380 0BA6                LNJ    $B3,<CLFG            CLEAR ALL TEST FLAGS
079C  A800 11C6                LDR    $R2,<BIT2            LOAD $R2 WITH FUNCTION COMMAND
079E  8A80 1170                INC    <STPFG
07A0  8700 12F6                CL     <LC5SYF              CLEAR SYNC. FLAG FOR ASYNC. LINES
07A2  C870 0F18                LDR    $R4,=X'F18'
07A4  CF00 12F2                STR    $R4,<LCT5S
07A6  7C02                     LDV    $R7,=2               NUMBER OF 30 MS DELAY
                     *
                     *             TEST ASYNCHRONOUS LINES WITH OLD BAUD RATES
                     *                AT 9600 BAUD, WITH 2 STOP BITS
                     *
07A7  9B80 11BA                LAB    $B1,<ASYN            STORE ADDRESS OF LINE TYPE
07A9  B380 0A8C                LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
07AB  0F85                     B      >T19
07AC  131D                     DC     <WCP1                ADDRESS TABLE FOR BLOCK WRITE
07AD  12EE                     DC     <LCT5                LCT TABLE
07AE  01B6                     DC     438                  RANGE IN WORDS
07AF  11E3                     DC     <DOUT                TRANSMIT BUFFER
                     *
                     *             TEST ASYNCHRONOUS LINES WITH NEW BAUD RATES
                     *                AT 19200 BAUD, WITH 2 STOP BITS
                     *
07B0  9B80 11BB      T19       LAB    $B1,<ASYN1           LINE TYPE
07B2  B380 0A8C      T19A      LNJ    $B3,<DTALOP
07B4  0F85                     B      >T20
07B5  131D                     DC     <WCP1
07B6  12EE                     DC     <LCT5
```

```
001741  07B7  01B6                      DC    438                 RANGE IN WORDS
001742  07B8  11E3                      DC    <DOUT               TRANSMIT BUFFER ADDRESS
001743                           *
001744                           *           TEST CURRENT LOOP ASYNCHRONOUS
001745                           *
001746  07B9  C871               T20     LDR   $R4,+$B1            LOAD ID TYPE INTO $R4
001747  07BA  C900 11BC                  CMR   $R4,<ASYNC          CHECK TO SEE IF ALREADY DONE
001748  07BC  0902                      BE    >T20A
001749  07BD  0FF5                      B     >T19A
001750        07BE               T20A    EQU   $
001751  07BE  C870 3A18                  LDR   $R4,=X'3A18'        PUT IN SYNC. CHARACTER TO TEST
001752  07C0  CF00 12F2                  STR   $R4,<LCT5S          SYNC. LINE TYPES
001753  07C2  C870 021C                  LDR   $R4,=X'21C'         SET LCT SYNC. FLAG
001754  07C4  CF00 12F6                  STR   $R4,<LC5SYF
001755  07C6  7C05                      LDV   $R7,=5              NUMBER OF 30 MS DELAY FOR 800 BAUD
001756                           *
001757                           *           TEST SYNCHRONOUS (3A AS SYNC. CHAR.)
001758                           *
001759  07C7  9B80 11BD                  LAB   $B1,<SYNC           LOAD LINE TYPE
001760  07C9  B380 0A8C          QA5     LNJ   $B3,<DTALOP         LOOP SUB-ROUTINE
001761  07CB  0F85                      B     >T21
001762  07CC  131D                      DC    <WCP1
001763  07CD  12EE                      DC    <LCT5
001764  07CE  01B6                      DC    438
001765  07CF  11F3                      DC    <DOUT1
001766                           *
001767                           *           TEST BI-SYNCHRONOUS LINES (3A AS SYNC. CHAR.)
001768                           *
001769  07D0  C871               T21     LDR   $R4,+$B1            LOAD ID TYPE INTO $R4
001770  07D1  C900 11BE                  CMR   $R4,<BISYNC         CHECK TO SEE IF ALREADY DONE
001771  07D3  0902                      BE    >T22
001772  07D4  0FF5                      B     >QA5
001773                           *
001774                           *           TEST MIL 188 SYNCHRONOUS (3A AS SYNC. CHAR.)
001775                           *
001776  07D5  9B80 11BF          T22     LAB   $B1,<SYN188         LOAD LINE TYPE
001777  07D7  B380 0A8C                  LNJ   $B3,<DTALOP         LOOP SUB-ROUTINE
001778  07D9  0F85                      B     >T23
001779  07DA  131D                      DC    <WCP1
001780  07DB  12EE                      DC    <LCT5
001781  07DC  01B6                      DC    438
001782  07DD  11F3                      DC    <DOUT1
001783                           *
001784                           *           TEST WIDEBAND 301/303 (3A AS SYNC. CHARACTER)
001785                           *
001786  07DE  8A80 1171          T23     INC   <WDBFG              SET BROADBAND FLAG
001787  07E0  8757                      CL    =$R7
001788  07E1  9B80 11C0                  LAB   $B1,<WDB3           LOAD ADDRESS OF LINE TYPE
001789  07E3  B380 0A8C          T23A    LNJ   $B3,<DTALOP         LOOP SUB-ROUTINE
```

```
001790  07E5  0F85                        B       >T33A
001791  07E6  1338                        DC      <WCP7
001792  07E7  12EE                        DC      <LCT5
001793  07E8  01B6                        DC      438
001794  07E9  11E3                        DC      <DOUT
001795                            *
001796                            *           TEST BI-SYNC. WIDEBAND/301/303 (3A AS SYNC. CHAR.)
001797                            *
001798  07EA  C871                T33A    LDR     $R4,+$B1              LOAD ID TYPE INTO $R4
001799  07EB  C900 11C1                   CMR     $R4,<WDB3B            CHECK TO SEE IF ALREADY DONE
001800  07ED  0902                        BE      >T33
001801  07EE  0FF5                        B       >T23A
001802                            *
001803                            *           TEST WIDEBAND V35 (3A AS SYNC. CHAR.)
001804                            *
001805  07EF  9B80 11C2           T33     LAB     $B1,<WDBV35           LOAD ADDRESS OF LINE TYPE
001806  07F1  B380 0A8C           T33Z    LNJ     $B3,<DTALOP
001807  07F3  0F85                        B       >T34A
001808  07F4  1338                        DC      <WCP7
001809  07F5  12EE                        DC      <LCT5
001810  07F6  01B6                        DC      438
001811  07F7  11E3                        DC      <DOUT
001812                            *
001813                            *           TEST BI-SYNC. WIDEBAND V35 (3A AS SYNC. CHAR.)
001814                            *
001815  07F8  C871                T34A    LDR     $R4,+$B1              LOAD ID TYPE INTO $R4
001816  07F9  C900 11C3                   CMR     $R4,<WDBV5B           CHECK TO SEE IF ALREADY DONE
001817  07FB  0902                        BE      >T34
001818  07FC  0FF5                        B       >T33Z
001819                            *
001820  07FD  8700 1170           T34     CL      <STPFG                CLEAR STOP I/O FLAG
001821  07FF  8382                        JMP     $B2
001822                            *
001823                            *
001824                            * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
001825                            *
001826                            *           STOP I/O  CHECK ROUTINE
001827                            *
001828  0800  8AD3                STIQ    INC     =$R3                  CHANGE TO TRASMIT SIDE OF LINE
001829  0801  C380 0D1E           STIQ1   LNJ     $B4,<STCR             STOP I/O
001830                            *
001831  0803  C380 0D25                   LNJ     $B4,<INXT             INPUT NEXT STATUS
001832  0805  82D5                STIQ5   LB      =$R5,=Z'1000'         CHECK STATUS COMPLETE
        0806  1000
001833  0807  0503                        BBT     >TSTF3
001834  0808  F380 0DC2                   LNJ     $B7,<ERDB             ERROR:  STATUS NOT COMPLETE
001835  080A  C380 0D2B           TSTF3   LNJ     $B4,<INRG             INPUT RANGE
001836  080C  5983                        BNEZ    $R5,>TSTF4
001837  080D  F380 0DC2                   LNJ     $B7,<ERDB             ERROR:  RANGE IS EQUAL TO ZERO
```

```
001838  080F  3B00 082E          TSTF4   BEVN    $R3,<TSTF4A         IF EVEN BOTH SIDE OF THE LINE SHUT-OFF
001839  0811  88D3                        DEC     =$R3                CHANGE $R3 TO RECEIVE SIDE OF THE LINE
001840  0812  DF00 116C                    STR     $R5,<TP1            STORE XMIT RANGE RESIDUE
001841  0814  8980 1171                    CMZ     <WDBFG              SEE IF BROADBAND LINE
001842  0816  0914                        BE      >STIQ2              SHUT RECV. LINE OFF AFTER 70 TRANSFERS
001843  0817  C380 0D25                    LNJ     $B4,<INXT           INPUT NEXT STATUS
001844  0819  C380 0D2B          STIQ3   LNJ     $B4,<INRG           INPUT RANGE
001845  081B  D970 0220                    CMR     $R5,=X'220'         SEE IF 166 TRANSFERS HAVE BEEN DONE
001846  081D  0A7C                        BAG     >STIQ3              IF DONE
001847  081E  C380 0D1E                    LNJ     $B4,<STCR           STOP I/O ON RECV.
001848                          *
001849  0820  C800 1144                    LDR     $R4,<FIS            INPUT STATUS
001850  0822  C380 0D95                    LNJ     $B4,<CGSCH
001851  0824  8055                        IO      =$R5,=$R4
        0825  0054
001852  0826  0703                        BIOT    >STIQ4
001853  0827  F380 0DBA                    LNJ     $B7,<ERMB
001854  0829  0FDC                  STIQ4   B       >STIQ5
001855                          *
001856  082A  7C03                  STIQ2   LDV     $R7,=3
001857  082B  D380 0D63                    LNJ     $B5,<TDLAY          TIME DELAY BEFORE SHUTTING OFF RECEIVE
001858  082D  0FD4                        B       >STIQ1
001859                          *
001860                          *         CHECK RECEIVED DATA
001861                          *
001862  082E  C800 113F          TSTF4A  LDR     $R4,<FOCC           CCB RESET FOR REC
001863  0830  C380 0D95                    LNJ     $B4,<CGSCH
001864  0832  8000 11C9                    IO      <BIT7,=$R4
        0834  0054
001865  0835  0703                        BIOT    >QT2
001866                          *
001867  0836  F380 0DBA                    LNJ     $B7,<ERMB           ERROR:  I/O WAS NAK'ED
001868                          *
001869  0838  8AD3                  QT2     INC     =$R3
001870  0839  C800 113F                    LDR     $R4,<FOCC           CCB RESET FOR TRANSMIT
001871  083B  C380 0D95                    LNJ     $B4,<CGSCH
001872  083D  8000 11C9                    IO      <BIT7,=$R4
        083F  0054
001873  0840  0703                        BIOT    >QT1
001874                          *
001875  0841  F380 0DBA                    LNJ     $B7,<ERMB           ERROR:  I/O WAS NAK'ED
001876                          *
001877  0843  DF00 116D          QT1     STR     $R5,<TP2            STORE RECEIVE RANGE RESIDUE
001878  0845  E380 0BB7                    LNJ     $B6,<BASE           LOAD $R5 WITH LCT BASE
001879  0847  5E10                        ADV     $R5,=16
001880  0848  DF00 084F                    STR     $R5,<TSTF2B
001881  084A  5E20                        ADV     $R5,=X'20'
001882  084B  DF00 0858                    STR     $R5,<TSTF8B
001883                          *
```

```
001884  084D  E380 0D75            LNJ   $B6,<RDLCT          READ LCT
001885  084F  0000          TSTF2B  DC    0                   ADDRESS
001886  0850  0002                  DC    2
001887                        *
001888  0851  8980 1173            CMZ   <LCTV
001889  0853  0903                  BE    >TSTF7K
001890  0854  F380 0DC2            LNJ   $B7,<ERDB           ERROR:  RECV. LCT STATUS NOT CLEAR
001891                        *
001892  0856  E380 0D75    TSTF7K  LNJ   $B6,<RDLCT          READ LCT
001893  0858  0000          TSTF8B  DC    0
001894  0859  0002                  DC    2
001895                        *
001896  085A  8980 1173            CMZ   <LCTV               CHECK LCT STATUS IS CLEARED
001897  085C  0903                  BE    >TSTF2
001898  085D  F380 0DC2            LNJ   $B7,<ERDB           ERROR:  LCT STATUS IS NOT CLEAR
001899  085F  9800 1166    TSTF2   LDR   $R1,<RANGE          CALCULATE RANGE OF GOOD DATA TO BE CHECK
001900  0861  9F00 115C            STR   $R1,<FBRG
001901  0863  1001                  SOL   $R1,1               CHANGE TO BYTE RANGE
001902  0864  9200 116C            SUB   $R1,<TP1
001903  0866  1041                  SOR   $R1,1               CHANGE TO WORDS
001904  0867  88D1                  DEC   =$R1
001905  0868  9F00 1166            STR   $R1,<RANGE
001906                        *
001907  086A  8980 12F6            CMZ   <LC5SYF             CHECK FOR SYNC LINE
001908  086C  0917                  BE    >TSTF4C
001909  086D  1E04                  ADV   $R1,=4
001910  086E  C800 116C            LDR   $R4,<TP1            SYNC LINE CHECK FOR LINE MARKING
001911  0870  C200 116D            SUB   $R4,<TP2
001912  0872  4041                  SOR   $R4,1
001913  0873  CA51                  ADD   $R4,=$R1
001914  0874  4EF9                  ADV   $R4,-7
001915  0875  CF00 116D            STR   $R4,<TP2
001916  0877  C870 AAAA            LDR   $R4,=Z'AAAA'        LINE MARKING DATA
001917  0879  C910 1C00    TSTF5A  CMR   $R4,<DIN.$R1
001918  087B  0903                  BE    >TSTF5
001919  087C  F380 0DC2            LNJ   $B7,<ERDB           ERROR:  SYNC. LINE NOT MARKING WITH XMIT OFF
001920  087E  8AD1          TSTF5   INC   =$R1
001921  087F  9900 116D            CMR   $R1,<TP2            SEE IF ALL MARKING DATA CHECK
001922  0881  0882                  BAGE  >TSTF4C
001923  0882  0FF7                  B     >TSTF5A
001924                        *
001925  0883  C800 1222    TSTF4C  LDR   $R4,<DFLT           CHECK REMAINING DATA IS DEFAULT (5555)
001926  0885  1E07                  ADV   $R1,=7
001927  0886  C910 1C00    TSTF4D  CMR   $R4,<DIN.$R1
001928  0888  0903                  BE    >TSTF4E
001929  0889  F380 0DC2            LNJ   $B7,<ERDB           ERROR:RECEIVE DATA IN NO DATA AREA
001930  088B  8AD1          TSTF4E  INC   =$R1
001931  088C  9900 115C            CMR   $R1,<FBRG
001932  088E  0878                  BAL   >TSTF4D
```

```
001933  088F  0F80 0B53              B       <JP4                CHECK FINAL GOOD DATA IN DTALOP
001934                          *
001935                          *
001936                          * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
001937                          *
001938                          *
001939                          *           UNDERRUN AND OVERRUN TEST
001940                          *
001941  0891  C870 4A20          UORNTH LDR    $R4,=A'J '
001942  0893  CF00 13A9                 STR     $R4,<ERMG+1
001943  0895  B380 0BA6                 LNJ     $B3,<CLFG           CLEAR ALL TEST FLAGS
001944  0897  8700 1313                 CL      <LCT8FG
001945  0899  8A80 1171                 INC     <WDBFG
001946  089B  8A80 116A                 INC     <RUNFG              SET UNDERRUN AND OVERRUN TEST FLAG
001947  089D  C870 021D                 LDR     $R4,=X'21D'
001948  089F  CF00 12FF                 STR     $R4,<LCT6TF
001949  08A1  7C06                      LDV     $R7,=6              NUMBER OF 30 MS TIME DELAY FOR TRANSFER
001950  08A2  C870 0F18                 LDR     $R4,=X'F18'         SET LINE SPEED FOR ASYC. LINES
001951  08A4  CF00 12FD                 STR     $R4,<LCT6S
001952                          *
001953                          *             TEST ASYNCHRONOUS LINES WITH OLD BAUD RATES
001954                          *                9600 BAUD, WITH 1 STOP BIT
001955                          *
001956  08A6  9B80 11BA                 LAB     $B1,<ASYN           STORE ADDRESS OF LINE TYPE
001957  08A8  B380 0A8C                 LNJ     $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
001958  08AA  0F85                      B       >T24
001959  08AB  132C                      DC      <WCP4               ADDRESS TABLE FOR BLOCK WRITE
001960  08AC  12F8                      DC      <LCT6               LCT TABLE
001961  08AD  003F                      DC      (DFLT-DOUT)         RANGE IN WORDS
001962  08AE  11E3                      DC      <DOUT
001963                          *
001964                          *             TEST ASYCHRONOUS LINES WITH NEW BAUD RATES
001965                          *                  19200 BAUD, WITH 1 STOP BIT
001966                          *
001967  08AF  9B80 11BB          T24    LAB     $B1,<ASYN1          LOAD  LINE TYPE
001968  08B1  B380 0A8C          T24Z   LNJ     $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
001969  08B3  0F85                      B       >T25
001970  08B4  132C                      DC      <WCP4               ADDRESS TABLE FOR BLOCK WRITE
001971  08B5  12F8                      DC      <LCT6               LCT TABLE
001972  08B6  003F                      DC      (DFLT-DOUT)         RANGE IN WORDS
001973  08B7  11E3                      DC      <DOUT
001974                          *
001975                          *             TEST CURRENT LOOP ASYNCHRONOUS
001976                          *
001977  08B8  C871               T25    LDR     $R4,+$B1            LOAD ID TYPE INTO $R4
001978  08B9  C900 11BC                 CMR     $R4,<ASYNC          CHECK TO SEE IF ALREADY DONE
001979  08BB  0902                      BE      >T25Z
001980  08BC  0FF5                      B       >T24Z
001981        08BD               T25Z   EQU     $
```

```
001982  08BD  8700 1171              CL    <WDBFG
001983  08BF  F870 401D              LDR   $R7,=Z'401D'
001984  08C1  FF00 12FF              STR   $R7,<LCT6TF
001985  08C3  F870 1618              LDR   $R7,=Z'1618'          STORE SYNC. CHAR. INTO LCT TABLE
001986  08C5  FF00 12FD              STR   $R7,<LCT6S
001987  08C7  7C40                   LDV   $R7,=X'40'            NUMBER OF 30 MS TIME DELAYS
001988                          *
001989                          *           TEST SYNCHRONOUS LINES (16 AS SYNC. CHAR.)
001990                          *
001991  08C8  9B80 11BD              LAB   $B1,<SYNC             LOAD LINE TYPE
001992  08CA  B380 0A8C        QA6   LNJ   $B3,<DTALOP           GO TO LOOP SUB-ROUTINE
001993  08CC  0F85                   B     >T26
001994  08CD  132C                   DC    <WCP4               ADDRESS TABLE FOR BLOCK WRITE
001995  08CE  12F8                   DC    <LCT6               LCT TABLE
001996  08CF  002F                   DC    (DFLT-DOUT1)        RANGE IN WORDS
001997  08D0  11F3                   DC    <DOUT1
001998                          *
001999                          *           TEST BI-SYNCHRONOUS LINES (16 AS SYNC. CHAR.)
002000                          *
002001  08D1  C871             T26   LDR   $R4,+$B1            LOAD ID TYPE INTO $R4
002002  08D2  C900 11BE              CMR   $R4,<BISYNC           CHECK TO SEE IF ALREADY DONE
002003  08D4  0902                   BE    >T27
002004  08D5  0FF5                   B     >QA6
002005                          *
002006                          *           TEST MIL 188 SYNCHRONOUS (16 AS SYNC. CHAR.)
002007                          *
002008  08D6  9B80 11BF        T27   LAB   $B1,<SYN188           LOAD LINE TYPE
002009  08D8  B380 0A8C              LNJ   $B3,<DTALOP           GO TO LOOP SUB-ROUTINE
002010  08DA  0F85                   B     >T28
002011  08DB  132C                   DC    <WCP4               ADDRESS TABLE FOR BLOCK WRITE
002012  08DC  12F8                   DC    <LCT6               LCT TABLE
002013  08DD  002F                   DC    (DFLT-DOUT1)        RANGE IN WORDS
002014  08DE  11F3                   DC    <DOUT1
002015                          *
002016                          *
002017                          *           TEST WIDEBAND 301/303 (3A AS SYNC. CHARACTER)
002018                          *
002019  08DF  8A80 1171        T28   INC   <WDBFG
002020  08E1  C870 011E              LDR   $R4,=X'11E'           SET LCT FLAG FOR OVERRUN
002021  08E3  CF00 1313              STR   $R4,<LCT8FG
002022  08E5  9B80 11C0        T48   LAB   $B1,<WDB3             LOAD ADDRESS OF LINE TYPE
002023  08E7  B380 0A8C        T48Z  LNJ   $B3,<DTALOP           LOOP SUB-ROUTINE
002024  08E9  0F85                   B     >T31A
002025  08EA  1333                   DC    <WCP6
002026  08EB  130B                   DC    <LCT8
002027  08EC  003F                   DC    (DFLT-DOUT)
002028  08ED  11E3                   DC    <DOUT
002029                          *
002030                          *           TEST BI-SYNC. WIDEBAND 301/303 (3A AS SYNC. CHAR.)
```

```
                            *
08EE  C871                  T31A   LDR    $R4,+$B1           LOAD ID TYPE INTO $R4
08EF  C900 11C1                    CMR    $R4,<WDB3B         CHECK TO SEE IF ALREADY DONE
08F1  0902                         BE     >T31
08F2  0FF5                         B      >T48Z
                            *
                            *             TEST WIDEBAND V35 (3A AS SYNC. CHAR.)
                            *
08F3  9B80 11C2             T31    LAB    $B1,<WDBV35        LOAD ADDRESS OF LINE-TYPE
08F5  B380 0A8C             T31Z   LNJ    $B3,<DTALOP
08F7  0F85                         B      >T32A
08F8  1333                         DC     <WCP6
08F9  130B                         DC     <LCT8
08FA  003F                         DC     (DFLT-DOUT)
08FB  11E3                         DC     <DOUT
                            *
                            *             TEST BI-SYNC. WIDEBAND V35 (3A AS SYNC. CHAR.)
                            *
08FC  C871                  T32A   LDR    $R4,+$B1           LOAD ID TYPE INTO $R4
08FD  C900 11C3                    CMR    $R4,<WDBV5B        CHECK TO SEE IF ALREADY DONE
08FF  0902                         BE     >T32
0900  0FF5                         B      >T31Z
                            *
0901  8980 1171             T32    CMZ    <WDBFG
0903  080B                         BAL    >T49
0904  8700 1171                    CL     <WDBFG
0906  8880 1171                    DEC    <WDBFG
0908  C870 0F1E                    LDR    $R4,=X'F1E'
090A  CF00 1313                    STR    $R4,<LCT8FG
090C  0F80 08E5                    B      <T48
090E  8700 1171             T49    CL     <WDBFG
0910  8700 116A                    CL     <RUNFG
0912  8700 1313                    CL     <LCT8FG
0914  8382                         JMP    $B2
                            *
                            *
                            * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                            *
                            *          UNDERRUN AND OVERRUN CHECK ROUTINE
                            *
0915  88D3                  RUNCK  DEC    =$R3
0916  E380 0BB7                    LNJ    $B6,<BASE          LOAD $R5 WITH LCT BASE ADDRESS
0918  5E3C                         ADV    $R5,=X'3C'
0919  DF00 0940                    STR    $R5,<RNCK1
                            *
091B  8980 1171                    CMZ    <WDBFG
091D  080F                         BAL    >RNCK4
091E  C800 1144                    LDR    $R4,<FIS
0920  C380 0D95                    LNJ    $B4,<CGSCH         INPUT STATUS TO CHECK DATA
```

```
002080  0922  8055                      IO      =$R5,=$R4           SERVICE ERROR (BIT2) SET
        0923  0054
002081  0924  0703                      BIOT    >RNCK4A
002082  0925  F380 0DBA                  LNJ     $B7,<ERMB           ERROR:  I/O NAK'ED
002083  0927  82D5                RNCK4A LB      =$R5,=Z'2000'
        0928  2000
002084  0929  0503                      BBT     >RNCK4
002085  092A  F380 0DC2                  LNJ     $B7,<ERDB           ERROR:  DATA SERVICE ERROR BIT NOT SET ON OVR
002086  092C  8AD3                RNCK4  INC     =$R3                CHECK XMIT DATA SERVICE ERROR
002087  092D  8980 1171                  CMZ     <WDBFG
002088  092F  0A0F                      BAG     >RNCK5
002089  0930  C800 1144                  LDR     $R4,<FIS
002090  0932  C380 0D95                  LNJ     $B4,<CGSCH
002091  0934  8055                      IO      =$R5,=$R4
        0935  0054
002092  0936  0703                      BIOT    >RNCK4B
002093  0937  F380 0DBA                  LNJ     $B7,<ERMB           ERROR:  I/O NAK'ED
002094                              *
002095  0939  82D5                RNCK4B LB      =$R5,=Z'2000'
        093A  2000
002096  093B  0503                      BBT     >RNCK5
002097  093C  F380 0DC2                  LNJ     $B7,<ERDB           ERROR:  DATA SERVICE ERROR BIT NOT SET ON UDR
002098  093E  E380 0D75            RNCK5  LNJ     $B6,<RDLCT          READ LCT
002099  0940  0000                RNCK1  DC      0
002100  0941  0002                       DC      2
002101                              *
002102  0942  8980 1171                  CMZ     <WDBFG              CHECK FLAG
002103  0944  080A                      BAL     >RNCK9
002104  0945  8280 1173                  LB      <LCTV,=Z'0200'
        0947  0200
002105  0948  050C                      BBT     >RNCK6
002106  0949  F380 0DC2                  LNJ     $B7,<ERDB           ERROR:  OVERRUN FAIL TO SET BIT IN LR5
002107  094B  8980 1171                  CMZ     <WDBFG              CHECK FLAG TO TEST UNDERRUN BIT
002108  094D  0987                      BNE     >RNCK6
002109                              *
002110  094E  8280 1173            RNCK9  LB      <LCTV,=Z'0100'
        0950  0100
002111  0951  0503                      BBT     >RNCK6
002112  0952  F380 0DC2                  LNJ     $B7,<ERDB           ERROR:  UNDERRUN FAIL TO SET BIT IN LR5
002113  0954  0F80 0BA0            RNCK6  B       <JP5A
002114                              *
002115                              *
002116                              * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
002117                              *
002118                              *           LOOP DSC INTO DSS WITH CABLE LOOP
002119                              *
002120                              *
002121                              *            THE SIGNALS ARE TIED IN ACCORDING TO THE FOLLOWING CHART:
002122                              *
```

```
                                 *  --------------------------------------------------------------------------------
                                 *  $             $             $             $             $             $
                                 *  $     DSS     $ASYNCHRONOUS$SYNCHRONOUS $  BROADBAND  $  BROADBAND  $
                                 *  $             $             $ & MIL 188 $   301/303   $     V35     $
                                 *  $_____________$_____________$_____________$_____________$_____________$
                                 *  $             $             $             $             $             $
                                 *  $  OUT CKT 1  $  IN CKT 1   $  IN CKT 1   $  IN CKT 4   $  IN CKT 1   $
                                 *  $             $             $             $             $  IN CKT 4   $
                                 *  $_____________$_____________$_____________$_____________$_____________$
                                 *  $             $             $             $             $             $
                                 *  $  OUT CKT 2  $  IN CKT 2   $  IN CKT 2   $  IN CKT 1   $  IN CKT 2   $
                                 *  $             $  IN CKT 3   $  IN CKT 3   $  IN CKT 2   $  IN CKT 3   $
                                 *  $             $             $             $  IN CKT 3   $             $
                                 *  $_____________$_____________$_____________$_____________$_____________$
                                 *  $             $             $             $             $             $
                                 *  $  OUT CKT 3  $  IN CKT 4   $     ---     $    ----     $    ----     $
                                 *  $             $  IN CKT 5   $             $             $             $
                                 *  $_____________$_____________$_____________$_____________$_____________$
                                 *  $             $             $             $             $             $
                                 *  $  OUT CKT 4  $     ---     $  IN CKT 4   $    ----     $    ----     $
                                 *  $_____________$_____________$_____________$_____________$_____________$
                                 *
                                 *
0956  C870 4B20                  LPCBLK LDR   $R4,=A'K '
0958  CF00 13A9                         STR   $R4,<ERMG+1
095A  B380 0BA6                         LNJ   $B3,<CLFG          CLEAR ALL TEST FLAGS
095C  8700 1160                         CL    <MASK
095E  9B80 127A                         LAB   $B1,<TLTBLA        PUT ADDRESS OF SHOULD BE TABLE IN ZV$C
0960  9F80 0A05                         STB   $B1,<LOPD7+4+$AF
0962  9B80 12A3                         LAB   $B1,<C4            PUT ADDRESS OF RANGE IN ZV$C
0964  9F80 0A08                         STB   $B1,<LOPD7+4+4*$AF
0966  C870 FC1D                         LDR   $R4,=Z'FC1D'
0968  CF00 1306                         STR   $R4,<LC7SM         SET MASK FO ASYC. LINE
096A  4C08                              LDV   $R4,=8             SET RANGE FOR CHECK ROUTINE
096B  CF00 116D                         STR   $R4,<TP2
                                 *##
                                 *#
096D  F870 00FA                         LDR   $R7,=250           LOAD $R7 WITH NUMBER OF 30MS DELAY
                                 *#
                                 *##
                                 *
096F  9B80 11BA                         LAB   $B1,<ASYN          LOAD $B1 WITH ADDRESS OF LINE TYPE
0971  B380 09C3                         LNJ   $B3,<LOPDS         BRANCH TO LOOP DSS INTO DSC AND CHECK
                                 *
0973  9B80 11BB                         LAB   $B1,<ASYN1         LOAD ADDRESS OF LINE TYPE
0975  B380 09C3                         LNJ   $B3,<LOPDS
                                 *
0977  9B80 127E                         LAB   $B1,<TLTBLC        STORE SHOULD BE ADDRESS FOR ZV$C
0979  9F80 0A05                         STB   $B1,<LOPD7+4+$AF
```

```
002172  097B  9B80 11BC                LAB    $B1,<ASYNC          LOAD ADDRESS OF LINE TYPE
002173  097D  B380 09C3                LNJ    $B3,<LOPDS
002174                          *
002175  097F  9B80 1282                LAB    $B1,<TLTBLS
002176                          *##
002177                          *#
002178  0981  C800 1149                LDR    $R4,<TEST           DON'T CHECK RING INDICATOR IF 'D' OR 'M'
002179  0983  4D44                     CMV    $R4,X'44'
002180  0984  0905                     BE     >LPCB1
002181  0985  4D4D                     CMV    $R4,X'4D'
002182  0986  0987                     BNE    >LPCB2
002183  0987  9B80 1286                LAB    $B1,<TLTBLM
002184  0989  C870 E0E0          LPCB1 LDR    $R4,=Z'E0E0'
002185  098B  CF00 1160                STR    $R4,<MASK
002186                          *#
002187                          *##
002188  098D  9F80 0A05          LPCB2 STB    $B1,<LOPD7+4+$AF
002189  098F  C870 F81D                LDR    $R4,=Z'F81D'        CHANGE MASK FOR SYNC. LINES
002190  0991  CF00 1306                STR    $R4,<LC7SM
002191  0993  9B80 11BE                LAB    $B1,<BISYNC
002192  0995  B380 09C3                LNJ    $B3,<LOPDS
002193                          *
002194  0997  9B80 11BD                LAB    $B1,<SYNC
002195  0999  B380 09C3                LNJ    $B3,<LOPDS
002196                          *
002197  099B  9B80 11BF                LAB    $B1,<SYN188
002198  099D  B380 09C3                LNJ    $B3,<LOPDS
002199                          *
002200                          *           TEST WIDEBAND 301/303
002201                          *
002202  099F  C870 F01D                LDR    $R4,=Z'F01D'        SET CCP MASK FOR BROADBAND LINES
002203  09A1  CF00 1306                STR    $R4,<LC7SM
002204  09A3  4C04                     LDV    $R4,=4              SET RANGE FOR CHECK ROUTINE
002205  09A4  CF00 116D                STR    $R4,<TP2
002206  09A6  9B80 12A2                LAB    $B1,<C2             CHANGE RANGE IN ZV$C
002207  09A8  9F80 0A08                STB    $B1,<LOPD7+4+4*$AF
002208  09AA  9B80 128A                LAB    $B1,<TLUWB3         SET ADDRESS OF SHOULD BE FOR ZV$C
002209  09AC  9F80 0A05                STB    $B1,<LOPD7+4+$AF
002210  09AE  9B80 11C0                LAB    $B1,<WDB3
002211  09B0  B380 09C3                LNJ    $B3,<LOPDS
002212                          *
002213                          *           TES BI-SYNC. WIDEBAND 31/303
002214                          *
002215  09B2  9B80 11C1                LAB    $B1,<WDB3B
002216  09B4  B380 09C3                LNJ    $B3,<LOPDS
002217                          *
002218                          *           TEST WIDEBAND V35
002219                          *
002220  09B6  9B80 128C                LAB    $B1,<TLUV35         PUT ADDRESS IN FOR ZV$C
```

```
09B8   9F80 0A05             STB    $B1,<LOPD7+4+$AF
09BA   9B80 11C2             LAB    $B1,<WDBV35
09BC   B380 09C3             LNJ    $B3,<LOPDS
                      *
                      *      TEST BI-SYNC. WIDEBAND V35
                      *
09BE   9B80 11C3             LAB    $B1,<WDBV5B
09C0   B380 09C3             LNJ    $B3,<LOPDS
09C2   8382                  JMP    $B2
                      *
                      * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                      *
                      *      LOOP AND TEST DSC INTO DSS LOOP
                      *
                      *
09C3   9F80 09C8      LOPDS  STB    $B1,<LOPD1          STORE ADDRESS OF LINE TYPE
09C5   8753                  CL     =$R3
09C6   D380 0D4F      LOPD   LNJ    $B5,<FLN            FIND LINE ON
09C8   0000           LOPD1  RESV   $AF,0               LINE TYPE ADDRESS
09C9   8383                  JMP    $B3
09CA   C380 0D0A             LNJ    $B4,<GENITZ         GENERAL INITIALIZE
                      *
                      *   WRITE  CHANNEL PROGRAM AND TLU TABLE INTO RAM
                      *
09CC   C570 FF80             AND    $R4,=Z'FF80'
09CE   C470 0040             OR     $R4,=X'40'
09D0   CF00 1151             STR    $R4,<CHNZ
                             CALL   ZV$MLW,CHLP5,WCP5,HX200,CHNZ
09D2   FBC0 0003
09D4   D380 0000    X
09D6   0F80
09D7   0F6A
09D8   1331
09D9   12A6
09DA   1151
                             CALL   ZV$MLW,TLUTB2,WCP5B,HX400,CHNZ
09DB   FBC0 0003
09DD   D380 0000    X
09DF   0F80
09E0   128F
09E1   1332
09E2   12A7
09E3   1151
                      *
09E4   D380 0D3B             LNJ    $B5,<SETLCT         SEND OUT LCT LOCATIONS
09E6   1302                  DC     <LCT7
                      *
09E7   C800 116D             LDR    $R4,<TP2            LOAD $R4 WITH RANGE
09E9   D380 0D99             LNJ    $B5,<MCCB           FORM CCB TO TRANSFER DSS INTO MEMORY
```

```
002256  09E8  1C00                    DC      <DIN
002257  09EC  0060                    DC      X'60'
002258                          *
002259  09ED  C800 113F                LDR     $R4,<FOCC               FORM I/O CONTROL WORD
002260  09EF  C380 0D95                LNJ     $B4,<CGSCH              FOR START I/O
002261  09F1  8000 11C5                IO      <BIT1,=$R4
        09F3  0054
002262  09F4  0703                    BIOT    >LOPD2
002263  09F5  F380 0DBA                LNJ     $B7,<ERMB               ERROR:  I/O WAS NAK'ED
002264  09F7  D380 0D63          LOPD2 LNJ     $B5,<TDLAY              TIME DELAY
002265                          *
002266  09F9  C380 0D25                LNJ     $B4,<INXT               INPUT NEXT STATUS
002267  09FB  82D5                    LB      =$R5,=Z'1000'           CHECK STATUS COMPLETE
        09FC  1000
002268  09FD  0503                    BBT     >LOPD7
002269  09FE  F380 0DBA                LNJ     $B7,<ERMB               ERROR:  STATUS NOT COMPLETE AFTER GNB
002270                          LOPD7 CALL    ZV$C,$,DIN,MASK,$,ERAR
        0A00  FBC0 0003
        0A02  D380 0000       X
        0A04  0F80
        0A05  0A00
        0A06  1C00
        0A07  1160
        0A08  0A00
        0A09  1176
002271  0A0A  8980 1176                CMZ     <ERAR                   CHECK FOR ERROR
002272  0A0C  090D                    BE      >LOPD3
002273  0A0D  D800 1178                LDR     $R5,<ERAR+2             PUT 'is' INTO $R5
002274  0A0F  E800 1177                LDR     $R6,<ERAR+1             PUT 'SHOULD BE' INTO $R6
002275  0A11  F800 1176                LDR     $R7,<ERAR               PUT WORD COUNT INTO $R7
002276  0A13  F380 0DC2                LNJ     $B7,<ERDB               ERROR:  DSC DID NOT LOOP INTO DSS CORRECTLY
002277                                  CALL    ZV$CO                   CONTINUE CHECKING FOR ERROR
        0A15  FBF0 0001
        0A17  D380 0000       X
002278  0A19  3E02              LOPD3 ADV     $R3,=2                  CHANGE TO NEXT LINE
002279  0A1A  0F80 09C6                B       <LOPD
002280                          *
002281                          *
002282                          * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
002283                          *
002284                          *            TRANSMIT MARK AND SPACE TEST
002285                          *
002286  0A1C  C870 4C20          XMMSL LDR     $R4,=A'L '
002287  0A1E  CF00 13A9                STR     $R4,<ERMG+1
002288  0A20  B380 0BA6                LNJ     $B3,<CLFG
002289  0A22  C870 FF07                LDR     $R4,=Z'FF07'            ALTER DATA
002290  0A24  CF00 11E6                STR     $R4,<DOUT+3
002291  0A26  8700 1160                CL      <MASK
002292  0A28  C870 081A                LDR     $R4,=X'81A'             LOAD LCT WITH MARK BIT
```

```
002293  0A2A  8A80 1163              INC     <PARFG
002294  0A2C  EB80 11BC        U2    LAB     $B6,<ASYNC          LOAD $B6 WITH ADDRESS OF TABLE END
002295  0A2E  CF00 1319              STR     $R4,<LCT9MS
002296  0A30  7C06                   LDV     $R7,=6              NUMBER OF 60 MS DELAY
002297                         *
002298                         *           TEST ALL ASYNCHRONOUS LINES
002299                         *
002300  0A31  2C01                   LDV     $R2,=1              LOAD INDEX
002301  0A32  9B80 11BA              LAB     $B1,<ASYN
002302  0A34  B380 0A8C        U3    LNJ     $B3,<DTALOP         LOOP SUB-ROUTINE
002303  0A36  0F85                   B       >U1
002304  0A37  1344                   DC      <WCP9
002305  0A38  1315                   DC      <LCT9
002306  0A39  0005                   DC      5
002307  0A3A  11E3                   DC      <DOUT
002308                         *
002309        0A3B             U1    EQU     $
002310  0A3B  9DD6                   CMB     $B1,=$B6            SEE IF ALL ASYNCHRONOUS  LINES TESTED
002311  0A3C  0903                   BE      >U5
002312  0A3D  9BA1                   LAB     $B1,$B1.$R2
002313  0A3E  0FF6                   B       >U3
002314                         *
002315  0A3F  C870 101A        U5    LDR     $R4,=X'101A'        LOAD LCT FOR SPACE CONDITION
002316  0A41  C900 1319              CMR     $R4,<LCT9MS         CHECK IF ALREADY DONE
002317  0A43  09E9                   BNE     >U2
002318  0A44  8700 1163              CL      <PARFG
002319  0A46  C870 0607              LDR     $R4,=X'607'         PUT DATA BACK TO NORMAL
002320  0A48  CF00 11E6              STR     $R4,<DOUT+3
002321  0A4A  8382                   JMP     $B2
002322                         *
002323                         ******************
002324                         *
002325                         *           DATA CHECK FOR XMIT MARK AND SPACE
002326                         *
002327        0A4B             XMSCK EQU     $
002328  0A4B  8F00 118A              SAVE    <SAV2,=Z'0002'      SAVE $B6
      0A4D  0002
002329  0A4E  DB80 129A              LAB     $B5,<XMSB           LOAD ADDRESS FOR ZV$C
002330  0A50  C800 1319              LDR     $R4,<LCT9MS         CHECK LCT TO SEE IF MARK
002331  0A52  C970 081A              CMR     $R4,=X'81A'         B IF MARK TEST
002332  0A54  0903                   BE      >XMSCK1
002333  0A55  DB80 129E              LAB     $B5,<XSSB           LOAD ADDRESS OF SPACE SHOULD BE
002334  0A57  DF80 0A5E        XMSCK1 STB    $B5,<XMSCK2+4+$AF   STORE SHOULD BE ADDRESS FOR ZV$C
002335                         XMSCK2 CALL   ZV$C,$,DIN,MASK,C4,ERAR
      0A59  FBC0 0003
      0A5B  D380 0000      X
      0A5D  0F80
      0A5E  0A59
      0A5F  1C00
```

```
         0A60  1160
         0A61  12A3
         0A62  1176
002336   0A63  8980 1176              CMZ     <ERAR                 CHECK FOR ERROR
002337   0A65  090D                   BE      >XMSCK3
002338   0A66  D800 1178              LDR     $R5,<ERAR+2           PUT 'is' INTO $R5
002339   0A68  E800 1177              LDR     $R6,<ERAR+1           PUT 'SHOULD BE' INTO $R6
002340   0A6A  F800 1176              LDR     $R7,<ERAR             PUT WORD COUNT INTO $R7
002341   0A6C  F380 0DC2              LNJ     $B7,<ERDB             ERROR:  TRANSMIT MARK OR SPACE ERROR
002342                                CALL    ZV$CO                 CONTINUE CHECKING FOR ERROR
         0A6E  FBF0 0001
         0A70  D380 0000       X
002343   0A72  C870 101A      XMSCK3 LDR     $R4,=X'101A'
002344   0A74  C900 1319              CMR     $R4,<LCT9MS
002345   0A76  0991                   BNE     >XMSCK5
002346   0A77  E380 0BB7              LNJ     $B6,<BASE             LOAD $R5 WITH LCT BASE
002347   0A79  5E3C                   ADV     $R5,=60
002348   0A7A  DF00 0A7E              STR     $R5,<XMSCK4           STORE ADDRESS TO READ LCT
002349                              *
002350   0A7C  E380 0D75              LNJ     $B6,<RDLCT
002351   0A7E  0000            XMSCK4 DC      0                     ADDRESS
002352   0A7F  0002                   DC      2
002353                              *
002354   0A80  D870 FEEE              LDR     $R5,=Z'FEEE'          LOAD $R5 WITH SHOULD BE
002355   0A82  D900 1173              CMR     $R5,<LCTV
002356   0A84  0903                   BE      >XMSCK5
002357   0A85  F380 0DC2              LNJ     $B7,<ERDB             ERROR:  FRAMING ERROR FAILURE
002358   0A87  8F80 118A      XMSCK5 RSTR    <SAV2,=Z'0002'        RESTORE $B6
         0A89  0002
002359   0A8A  0F80 0BA0              B       <JP5A
002360                              *
002361                              *
002362                              *=============================================================================
002363   0A8C  8F00 119A      DTALOP SAVE    <SAV3,=Z'0112'        SAVE $B3,$B6
         0A8E  0112
002364   0A8F  9F80 0A94              STB     $B1,<TG1              FOR FIND LINE ON
002365   0A91  8753                   CL      =$R3                  CLEAR $R3
002366   0A92  D380 0D4F      NXLT    LNJ     $B5,<FLN              FIND LINE ON
002367   0A94  0000            TG1     RESV    $AF,0
002368   0A95  8383                   JMP     $B3                   NO LINE ON RETURN
002369                              *
002370                                CALL    ZV$F,DIN,DFLT,HX200   FILL INPUT BUFFER
         0A96  FBC0 0003
         0A98  D380 0000       X
         0A9A  0F80
         0A9B  1C00
         0A9C  1222
         0A9D  12A6
002371   0A9E  C808 0A94              LDR     $R4,*<TG1
```

```
002372  0AA0  C900 11BC                CMR   $R4,<ASYNC
002373  0AA2  0983                     BNE   >DTAL                 B IF NOT CURRENT LOOP ADAPTER
002374  0AA3  C380 0C18                LNJ   $B4,<SCLS             SET  CURRENT LOOP ADAPTER SPEED
002375  0AA5  BBC3 0001          DTAL  LAB   $B3,$B3.1             DUMMY INCREMENT OF $B3
002376  0AA7  C380 0D0A                LNJ   $B4,<GENITZ           GENERAL INITIALIZE
002377  0AA9  DCF3                     LDB   $B5,+$B3              GET ADDRESS TO STORE FOR ZV$MLW CHANNEL
002378  0AAA  8980 1152                CMZ   <COUNT                IF CABLE LOOP DON'T STMD
002379  0AAC  0986                     BNE   >DT1
002380  0AAD  82D4                     LB    =$R4,=X'80'           SEE IF SECOND LINE OF CLA PAIR
        0AAE  0080
002381  0AAF  0583                     BBF   >DT1
002382  0AB0  C380 0C11                LNJ   $B4,<STMD             SET TEST MODE ON FIRST LINE OF CLA PAIR
002383  0AB2  C570 FF80          DT1   AND   $R4,=Z'FF80'          SET CHANNEL FOR ZV$MLW
002384  0AB4  C470 0040                OR    $R4,=X'40'            OR IN DIRECTION BIT
002385  0AB6  CF00 1151                STR   $R4,<CHNZ             STORE CHANNEL FOR ZV$LIB CALL
002386  0AB8  CCF5                     LDB   $B4,+$B5              LOAD AND STORE ADDRESS FOR
002387  0AB9  CF80 0ACD                STB   $B4,<LPWRT+4+$AF      ZV$LIB CALL
002388  0ABB  CCF5                     LDB   $B4,+$B5
002389  0ABC  CF80 0AD6                STB   $B4,<LPWRT1+4+$AF
002390  0ABE  DF80 0ACE                STB   $B5,<LPWRT+4+2*$AF
002391  0AC0  DBC5 0001                LAB   $B5,$B5.1             INCREMENT $B5
002392  0AC2  DF80 0AD7                STB   $B5,<LPWRT1+4+2*$AF
002393  0AC4  DBC5 0001                LAB   $B5,$B5.1
002394  0AC6  DF80 0AD8                STB   $B5,<LPWRT1+4+3*$AF
002395                           *
002396                           LPWRT CALL  ZV$MLW,$,$,HX200,CHNZ
        0AC8  FBC0 0003
        0ACA  D380 0000      X
        0ACC  0F80
        0ACD  0AC8
        0ACE  0AC8
        0ACF  12A6
        0AD0  1151
002397                           LPWRT1 CALL ZV$MLW,$,$,$,CHNZ
        0AD1  FBC0 0003
        0AD3  D380 0000      X
        0AD5  0F80
        0AD6  0AD1
        0AD7  0AD1
        0AD8  0AD1
        0AD9  1151
002398                           *
002399  0ADA  DCF3                     LDB   $B5,+$B3              STORE ADDRESS FOR LCT TABLE
002400  0ADB  DF80 0ADF                STB   $B5,<TG2
002401                           *
002402  0ADD  D380 0D3B                LNJ   $B5,<SETLCT           SEND OUT LCT
002403  0ADF  0000               TG2   RESV  $AF,0
002404                           *
002405  0AE0  C800 113F                LDR   $R4,<F0CC             CCB RESET FOR REC
```

```
002406  0AE2  C380 0D95            LNJ   $B4,<CGSCH
002407  0AE4  8000 11C9            IO    <BIT7,=$R4
        0AE6  0054
002408  0AE7  0703                 BIOT  >DT2
002409                        *
002410  0AE8  F380 0DBA            LNJ   $B7,<ERMB          ERROR:  I/O WAS NAK'ED
002411                        *
002412  0AEA  8AD3                 INC   =$R3
002413  0AEB  C800 113F            LDR   $R4,<FOCC          CCB RESET FOR TRANSMIT
002414  0AED  C380 0D95            LNJ   $B4,<CGSCH
002415  0AEF  8000 11C9            IO    <BIT7,=$R4
        0AF1  0054
002416  0AF2  0703                 BIOT  >DT3
002417                        *
002418  0AF3  F380 0DBA            LNJ   $B7,<ERMB          ERROR:  I/O WAS NAK'ED
002419                        *
002420  0AF5  C803                 LDR   $R4,$B3            LOAD RANGE
002421  0AF6  CF00 1166            STR   $R4,<RANGE         STORE RANGE FOR ZV$C
002422  0AF8  4001                 SOL   $R4,1              CONVERT TO BYTES FOR CCB RANGE
002423                        *
002424  0AF9  BBC3 0001            LAB   $B3,$B3.1
002425  0AFB  DCF3                 LDB   $B5,+$B3           STORE ADDRESS FOR TRANSMIT BUFFER
002426  0AFC  DF80 0B02            STB   $B5,<TG3           FOR CCB FORMATION
002427  0AFE  DF80 0B58            STB   $B5,<JP4+4+$AF     FOR DATA CHECK CALL(ZV$C)
002428                        *
002429  0B00  D380 0D99            LNJ   $B5,<MCCB          FORM CCB FOR TRANSMIT
002430  0B02  0000                 RESV  $AF,0
002431  0B03  0060                 DC    X'60'
002432                        *
002433  0B04  88D3                 DEC   =$R3               GET RECV. CHANNEL
002434  0B05  D380 0D99            LNJ   $B5,<MCCB          FORM CCB FOR RECIEVE SIDE
002435  0B07  1C00                 DC    <DIN
002436  0B08  0060                 DC    X'60'
002437                        *
002438  0B09  5C01                 LDV   $R5,=1             START I/O ON XMIT SIDE TO SET
002439  0B0A  8AD3                 INC   =$R3               CONFIG. AND XMIT FILL CHAR. (SYNC. ONLY)
002440  0B0B  FE55                 SWR   $R7,=$R5
002441  0B0C  C800 113F            LDR   $R4,<FOCC
002442  0B0E  C380 0D95            LNJ   $B4,<CGSCH
002443  0B10  8000 11C5            IO    <BIT1,=$R4
        0B12  0054
002444  0B13  0703                 BIOT  >MELCO
002445  0B14  F380 0DBA            LNJ   $B7,<ERMB
002446  0B16  D380 0D63            MELCO LNJ   $B5,<TDLAY
002447  0B18  FE55                 SWR   $R7,=$R5
002448  0B19  88D3                 DEC   =$R3
002449                        *
002450  0B1A  C800 113F            LDR   $R4,<FOCC
002451  0B1C  C380 0D95            LNJ   $B4,<CGSCH         FORM I/O FOR START I/O
```

```
002452  0B1E  8000 11C5              IO      <BIT1,=$R4
        0B20  0054
002453  0B21  0703                   BIOT    >JP2
002454  0B22  F380 0DBA              LNJ     $B7,<ERMB
002455  0B24  D380 0D63        JP2   LNJ     $B5,<TDLAY          TIME DELAY
002456  0B26  8980 1170              CMZ     <STPFG              CHECK STOP I/O TEST FLAG
002457  0B28  0903                   BE      >JP2A
002458  0B29  0F80 0800              B       <STIQ               BRANCH TO STOP I/O AND CHANNEL RESET TEST
002459  0B2B  8980 1163        JP2A  CMZ     <PARFG
002460  0B2D  0908                   BE      >JP2B
002461  0B2E  8980 1152              CMZ     <COUNT
002462  0B30  0900 077B              BE      <PAR1               BRANCH TO CHECK THAT PARITY ERROR SET
002463  0B32  8AD3                   INC     =$R3
002464  0B33  0F80 0A4B              B       <XMSCK              B TO MARK AND SPACE CHECK ROUTINE
002465  0B35  C380 0D25        JP2B  LNJ     $B4,<INXT           INPUT NEXT STATUS ON REC. SIDE
002466  0B37  E870 1000              LDR     $R6,=Z'1000'
002467  0B39  82D5                   LB      =$R5,=Z'1000'       SEE IF STATUS COMPLETE
        0B3A  1000
002468  0B3B  0503                   BBT     >JP3
002469  0B3C  F380 0DC2              LNJ     $B7,<ERDB           STATUS NOT COMPLETE ON REC.
002470                         *
002471  0B3E  8AD3             JP3   INC     =$R3
002472  0B3F  C380 0D25              LNJ     $B4,<INXT           INPUT NEXT STATUS ON XMIT SIDE
002473  0B41  82D5                   LB      =$R5,=Z'1000'       SEE IF STATUS COMPLETE
        0B42  1000
002474  0B43  0503                   BBT     >JP4A
002475  0B44  F380 0DC2              LNJ     $B7,<ERDB           STATUS INCOMPLETE ON XMIT SIDE
002476  0B46  8980 116A        JP4A  CMZ     <RUNFG
002477  0B48  0980 0915              BNE     <RUNCK              BRANCH TO OVERRUN AND UNDERRUN TEST
002478                         *
002479  0B4A  8980 1156              CMZ     <CRCFG              TEST CRC AND PARITY FLAG
002480  0B4C  0903                   BE      >JP4C               FLAG NOT SET JUST CHECK DATA
002481  0B4D  D380 0570              LNJ     $B5,<CRPRT          GO CHECK CRC RESIDUE AND PARITY
002482                         *
002483  0B4F  8980 114D        JP4C  CMZ     <ARFG               CHECK ADAPTER READY FLAG
002484  0B51  0980 04CF              BNE     <ARCK               GO TO ADAPTER READY CHECK ROUTINE
002485                         *
002486                         *               CHECK IF DATA LOOP CORRECTLY
002487                         *
002488                         JP4   CALL    ZV$C,$,DIN,MASK,RANGE,ERAR
        0B53  FBC0 0003
        0B55  D380 0000      X
        0B57  0F80
        0B58  0B53
        0B59  1C00
        0B5A  1160
        0B5B  1166
        0B5C  1176
002489                         *##
```

```
002490                                  *#
002491  0B5D  8A80 114A                         INC     <MLXCT                  INC MODEM LOOPBACK XMIT CTR
002492                                  *#
002493                                  *##
002494  0B5F  8980 1176                         CMZ     <ERAR                   CHECK FOR ERROR
002495  0B61  0913                              BE      >JP5
002496                                  *##
002497                                  *#
002498  0B62  8A80 114B                         INC     <MLECT                  INC MODEM LOOPBACK ERROR CTR
002499  0B64  D800 1149                         LDR     $R5,<TEST
002500  0B66  5D4D                              CMV     $R5,X'4D'
002501  0B67  090D                              BE      >JP5                    SUPRESS ERROR MSG IF MODEM LOOPBACK
002502                                  *#
002503                                  *##
002504  0B68  D800 1178                         LDR     $R5,<ERAR+2             PUT IS IN $R5
002505  0B6A  E800 1177                         LDR     $R6,<ERAR+1             PUT SHOULD BE INTO $R6
002506  0B6C  F800 1176                         LDR     $R7,<ERAR               PUT WORD COUNT INTO $R7
002507  0B6E  F380 0DC2                         LNJ     $B7,<ERDB
002508                                          CALL    ZV$CO                   CONTINUE LOOKING FOR ERRORS
        0B70  FBF0 0001
        0B72  D380 0000         X
002509  0B74  8980 1150                 JP5     CMZ     <CHSZFG                 CHECK CHARACTER SIZE TEST FLAG
002510  0B76  092A                              BE      >JP5A                   IF SET CHECK HIGH ORDER DIGITS FOR ZERO'S
002511  0B77  5C08                              LDV     $R5,=8
002512  0B78  D900 1153                         CMR     $R5,<CS
002513  0B7A  0926                              BE      >JP5A
002514  0B7B  86C0 1160                         CPL     <MASK                   COMPLEMENT MASK TO CHECK OTHER BITS
002515                                          CALL    ZV$F,DIN1,CPFLG+3,RANGE CLEAR SHOULD BE TO ZERO'S
        0B7D  FBC0 0003
        0B7F  D380 0000         X
        0B81  0F80
        0B82  1D00
        0B83  11DF
        0B84  1166
002516                                          CALL    ZV$C,DIN1,DIN,MASK,RANGE,ERAR
        0B85  FBC0 0003
        0B87  D380 0000         X
        0B89  0F80
        0B8A  1D00
        0B8B  1C00
        0B8C  1160
        0B8D  1166
        0B8E  1176
002517  0B8F  8980 1176                         CMZ     <ERAR                   CHECK FOR ERRORS
002518  0B91  090D                              BE      >JP5B
002519  0B92  D800 1178                         LDR     $R5,<ERAR+2             PUT IS INTO $R5
002520  0B94  E800 1177                         LDR     $R6,<ERAR+1             PUT SHOULD BE INTO $R6
002521  0B96  F800 1176                         LDR     $R7,<ERAR               PUT WORD COUNT INTO $R7
002522  0B98  F380 0DC2                         LNJ     $B7,<ERDB               ERROR:  SOME HIGH ORDER BITS SET IN CHARACTER
```

```
002523                                      CALL   ZV$CO              CONTINUE LOOKING FOR ERRORS
        0B9A  FBF0 0001
        0B9C  D380 0000      X
002524  0B9E  8600 1160              JP5B  CPL    <MASK               RESET MASK
002525                               *
002526  0BA0  8AD3                   JP5A  INC    =$R3                FIND NEXT LINE ON OF THIS TYPE
002527  0BA1  8F80 119A                    RSTR   <SAV3,=Z'0112'      RESTORE $B3,$B6
        0BA3  0112
002528  0BA4  0F80 0A92                    B      <NXLT
002529                               *
002530                               *
002531                               *=============================================================================
002532                               *
002533                               *           CLEAR TEST FLAGS
002534                               *
002535  0BA6  8700 1156              CLFG  CL     <CRCFG
002536  0BA8  8700 1163                    CL     <PARFG
002537  0BAA  8700 116E                    CL     <SBTFG
002538  0BAC  8700 1170                    CL     <STPFG
002539  0BAE  8700 1171                    CL     <WDBFG
002540  0BB0  8700 114D                    CL     <ARFG
002541  0BB2  8700 1150                    CL     <CHSZFG
002542  0BB4  8700 116A                    CL     <RUNFG
002543  0BB6  8383                         JMP    $B3
002544                               *
002545                               *=============================================================================
002546                               *
002547                               *           LOAD $R5 WITH LCT BASE ADDRESS
002548                               *
002549  0BB7  D853                   BASE  LDR    $R5,=$R3
002550  0BB8  5041                         SOR    $R5,1
002551  0BB9  5F40                         MLV    $R5,=X'40'
002552  0BBA  8386                         JMP    $B6
002553                               *
002554                               *=============================================================================
002555                               *
002556                               *                CRC GENERATOR
002557                               *
002558                               *           OUTPUT - TAKES TABLE OF DATA AND FORMS ACCUMULATED CRC.
002559                               *                       ONLY FOR CRC16 AND CCITT.
002560                               *
002561                               *      TO USE, MUST SET:
002562                               *           R6 - NUMBER OF BITS IN CHAR.
002563                               *           R7 - RANGE IN BYTES
002564                               *           B3 - TABLE ADDRESS
002565                               *           B6 - ADDRESS OF CRC TYPE.
002566                               *
002567                               *
002568                               *         LNJ    $B4,<CRC
```

```
002569                                  *
002570                                  *
002571                                  *       $B3 - ADDRESS WHERE CRC ACCUMULATOR IS STORED
002572                                  *       $B6 - ADDRESS OF CRC TYPE
002573                                  *
002574                                  *       $R6 - CONTAINS NUMBER OF BITS IN CHARACTER
002575                                  *       $R7 - THE NUMBER OF CRC TO BE
002576                                  *             GENERATED FROM THE TABLE AT $B3
002577                                  *
002578                                  *         THE ACCUMULATED CRC IS STORED IN LOCATION CRCAC.
002579                                  *
002580  0BBB  8F00 117A                 CRC     SAVE    <SAV1,=Z'F994'
       0BBD  F994
002581  0BBE  C876                              LDR     $R4,+$B6                PUT CRC TYPE INTO $R4
002582  0BBF  8751                              CL      =$R1
002583  0BC0  8753                              CL      =$R3                    CLEAR $R3 FOR LONG SHIFT
002584  0BC1  8752                              CL      =$R2
002585  0BC2  8256                              NEG     =$R6
002586  0BC3  EF00 116B                         STR     $R6,<TLOC               STORE COUNTER
002587  0BC5  D2FF                              LLH     $R5,$B3.+$R3
002588  0BC6  7001                              SOL     $R7,1
002589  0BC7  8257                              NEG     =$R7
002590  0BC8  DF51                      CRCA    STR     $R5,=$R1
002591  0BC9  5041                              SOR     $R5,1
002592  0BCA  9652                              XOR     $R1,=$R2
002593  0BCB  2041                              SOR     $R2,1
002594  0BCC  1B02                              BEVN    $R1,>CRCB
002595  0BCD  A654                              XOR     $R2,=$R4
002596  0BCE  67FA                      CRCB    BINC    $R6,>CRCA
002597  0BCF  D0FF                              LDH     $R5,$B3.+$R3
002598  0BD0  E800 116B                         LDR     $R6,<TLOC               RESET CHAR. COUNT
002599  0BD2  77F6                              BINC    $R7,>CRCA               DO NEW CHARACTER
002600  0BD3  AF00 1155                         STR     $R2,<CRCAC              STORE LAST CRC ACCUMULATED
002601  0BD5  8F80 117A                         RSTR    <SAV1,=Z'F994'          RESTORE REGS.
       0BD7  F994
002602  0BD8  8384                              JMP     $B4
002603                                  *
002604                                  *=========================================================================
002605                                  *
002606                                  *           PARITY CALCULATOR
002607                                  *
002608                                  *                       TO USE, MUST SET:
002609                                  *                          $R7 - RANGE  OF DATA IN WORDS
002610                                  *                          $B1 - ADDRESS OF DATA AFTER PARITY HAS BEEN ADDED
002611                                  *                          $B3 - ADDRESS OF DATA TO ADD PARITY ON
002612                                  *                          PTTY - 1MEANS ODD PARITY
002613                                  *                                 0 MEANS EVEN PARITY
002614                                  *
002615  0BD9  8F00 117A                 CPAR    SAVE    <SAV1,=Z'EFD0'          SAVE REGS.
```

```
        0BDB  EFD0
002616  0BDC  9800 1153                LDR     $R1,<CS             LDR $R1 WITH NUMBER TO SHIFT
002617  0BDE  88D1                     DEC     =$R1
002618  0BDF  8755                     CL      =$R5
002619  0BE0  8752                     CL      =$R2                CLEAR INDEX
002620  0BE1  8257                     NEG     =$R7                NEGATE RANGE FOR BRANCH AND INCREMENT
002621  0BE2  C2A3              CPAR2  LLH     $R4,$B3.$R2         LOAD $R4 WITH HALF WORD OF DATA BUFFER
002622  0BE3  8980 1150                CMZ     <CHSZFG             SEE IF CHAR. SIZE TEST
002623  0BE5  090F                     BE      >CPAR3
002624  0BE6  6C08                     LDV     $R6,=8              IF 8 BIT CHAR. DO NOT STRIP BITS
002625  0BE7  E900 1153                CMR     $R6,<CS
002626  0BE9  090B                     BE      >CPAR3
002627  0BEA  88D6                     DEC     =$R6                IF 7 OR 6 BIT CHAR. STRIP FOR PARITY TO BE SE
002628  0BEB  E900 1153                CMR     $R6,<CS
002629  0BED  0904                     BE      >CPAR4
002630  0BEE  E870 001F                LDR     $R6,=X'1F'          MASK TO STRIP WITH 6 BIT CHAR.
002631  0BF0  0F83                     B       >CPAR5
002632  0BF1  E870 003F         CPAR4  LDR     $R6,=X'3F'          MASK TO STRIP WITH 7 BIT CHAR.
002633  0BF3  C556              CPAR5  AND     $R4,=$R6            STRIP BIT TO CHAR SIZE MINUS PARITY BIT
002634  0BF4  CF56              CPAR3  STR     $R4,=$R6            SAVE CHAR IN $R6
002635                                 *
002636                                 *           CALCULATE PARITY FOR THE CHAR.
002637                                 *
002638  0BF5  50C4                     DOR     $R5,4
002639  0BF6  CF00 116B                STR     $R4,<TLOC
002640  0BF8  8754                     CL      =$R4
002641  0BF9  5084                     DOL     $R5,4
002642  0BFA  C600 116B                XOR     $R4,<TLOC
002643  0BFC  50C2                     DOR     $R5,2
002644  0BFD  CF00 116B                STR     $R4,<TLOC
002645  0BFF  8754                     CL      =$R4
002646  0C00  5082                     DOL     $R5,2
002647  0C01  C600 116B                XOR     $R4,<TLOC
002648  0C03  4041                     SOR     $R4,1
002649  0C04  0682                     BCF     >CPAR1
002650  0C05  8AD4                     INC     =$R4
002651  0C06  C600 1164         CPAR1  XOR     $R4,<PTTY
002652                                 *
002653  0C08  4000                     SOL     $R4,0               SHIFT PARITY BIT TO POSITION
002654  0C09  E454                     OR      $R6,=$R4            PUT PARITY BIT INTO DATA
002655  0C0A  E7ED                     STH     $R6,$B1.+$R2        STORE INTO DATA SHOULD BE
002656  0C0B  7780 0BE2                BINC    $R7,<CPAR2          DO NEXT CHARACTER
002657  0C0D  8F80 117A                RSTR    <SAV1,=Z'EFD0'      RESTORE REG.
        0C0F  EFD0
002658  0C10  8384                     JMP     $B4                 JUMP OUT OF PARITY FORMATION SUB-ROUTINE
002659                                 *
002660                                 *=============================================================================
002661                                 *
002662                                 *           SET TEST MODE ON FIRST LINE OF CLA
```

```
002663                                  *
002664                                  *                       LNJ $B4,<STMD
002665                                  *
002666  0C11  8F00 118A                STMD    SAVE    <SAV2,=Z'DF8C'          SAVE REGS.
        0C13  DF8C
002667  0C14  3EFE                            ADV     $R3,=-2                 SET $R3 TO RECV. CHAN. ON FIRST LINE
002668  0C15  CB80 0E11                        LAB     $B4,<CHLP               LOAD $B4 WITH ADDRESS OF CHANNEL PROG.
002669  0C17  0F8E                            B       >STMD2
002670                                  *=======================================================================
002671                                  *
002672                                  *           SET CURRENT LOOP ADAPTER FOR FASTEST SPEED
002673                                  *
002674  0C18  8F00 118A                SCLS    SAVE    <SAV2,=Z'DF8C'          SAVE REGS.
        0C1A  DF8C
002675  0C1B  C800 113F                        LDR     $R4,<FOCC
002676  0C1D  C380 0D95                        LNJ     $B4,<CGSCH
002677  0C1F  8000 11C4                SCLS1   IO      <BIT0,=$R4              CHANNEL INITIALIZE
        0C21  0054
002678  0C22  07FD                            BIOF    >SCLS1
002679  0C23  CB80 0E14                        LAB     $B4,<CHLPS              LOAD $B4 WITH ADDRESS OF CHANNEL PROG.
002680  0C25  CF80 0C34                STMD2   STB     $B4,<STMD3+4+$AF        STORE ADDRESS FOR ZV$MLW
002681  0C27  C800 1145                        LDR     $R4,<FINS               LOAD AN I/O TO GET CHANNEL FOR ZV$MLW
002682  0C29  C380 0D95                        LNJ     $B4,<CGSCH
002683  0C2B  C570 FFC0                        AND     $R4,=Z'FFC0'            STRIP OFF DIRECTION BIT AND FUN CODE
002684  0C2D  CF00 1151                        STR     $R4,<CHNZ               STORE CHANNEL FOR CALL
002685                                  *
002686                                  STMD3   CALL    ZV$MLW,$,WCP,HX200,CHNZ
        0C2F  FBC0 0003
        0C31  D380 0000         X
        0C33  0F80
        0C34  0C2F
        0C35  131C
        0C36  12A6
        0C37  1151
002687  0C38  D380 0D3B                        LNJ     $B5,<SETLCT             SEND OUT LCT
002688  0C3A  12CC                            DC      <LCT
002689                                  *
002690  0C3B  8757                            CL      =$R7                    CLEAR FOR SHORT DELAY OF 30 MS.
002691  0C3C  C800 113F                        LDR     $R4,<FOCC               FORM CONTROL WORD
002692  0C3E  C380 0D95                        LNJ     $B4,<CGSCH              FOR
002693  0C40  8000 11C5                        IO      <BIT1,=$R4              START I/O
        0C42  0054
002694  0C43  0700 0C47                        BIOT    <STMD1
002695  0C45  F380 0DBA                        LNJ     $B7,<ERMB               ERROR:  I/O WAS NAK'ED
002696                                  *
002697  0C47  D380 0D63                STMD1   LNJ     $B5,<TDLAY              TIME DELAY
002698  0C49  8F80 118A                        RSTR    <SAV2,=Z'DF8C'          RESTORE REGS.
        0C4B  DF8C
002699  0C4C  8384                            JMP     $B4
```

```
002700                                  *
002701                                  *=============================================================================
002702                                  *
002703                                  *          CONVERT RTC TICKS TO MILSEC.
002704                                  *
002705  0C4D  8F00 117A                 TKSEC   SAVE    <SAV1,=Z'D380'          SAVE REGS.
        0C4F  D380
002706                                  *
002707  0C50  CF51                              STR     $R4,=$R1                DUPLICATE VALUE IN $R1
002708  0C51  CB00 1157                         MUL     $R4,<DIV1
002709  0C53  CF00 116B                         STR     $R4,<TLOC
002710  0C55  C851                              LDR     $R4,=$R1
002711  0C56  CB00 1158                         MUL     $R4,<DIV2
002712  0C58  C370 000A                         DIV     $R4,=10
002713  0C5A  CA00 116B                         ADD     $R4,<TLOC
002714  0C5C  CF00 116B                         STR     $R4,<TLOC
002715  0C5E  C851                              LDR     $R4,=$R1
002716  0C5F  CB00 1159                         MUL     $R4,<DIV3
002717  0C61  C370 0064                         DIV     $R4,=100
002718  0C63  CA00 116B                         ADD     $R4,<TLOC
002719  0C65  CF00 116B                         STR     $R4,<TLOC
002720  0C67  C851                              LDR     $R4,=$R1
002721  0C68  CB00 115A                         MUL     $R4,<DIV4
002722  0C6A  C370 03E8                         DIV     $R4,=1000
002723  0C6C  CA00 116B                         ADD     $R4,<TLOC
002724                                  *
002725  0C6E  8F80 117A                         RSTR    <SAV1,=Z'D380'
        0C70  D380
002726  0C71  8385                              JMP     $B5
002727                                  *
002728                                  *=============================================================================
002729                                  *
002730                                  *          CONVERT MS TO BITS/SEC (GIVEN CHAR. SIZE & NUMBER)
002731                                  *
002732  0C72  8F00 117A                 BPS     SAVE    <SAV1,=Z'DF80'          SAVE REGS.
        0C74  DF80
002733  0C75  8756                              CL      =$R6                    CLEAR $R6 FOR DIVIDE
002734  0C76  9826                              LDR     $R1,$B6.$R2             LOAD $R1 WITH NUMBER OF CHAR.
002735  0C77  9970 0F00                         CMR     $R1,=3840
002736  0C79  0983                              BNE     >BPS1
002737  0C7A  1E07                              ADV     $R1,=7
002738  0C7B  9255                              SUB     $R1,=$R5
002739  0C7C  F870 03E8                 BPS1    LDR     $R7,=1000               CONVERSION FACTOR FOR MS TOC.
002740  0C7E  FB00 1154                         MUL     $R7,<CSB                MULTIPLY BY CHARACTER SIZE
002741  0C80  F354                              DIV     $R7,=$R4                DIVIDE BY ACTUAL TIME IN MS
002742  0C81  8D00 116C                         SDI     <TP1                    STORE DOUBLE INTEGER
002743  0C83  8756                              CL      =$R6
002744  0C84  F800 116D                         LDR     $R7,<TP2
002745  0C86  FB51                              MUL     $R7,=$R1
```

```
002746  0C87  82D7                    LB      =$R7,=Z'8000'
        0C88  8000
002747  0C89  0502                    BBT     >BPS2A
002748  0C8A  69C3                    BEZ     $R6,>BPS2
002749  0C8B  8A80 115E        BPS2A   INC     <HGSPD
002750  0C8D  8D00 1161        BPS2    SDI     <MSBS
002751  0C8F  8D30 1227                SDI     <SYLSP0.$R3
002752  0C91  F800 116C                LDR     $R7,<TP1
002753  0C93  7F0A                    MLV     $R7,=10
002754  0C94  F354                    DIV     $R7,=$R4
002755  0C95  FF55                    STR     $R7,=$R5
002756  0C96  DB51                    MUL     $R5,=$R1
002757  0C97  D370 000A                DIV     $R5,=10
002758  0C99  DA00 1162                ADD     $R5,<MSBS+1
002759  0C9B  70D0                    DOR     $R7,16
002760  0C9C  7F0A                    MLV     $R7,=10
002761  0C9D  F354                    DIV     $R7,=$R4
002762  0C9E  FB51                    MUL     $R7,=$R1
002763  0C9F  F370 0064                DIV     $R7,=100
002764  0CA1  FA55                    ADD     $R7,=$R5
002765  0CA2  FF00 1162                STR     $R7,<MSBS+1
002766  0CA4  8F80 117A                RSTR    <SAV1,=Z'DF80'
        0CA6  DF80
002767  0CA7  8385                    JMP     $B5
002768                          *
002769                          *=====================================================================
002770                          *
002771                          *           PRINT LINE SPEED
002772                          *
002773  0CA8  8F00 118A        PRTLSP  SAVE    <SAV2,=Z'B88F'          SAVE REGS.
        0CAA  B88F
002774  0CAB  3041                    SOR     $R3,1                   SHIFT $R3 TO REPRESENT LINE NUMBER
002775  0CAC  BF00 116B                STR     $R3,<TLOC               STORE LINE NUMBER FOR ZV$TD
002776                                 CALL    ZV$TC,MESG2,BYTP        PRINT LINE TYPE
        0CAE  FBC0 0003
        0CB0  D380 0000        X
        0CB2  0F80
        0CB3  135B
        0CB4  114E
002777                                 CALL    ZV$TD,TLOC              PRINT LINE NUMBER
        0CB5  FBC0 0003
        0CB7  D380 0000        X
        0CB9  0F80
        0CBA  116B
002778                                 CALL    ZV$T,MESG4              PRINT 'speed'
        0CBB  FBC0 0003
        0CBD  D380 0000        X
        0CBF  0F80
        0CC0  1364
```

```
002779  0CC1  D380 0C4D            LNJ   $B5,<TKSEC          CONVERT NUMBER OF TICK IN $R4 TO MS
002780                        *
002781  0CC3  D380 0C72            LNJ   $B5,<BPS            USE $R4 TO CALCULATE BITS/SEC
002782  0CC5  8980 115E            CMZ   <HGSPD              CHECK FOR HIGH SPEED LINE
002783  0CC7  0919                 BE    >PRLT3
002784  0CC8  8C80 1161            LDI   <MSBS               LOAD DOUBLE INTERGER INTO $R6,$R7
002785  0CCA  F370 03E8            DIV   $R7,=1000           FIND NUMBER OF KB
002786  0CCC  FF00 1161            STR   $R7,<MSBS
002787  0CCE  6D64                 CMV   $R6,=100
002788  0CCF  0A02                 BAG   >PRLT5
002789  0CD0  6C64                 LDV   $R6,=100
002790  0CD1  EF00 1162      PRLT5 STR   $R6,<MSBS+1
002791                      CALL  ZV$TD,MSBS,BKLS     PRINT NUMBER OF KB
        0CD3  FBC0 0003
        0CD5  D380 0000     X
        0CD7  0F80
        0CD8  1161
        0CD9  11E2
002792                      CALL  ZV$T,COMMA
        0CDA  FBC0 0003
        0CDC  D380 0000     X
        0CDE  0F80
        0CDF  11E1
002793                      PRLT3 CALL  ZV$TD,MSBS+1,BKLS     PRINT LINE SPEED
        0CE0  FBC0 0003
        0CE2  D380 0000     X
        0CE4  0F80
        0CE5  1162
        0CE6  11E2
002794  0CE7  8980 12E5            CMZ   <LC3SYF             IF SYNC. LINE ALL DONE
002795  0CE9  0999                 BNE   >PRLT1              IF ASYNC. PRINT SHOULD BE
002796  0CEA  EB80 1257            LAB   $B6,<ALSPBL         LOAD ADDRESS OF 'should be' SPEED TABLE
002797  0CEC  C870 2108            LDR   $R4,=X'2108'        IF NEW LINE SPEED CHANGE $B6 TO POINT TO TABL
002798  0CEE  C908 03D6            CMR   $R4,*<ST1
002799  0CF0  0903                 BE    >PRLT7
002800  0CF1  EB80 1266            LAB   $B6,<CMSPBL
002801                      PRLT7 CALL  ZV$T,MESG5          PRINT 'SHOULD BE'
        0CF3  FBC0 0003
        0CF5  D380 0000     X
        0CF7  0F80
        0CF8  136A
002802  0CF9  B826                 LDR   $R3,$B6.$R2         LOAD $R3 WITH LINE SPEED
002803  0CFA  BF00 116B            STR   $R3,<TLOC
002804                      CALL  ZV$TD,TLOC
        0CFC  FBC0 0003
        0CFE  D380 0000     X
        0D00  0F80
        0D01  116B
002805  0D02  8700 1161      PRLT1 CL    <MSBS
```

```
002806  0D04  8700 1162              CL      <MSBS+1
002807  0D06  8F80 118A              RSTR    <SAV2,=Z'B88F'        RESTORE R
        0D08  B88F
002808  0D09  8384                   JMP     $B4
002809                          *
002810                          *=====================================================================
002811                          *
002812                          * GENERAL INITIALIZE FOR MLCC
002813                          *
002814  0D0A  8F00 118A         GENITZ SAVE   <SAV2,=Z'1118'        SAVE B7,$R7
        0D0C  1118
002815  0D0D  8757                   CL      =$R7                  ALLOW 30 MS DELAY
002816  0D0E  C800 113D              LDR     $R4,<FOMC             GET FUNCTION CODE FOR INITIALIZE
002817  0D10  C380 0D95              LNJ     $B4,<CGSCH            MODIFY FOR CHANNEL
002818  0D12  8000 11C4              IO      <BITO,=$R4            INITIALIZE
        0D14  0054
002819  0D15  0703                   BIOT    >N2
002820  0D16  F380 0DBA              LNJ     $B7,<ERMB             INITIALIZE WAS NAK'ED
002821  0D18  C380 0D63         N2   LNJ     $B5,<TDLAY            WAIT 30 MS
002822  0D1A  8F80 118A              RSTR    <SAV2,=Z'1118'
        0D1C  1118
002823  0D1D  8384                   JMP     $B4                   RETURN
002824                          *
002825                          *=====================================================================
002826                          *
002827                          *            OUTPUT CHANNEL CONTROL
002828                          *
002829                          *               $R2 - CONTAINS CHANNEL CONTROL TYPE
002830                          *
002831  0D1E  8F00 117A         STCR   SAVE   <SAV1,=Z'0008'        SAVE REG.
        0D20  0008
002832  0D21  D852                   LDR     $R5,=$R2
002833  0D22  C800 113F              LDR     $R4,<FOCC             LOAD FUNCTION CODE FOR CHANNEL CONTROL
002834  0D24  0F8C                   B       >N1A
002835                          *=====================================================================
002836                          *
002837                          * INPUT NEXT STATUS TO R5
002838                          *
002839  0D25  8F00 117A         INXT   SAVE   <SAV1,=Z'0008'        B4
        0D27  0008
002840  0D28  C800 1145              LDR     $R4,<FINS             GET CONTROL WORD FOR INPUT NEXT STATUS
002841  0D2A  0F86                   B       >N1A
002842                          *=====================================================================
002843                          *
002844                          *           INPUT RANGE
002845                          *
002846                          *             $R5 - RESULT OF RANGE INPUTED
002847                          *
002848  0D2B  8F00 117A         INRG   SAVE   <SAV1,=Z'0008'        SAVE REG.
```

```
      0D2D  0008
002849 0D2E  C800 1143              LDR    $R4,<FIR             LOAD $R4 WITH FUNCTION CODE FOR INPUT RANGE
002850 0D30  C380 0D95       N1A    LNJ    $B4,<CGSCH           MODIFY FOR CHANNEL
002851 0D32  8055                   IO     =$R5,=$R4
       0D33  0054
002852 0D34  0703                   BIOT   >N1
002853 0D35  F380 0DBA              LNJ    $B7,<ERMB            ERROE:   I/O WAS NAK'ED
002854 0D37  8F80 117A       N1     RSTR   <SAV1,=Z'0008'
       0D39  0008
002855 0D3A  8384                   JMP    $B4
002856                       *
002857                       *=============================================================================
002858                       *
002859                       * SET UP LCT FOR CHANNEL SPECIFIED IN R3
002860                       *
002861                       *             LNJ $B5,<SETLCT
002862                       *             DC  TABLE
002863                       *
002864 0D3B  8F00 117A       SETLCT SAVE   <SAV1,=Z'E8E0'       R1,R2,R4,B2,B1
       0D3D  E8E0
002865 0D3E  9CF5                   LDB    $B1,+$B5             GET ADDRESS OF TABLE
002866 0D3F  8751                   CL     =$R1
002867                       *
002868 0D40  A811            LCT4A  LDR    $R2,$B1,$R1          GET BYTE TO OUTPUT
002869 0D41  8AD1                   INC    =$R1
002870 0D42  2985                   BNEZ   $R2,>LCT5A           BRANCH IF NOT AT END OF TABLE
002871 0D43  8F80 117A              RSTR   <SAV1,=Z'E8E0'
       0D45  E8E0
002872 0D46  8385                   JMP    $B5                  RETURN
002873                       *
002874 0D47  C800 1141       LCT5A  LDR    $R4,<FOBL            FUNCTION CODE FOROUT LCT BYTE
002875 0D49  C380 0D95              LNJ    $B4,<CGSCH           FORM IO CONTROL WORD
002876 0D4B  8052            LCT3A  IO     =$R2,=$R4            OUTPUT BYTE
       0D4C  0054
002877 0D4D  07FE                   BIOF   >LCT3A               CHECK IF TAKEN
002878 0D4E  0FF2                   B      >LCT4A               GET NEXT BYTE
002879                       *
002880                       *=============================================================================
002881                       *
002882                       *          FIND FIRST ACTIVE LINE
002883                       *
002884                       *            $R3-START CHECK ON THIS LINE AND
002885                       *                CYCLE THROUGH ALL LINES IF
002886                       *                ONE FOUND ON LINE NUMBER IS HERE
002887                       *
002888                       *             LNJ  $B5,<FLN
002889                       *               DC  OF LINE TYPE  TO SEARCH FOR
002890                       *             <RETURN> IF NO LINES ON
002891                       *             <RETURN>
```

```
002892                                  *
002893  0D4F  8F00 117A                 FLN     SAVE    <SAV1,=Z'A0E0'          SAVE REG. $R2,$B2,$B1
        0D51  A0E0
002894  0D52  3041                             SOR     $R3,1
002895  0D53  9CF5                             LDB     $B1,+$B5
002896  0D54  A801                             LDR     $R2,$B1                 PUT LINE TYPE INTO $R2
002897  0D55  AB80 11CB                        LAB     $B2,<ATLT               ADDRESS OF ACTIVE LINE TABLE
002898  0D57  3DC8                     FLN2    CMV     $R3,=8                  SEE IF AT BOTTOM OF TABLE
002899  0D58  0287                             BGE     >FLN3
002900  0D59  A97E                             CMR     $R2,$B2.+$R3            SEE IF INDEX LINE ON
002901  0D5A  09FD                             BNE     >FLN2
002902  0D5B  8803                             DEC     =$R3                    SET $R3 BACK TO VALUE FOUND
002903  0D5C  3001                             SOL     $R3,1
002904  0D5D  DBC5 0001                        LAB     $B5,$B5.1               INCREMENT TO VALID RETURN
002905  0D5F  8F80 117A                 FLN3    RSTR    <SAV1,=Z'A0E0'          RESTORE REG.
        0D61  A0E0
002906  0D62  8385                             JMP     $B5                     JUMP OUT OF ROUTINE
002907                                  *
002908                                  *=======================================================================
002909                                  *
002910                                  *                       TIME DELAY
002911                                  *
002912                                  *           LNJ  $B5,<TDLAY
002913                                  *
002914  0D63  8F00 117A                 TDLAY   SAVE    <SAV1,=Z'C180'          SAVE REG. $R1,$R7
        0D65  C180
002915  0D66  1C09                     TDLA    LDV     $R1,=9                  LOAD CLOCK INITIAL COUNT
002916  0D67  9F00 0000          X             STR     $R1,<ZHRTCC
002917  0D69  1C05                             LDV     $R1,=5
002918  0D6A  0004                             RTCN                            TURN ON REAL TIME CLOCK
002919  0D6B  9900 0000          X     TDLA1   CMR     $R1,<ZHRTCC
002920  0D6D  0882                             BAGE    >TDLA3
002921  0D6E  0FFD                             B       >TDLA1
002922  0D6F  0005                     TDLA3   RTCF                            TURN OFF REAL TIME CLOCK
002923  0D70  7776                             BDEC    $R7,>TDLA
002924  0D71  8F80 117A                        RSTR    <SAV1,=Z'C180'          RESTORE $R1
        0D73  C180
002925  0D74  8385                             JMP     $B5
002926                                  *
002927                                  *=======================================================================
002928                                  *
002929                                  *           READ LCT
002930                                  *
002931                                  *             LNJ   $B6,<RDLCT
002932                                  *             DC    <ADDRESS>
002933                                  *             DC    <RANGE IN BYTES>
002934                                  *
002935  0D75  8F00 117A                 RDLCT   SAVE    <SAV1,=Z'C88C'          SAVE REGS.
        0D77  C88C
```

```
002936  0D78  C800 1143                LDR     $R4,<FIR              LOAD DUMMY I/O
002937  0D7A  C380 0D95                LNJ     $B4,<CGSCH            PUT PROPER CHANNEL IN
002938  0D7C  C570 FF80                AND     $R4,=Z'FF80'
002939  0D7E  CF00 1151                STR     $R4,<CHNZ             STORE CHANNEL NUMBER  FOR CALL
002940  0D80  C806                     LDR     $R4,$B6               STORE ADDRESS IN ZV$MLR CALL
002941  0D81  CF00 1168                STR     $R4,<RLCTAD
002942  0D83  C846 0001                LDR     $R4,$B6.1             STORE RANGE FOR ZV$MLR
002943  0D85  CF00 1169                STR     $R4,<RLCTRG
002944                                 CALL    ZV$MLR,LCTV,RLCTRG,RLCTAD,CHNZ
        0D87  FBC0 0003
        0D89  D380 0000       X
        0D8B  0F80
        0D8C  1173
        0D8D  1169
        0D8E  1168
        0D8F  1151
002945  0D90  8F80 117A                RSTR    <SAV1,=Z'C88C'
        0D92  C88C
002946  0D93  83C6 0002                JMP     $B6.2
002947                           *
002948                           *=============================================================================
002949                           *
002950                           *               CHANGE SUBCHANNEL
002951                           *
002952                           *      $R3 -CONTAINS SUBCHANNEL WANTED
002953                           *      $R4 -CONTAINS I/O CONTROL WORD TO BE CHANGED
002954                           *
002955                           *       LNJ     $B4,<CGSCH
002956                           *
002957                           *
002958  0D95  3006               CGSCH  SOL     $R3,6                 SHIFT TO SUBCHANNEL POSITION
002959  0D96  C453                       OR      $R4,=$R3              OR SUBCHANNEL NUMBER INTO CONTROL WORD
002960  0D97  3046                       SOR     $R3,6                 SHIFT TO NORMAL POSITION
002961  0D98  8384                       JMP     $B4                   GO BACK
002962                           *
002963                           *=============================================================================
002964                           *
002965                           *       CCB FORMATION
002966                           *
002967                           *         $R3 - CONTAINS SUBCHANNEL WANTED
002968                           *         $R4 - CONTAINS RANGE (NUMBER OF BYTES)
002969                           *
002970                           *         LNJ     $B5,<MCCB
002971                           *         DC      CPU ADDRESS
002972                           *         DC      RAM ADDRESS NUMBER OR CHANNEL CONTROL WORD.
002973                           *
002974  0D99  8F00 117A           MCCB   SAVE    <SAV1,=Z'FDF8'        SAVES $B1,$B3,$B2,$B4,$R7,$R5,$R4,R2,& $R1
        0D9B  FDF8
002975  0D9C  7C7E                       LDV     $R7,=126
```

```
002976  0D9D  ACF5                    LDB   $B2,+$B5          LOAD $B2 WITH CPU ADDRESS
002977  0D9E  A805                    LDR   $R2,$B5           PUT RAM ADDRESS IN $R2
002978  0D9F  D854                    LDR   $R5,=$R4
002979  0DA0  C800 1140               LDR   $R4,<FOLD         LOAD $R4 WITH I/O CONTROL WORD
002980  0DA2  C380 0D95               LNJ   $B4,<CGSCH
002981  0DA4  7900 0DB3        MCB5   BEZ   $R7,<MCER
002982  0DA6  8182                    IOLD  $B2,=$R4,=$R5     OUTPUT ADDRESS AND RANGE
        0DA7  0054
        0DA8  0055
002983  0DA9  88D7                    DEC   =$R7
002984  0DAA  07FA                    BIOF  >MCB5
002985  0DAB  C800 1142               LDR   $R4,<FOCB         LOAD $R4 WITH I/O CONTROL WORD
002986  0DAD  C380 0D95               LNJ   $B4,<CGSCH        PUT I/O CONTROL WORD IN $R4
002987  0DAF  8052              MCB2   IO    =$R2,=$R4        OUTPUT MLCC RAM ADDRESS
        0DB0  0054
002988  0DB1  07FE                    BIOF  >MCB2
002989  0DB2  0F83                    B     >MCB6
002990  0DB3  F380 0DBA         MCER   LNJ   $B7,<ERMB        IOLD HAS BEEN NAK'ED
002991  0DB5  8F80 117A         MCB6   RSTR  <SAV1,=Z'FDF8'   RESTORE REGS.
        0DB7  FDF8
002992  0DB8  83C5 0001                JMP   $B5.1
002993                          *
002994                          *=============================================================================
002995                          *
002996                          *               ERROR PRINT ROUTINE
002997                          *
002998  0DBA  8F00 11AA         ERMB   SAVE  <SAV4,=Z'FFFF'   SAVE ALL REGS. FOR PRINT OUT
        0DBC  FFFF
002999  0DBD  C870 4D42                LDR   $R4,=A'MB'        STORE "MB" FOR PRINT
003000  0DBF  CF00 13A8                STR   $R4,<ERMG
003001  0DC1  0F88                    B     >ERROR
003002                          *
003003  0DC2  8F00 11AA         ERDB   SAVE  <SAV4,=Z'FFFF'
        0DC4  FFFF
003004  0DC5  C870 4442                LDR   $R4,=A'DB'
003005  0DC7  CF00 13A8                STR   $R4,<ERMG
003006                          *
003007  0DC9  8800 13A9         ERROR  LBF   <ERMG+1,=X'FF'
        0DCB  00FF
003008  0DCC  3041                     SOR   $R3,1             SHIFT TO GET LINE NUMBER
003009  0DCD  3E30                     ADV   $R3,=X'30'        CHANGE NUMBER INTO ASCII
003010  0DCE  B400 13A9               OR    $R3,<ERMG+1
003011  0DD0  BF00 13A9               STR   $R3,<ERMG+1       STORE LINE NUMBER
003012  0DD2  1CF0                    LDV   $R1,=-16          INTIALIZE LOOP COUNTER
003013  0DD3  8753                    CL    =$R3              CLEAR INDEX
003014  0DD4  EBC7 FFFE               LAB   $B6,$B7.-1-$AF    DECREMENT $B7
003015  0DD6  EF80 0DDD               STB   $B6,<ERR1+4+$AF   STORE ERROR ADDRESS FOR ERROR CALL
003016                          ERR1   CALL  ZV$ER,$,ERMG
        0DD8  FBC0 0003
```

```
          0DDA  D380 0000      X
          0DDC  0F80
          0DDD  0DD8
          0DDE  13A8
003017    0DDF  EB80 11E0              LAB    $B6,<CRLF            PUT CR-L INTO CALL
003018    0DE1  CB80 128E              LAB    $B4,<SPACE
003019    0DE3  AB80 0DF5              LAB    $B2,<ERR3+4+2*$AF
003026    0DE5  EF82             ER11  STB    $B6,$B2
003027    0DE6  8980 115B              CMZ    <ERF                 CHECK FLAG FOR SHORT REPORT
003028    0DE8  0900 0E00              BE     <ERR5
003029    0DEA  BB80 11AA        ERR2  LAB    $B3,<SAV4.$R3        FECTH AND STORE REGISTER
003030    0DEC  9B80 0DF4              LAB    $B1,<ERR3+4+$AF
003036    0DEE  BF81             ERR21 STB    $B3,$B1
003044          ~                ERR3  CALL   ZV$THZ,$,$
          0DEF  FBC0 0003
          0DF1  D380 0000      X
          0DF3  0F80
          0DF4  0DEF
          0DF5  0DEF
003049    0DF6  8AD3             ERR34 INC    =$R3                 BUMP INDEX
003050    0DF7  CF82                   STB    $B4,$B2
003051    0DF8  3D08                   CMV    $R3,=7*$AF+1         SEE IF CR-LF NEEDED
003052    0DF9  0984                   BNE    >ERR4
003053    0DFA  EB80 11E0              LAB    $B6,<CRLF
003054    0DFC  EF82                   STB    $B6,$B2              PUT CR-LF INTO CALL
003055    0DFD  1780 0DEA        ERR4  BINC   $R1,<ERR2            DO NEXT REGISTER
003056    0DFF  0000                   HLT                         HALT AFTER ERROR
003057    0E00  8F80 11AA        ERR5  RSTR   <SAV4,=Z'FFFF'       RESTORE REGISTERS
          0E02  FFFF
003058    0E03  8387                   JMP    $B7                  RETURN
003068                           SAFF  NULL
003069                           *
003070                           *=============================================================================
003071                           *
003072                           *                END OF PASS CHECK
003073                           *
003074    0E04  8700 00FF        EDPCK CL     <STOP
003075    0E06  8980 0000      X       CMZ    <ZV$TTY
003076    0E08  0900 00FF              BE     <STOP
003077                                 CALL   ZV$TC,MESG8
          0E0A  FBC0 0003
          0E0C  D380 0000      X
          0E0E  0F80
          0E0F  137B
003078    0E10  8382                   JMP    $B2
003079                           *              CHANNEL PROGRAMS
003080                           *
003081                           *
003082                           *
```

```
003083                          *
003084                          *       CHANNEL PROHRAM FOR REV H
003085                          *          SET TEST MODE ON CLA FOR SECOND LINE OF CLA
003086                          *        ORG    X'200'
003087          0E11            CHLP    EQU    $
003088                          *        LD     20
003091  0E11    5014
003092                          *        OUT    2
003095                          *        WAIT
003096  0E12    3201
003097                          *        NOP
003098                          *
003099                          *          SET CURRENT LOOP ADAPTER TO FASTEST SPEED
003100                          *        ORG    X'200'
003101  0E13    0000
003102          0E14            CHLPS   EQU    $
003103                          *        LD     =X'F'
003106  0E14    900F
003107                          *        OUT    4
003110                          *        WAIT
003111  0E15    3401
003112                          *        NOP
003113                          *
003114                          *                    CHANNEL PROGRAM FOR LOOP AT LINE ADAPT (REC)
003115                          *
003116                          *        ORG    X'200'
003117  0E16    0000
003118          0E17            CHLP1   EQU    $
003119                          *        LOC    HLP1
003120          0200            HLP1 EQU X'0200'
003121                          *        LD     24                  GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
003124  0E17    5018
003125                          *        OUT    4                   OUTPUT IT
003128                          *        RECV   0                   DUMMY REC. FOR CURRENT LOOP ADAPT
003131  0E18    34A0
003132                          *        LD     20                  GET LINE CONTROL
003135  0E19    5014
003136                          *        OUT    2                   OUTPUT IT
003139                          *        WAIT
003140  0E1A    3201
003141                          *        LD     28                  LOAD SYNC. FLAG IF ZERO LINE IS ASYNC.
003144  0E1B    501C
003145                          *        BZT    GETC
003148  0E1C    E211
003149                          *        LOC    TRNX
003150          020C            TRNX EQU X'020C'
003151                          *##
003152                          *#
003153                          *        RECV   0                   INPUT CHARACTER
```

```
003156                          *       C       24
003157  0E1D  A052
003160                          *       BEF     CKFF
003161  0E1E  18F1
003164                          *       WAIT
003165  0E1F  0401
003166                          *       B       TRNX
003169  0E20  E0F9
003170                          *       LOC     CKFF
003171        0214              CKFF EQU X'0214'
003172                          *       C       =X'FF'
003175  0E21  92FF
003176                          *       BET     GETC1
003179  0E22  E109
003180                          *       SFS
003181                          *       WAIT
003182  0E23  0301
003183                          *       B       TRNX
003186  0E24  E0F1
003187                          *#
003188                          *##
003189                          *       LOC     GETC
003190        021C              GETC EQU X'021C'
003191                          *       RECV    0               INPUT CHARACTER
003194                          *       ST,     STORE           INTO DATA BUFFER
003195  0E25  A011
003196                          *       BLCT    HLP1A
003199  0E26  E304
003200                          *       LOC     GETC1
003201        0220              GETC1 EQU X'0220'
003202                          *       WAIT
003203                          *       B       GETC
003204  0E27  01E0
003207                          *       LOC     HLP1A
003208        0223              HLP1A EQU X'0223'
003209                          *       BLBT    GETX
003210  0E28  FAE4
003213                          *       GNB
003214  0E29  0402
003215                          *       B       GETC
003218  0E2A  E0F5
003219                          *       LOC     GETX
003220        0228              GETX EQU X'0228'
003221                          *       LD      =0              TURN OFF REC.
003224  0E2B  9000
003225                          *       OUT     2
003228                          *       GNB
003229  0E2C  3202
003230                          *       WAIT
```

```
003231                          *       B       HLP1
003232  0E2D  01E0
003235                          *       NOP
003236  0E2E  D200
003237                          *
003238                          *         CHANNEL PROGRAM FOR LOOP AT LINE ADAPT (XMIT)
003239                          *
003240                          *       ORG     X'400'
003241        0E2F              CHLP1A  EQU     $
003242                          *       LOC     LP1A
003243        0400              LP1A    EQU X'0400'
003244                          *       LD      2                       GET LINE CONFIGURATION
003247  0E2F  5002
003248                          *       OUT     6                       OUTPUT IT
003251                          *       LD      =X'AA'                  LOAD FILL CHAR.
003252  0E30  3690
003255                          *       OUT     4                       OUTPUT SYNC. FILL CHAR.
003258  0E31  AA34
003259                          *       WAIT
003260                          *       LD      28                      LOAD SYNC. FLAG
003261  0E32  0150
003264                          *       BZT     LP1ABD                  IF ZERO BRANCH TO OUTPUT CHAR.
003265  0E33  1CE2
003268                          *       LD      =6
003269  0E34  1590
003272                          *       LOC     XMSY
003273        040D              XMSY    EQU X'040D'
003274                          *       ST      63
003275  0E35  0651
003278                          *       LD      24                      LOAD SYNC. CHAR
003279  0E36  3F50
003282                          *       SEND    0                       SEND IT
003285  0E37  1860
003286                          *       LD      63                      DO SIX TIMES
003289  0E38  503F
003290                          *       DEC
003291                          *       BZT     SDSY
003292  0E39  05E2
003295                          *       WAIT
003296  0E3A  0401
003297                          *       B       XMSY
003300  0E3B  E0F4
003301                          *       LOC     SDSY
003302        041A              SDSY    EQU X'041A'
003303                          *       WAIT
003304                          *       LD      =X'FF'
003305  0E3C  0190
003308                          *       SEND    0
003311  0E3D  FF60
```

```
003312                          *       WAIT
003313                          *       LOC     LP1ABD
003314          041F            LP1ABD EQU X'041F'
003315                          *       LD      GET             CHARACTER FROM DATA BUFFER
003316  0E3E    0110
003317                          *       SEND    0               OUTPUT IT
003320                          *       BLCT    LP1AB
003321  0E3F    60E3
003324                          *       LOC     LP1AA
003325          0423            LP1AA EQU X'0423'
003326                          *       WAIT
003327  0E40    0401
003328                          *       B       LP1ABD
003331  0E41    E0FA
003332                          *       LOC     LP1AB
003333          0426            LP1AB EQU X'0426'
003334                          *       WAIT
003335                          *       LD      =X'FF'
003336  0E42    0190
003339                          *       SEND    0
003342  0E43    FF60
003343                          *       BLBT    LP1AC
003346  0E44    E404
003347                          *       GNB
003348                          *       B       LP1AA
003349  0E45    02E0
003352                          *       LOC     LP1AC
003353          042F            LP1AC EQU X'042F'
003354                          *       LD      28
003355  0E46    F550
003358                          *       BZT     LP1AC1
003359  0E47    1CE2
003362                          *       WAIT
003363  0E48    0501
003364                          *       LD      =X'FF'
003367  0E49    90FF
003368                          *       SEND    0
003371                          *       LOC     LP1AC1
003372          0437            LP1AC1 EQU X'0437'
003373                          *       GNB
003374  0E4A    6002
003375                          *       WAIT
003376                          *       B       LP1A
003377  0E4B    01E0
003380                          *       NOP
003381  0E4C    C600
003382                          *
003383                          *           RECEIVE PROGRAM FOR CRC AND PARITY
003384                          *
```

```
003385                          *       ORG     X'200'
003386          0E4D            CHLP2   EQU     $
003387                          *       LD      24              GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
003390  0E4D    5018
003391                          *       OUT     4               OUTPUT IT
003394                          *       LD      20              LOAD LINE CONTROL
003395  0E4E    3450
003398                          *       OUT     2               OUTPUT IT
003401  0E4F    1432
003402                          *       RECV    0
003405                          *       WAIT
003406  0E50    A001
003407                          *       LD      24              SEE IF SYNC. LINE
003410  0E51    5018
003411                          *       C       =X'F'           IF SYNC. SET UP TO  RECV.
003414  0E52    920F
003415                          *       BET     RCVA            4 OR 5 SYNC. CHAR.
003418  0E53    E11B
003419                          *       LOC     JHQQ
003420          020E            JHQQ EQU X'020E'
003421                          *       RECV    0               RECEIVE FIRST SYNC CHAR.
003424                          *       WAIT
003425  0E54    A001
003426                          *       RECV    0               RECEIVE SECOND SYNC.
003429                          *       C       24              CHECK TO SEE IF TRUE SYNC.
003430  0E55    A052
003433                          *       BET     JFQQ
003434  0E56    18E1
003437                          *       LD      20              SHUT OFF LINE
003438  0E57    0B50
003441                          *       AND     =X'FD'
003442  0E58    1493
003445                          *       OUT     2
003448  0E59    FD32
003449                          *       LD      20              SHUT LINE BACK ON
003452  0E5A    5014
003453                          *       OUT     2
003456                          *       B       JHQQ
003457  0E5B    32E0
003460                          *       LOC     JFQQ
003461          021F            JFQQ EQU X'021F'
003462                          *       WAIT
003463  0E5C    F001
003464                          *       RECV    0               RECEIVE THIRD SYNC.
003467                          *       LOC     NSTR
003468          0221            NSTR EQU X'0221'
003469                          *       WAIT
003470  0E5D    A001
003471                          *       RECV    0               RECEIVE SYNC. OR START OF DATA CHAR. (FF)
```

```
003474                          *       C       =X'FF'
003475  0E5E  A092
003478                          *       BEF     NSTR
003479  0E5F  FFF1
003482                          *       WAIT
003483  0E60  FB01
003484                          *       LOC     RCVA
003485        0228              RCVA EQU X'0228'
003486                          *       LD      23              LOAD FLAG FOR TYPE OF RECV.
003489  0E61  5017
003490                          *       C       =1              IF TRUE TEST JUST CRC
003493  0E62  9201
003494                          *       BET     RECV1
003497  0E63  E10C
003498                          *       C       =X'F0'          IF TRUE TEST JUST
003501  0E64  92F0
003502                          *       BET     RECV2
003505  0E65  E10F
003506                          *       LOC     RECV3
003507        0232              RECV3 EQU X'0232'
003508                          *       RECV    3               TEST CRC AND PARITY
003511                          *       ST,     STORE           INTO BUFFER
003512  0E66  A311
003513                          *       BLCT    RCV             BRANCH IF LAST CHARACTER TRUE
003516  0E67  E312
003517                          *       WAIT
003518                          *       B       RECV3
003519  0E68  01E0
003522                          *
003523                          *       LOC     RECV1
003524        0239              RECV1 EQU X'0239'
003525                          *       RECV    1               TEST CRC
003528  0E69  FAA1
003529                          *       ST,     STORE           IN BUFFER
003530                          *       BLCT    RCV
003531  0E6A  11E3
003534                          *       WAIT
003535  0E6B  0B01
003536                          *       B       RECV1
003539  0E6C  E0FA
003540                          *
003541                          *       LOC     RECV2
003542        0240              RECV2 EQU X'0240'
003543                          *       RECV    2               TEST PARITY
003546                          *       ST,
003547  0E6D  A211
003548                          *       BLCT    RCV
003551  0E6E  E304
003552                          *       WAIT
```

```
003553                          *       B       RECV2
003554  0E6F    01E0
003557                          *
003558                          *       LOC     RCV
003559          0247            RCV EQU X'0247'
003560                          *       LD      =0                      LOAD ZERO TO TURN OFF
003561  0E70    FA90
003564                          *       OUT     2                       RECEIVE
003567  0E71    0032
003568                          *       GNB
003569                          *       WAIT
003570  0E72    0201
003571                          *       B       RCVA
003574  0E73    E0DB
003575                          *       NOP
003576                          *
003577                          *               XMIT PROGRAM FOR CRC AND PARITY TEST
003578                          *
003579                          *       ORG     X'400'
003580  0E74    0000
003581          0E75            CHLP2A EQU      $
003582                          *       LD      2                       GET LINE CONFIGURATION
003585  0E75    5002
003586                          *       OUT     6                       OUTPUT IT
003589                          *       LD      =X'FA'                  LOAD FILL CHAR.
003590  0E76    3690
003593                          *       OUT     4                       OUTPUT SYNC. FILL CHAR.
003596  0E77    FA34
003597                          *       WAIT
003598                          *       LD      24                      SEE IF SYNC. LINE
003599  0E78    0150
003602                          *       C       =X'F'                   IF SYNC. LINE SEND OUT FILL CHAR (LR 4)
003603  0E79    1892
003606                          *       BET     SNC1                    AND 3 SYNC. (OR FILL)
003607  0E7A    0FE1
003610                          *       LD      =X'FA'                  LOAD FILL CHAR.
003611  0E7B    1890
003614                          *       OUT     4
003617  0E7C    FA34
003618                          *       LD      =6
003621  0E7D    9006
003622                          *       LOC     XMVV
003623          0412            XMVV EQU X'0412'
003624                          *       ST      63
003627  0E7E    513F
003628                          *       LD      24                      LOAD SYNC. CHAR
003631  0E7F    5018
003632                          *       SEND    0                       SEND IT
003635                          *       LD      63                      DO SIX TIMES
```

```
003636  0E80  6050
003639                    *       DEC
003640  0E81  3F05
003641                    *       BZT     SDVV
003644  0E82  E204
003645                    *       WAIT
003646                    *       B       XMVV
003647  0E83  01E0
003650                    *       LOC     SDVV
003651        041F        SDVV EQU X'041F'
003652                    *       WAIT
003653  0E84  F401
003654                    *       LD      =X'FF'
003657  0E85  90FF
003658                    *       SEND    0               XMIT START OF DATA CHAR. (FF)
003661                    *       WAIT
003662  0E86  6001
003663                    *       LOC     SNC1
003664        0424        SNC1 EQU X'0424'
003665                    *       LD      23              LOAD FLAG FOR TYPE OF SEND INST.
003668  0E87  5017
003669                    *       C       =1              IF TRUE TEST JUST CRC
003672  0E88  9201
003673                    *       BET     CPCRC
003676  0E89  E10C
003677                    *       C       =X'F0'          IF TRUE TEST PARITY
003680  0E8A  92F0
003681                    *       BET     CPPAR
003684  0E8B  E10F
003685                    *       LOC     SNC3            TEST BOTH CRC AND PARITY ON XMIT
003686        042E        SNC3 EQU X'042E'
003687                    *       LD,
003688                    *       SEND    3
003691  0E8C  1063
003692                    *       BLCT    LCB1            BRANCH ON LAST CHAR.
003695  0E8D  E312
003696                    *       WAIT
003697                    *       B       SNC3            DUE NEXT CHAR.
003698  0E8E  01E0
003701                    *       LOC     CPCRC
003702        0435        CPCRC EQU X'0435'
003703                    *       LD,     LOAD            CHAR. FROM DATA BUFFER
003704  0E8F  FA10
003705                    *       SEND    1               XMIT CHAR. AND CALCULATE CRC
003708                    *       BLCT    LCB1            BRANCH ON LAST CHAR.
003709  0E90  61E3
003712                    *       WAIT
003713  0E91  0B01
003714                    *       B       CPCRC           DUE NEXT CHAR.
```

```
003717  0E92  E0FA
003718                        *       LOC     CPPAR
003719        043C            CPPAR EQU X'043C'
003720                        *       LD,     LOAD            CHAR. FROM DATA BUFFER
003721                        *       SEND    2               MIT CHAR. WITH PARITY
003724  0E93  1062
003725                        *       BLCT    LCB1
003728  0E94  E304
003729                        *       WAIT
003730                        *       B       CPPAR
003731  0E95  01E0
003734                        *       LOC     LCB1
003735        0443            LCB1 EQU X'0443'
003736                        *       WAIT
003737  0E96  FA01
003738                        *       LD      =X'FF'          OUTPUT DUMMY CHAR. FOR SHIFT REG.
003741  0E97  90FF
003742                        *       SEND    0
003745                        *       BLBT    LCB2            BRANCH ON LAST BLOCK
003746  0E98  60E4
003749                        *       B       SNC1
003750  0E99  03E0
003753                        *       LOC     LCB2
003754        044B            LCB2 EQU X'044B'
003755                        *       LD      24
003756  0E9A  DA50
003759                        *       C       =X'F'
003760  0E9B  1892
003763                        *       BET     NWT
003764  0E9C  0FE1
003767                        *       WAIT
003768  0E9D  0501
003769                        *       LD      =X'FF'
003772  0E9E  90FF
003773                        *       SEND    0
003776                        *       LOC     NWT
003777        0455            NWT EQU X'0455'
003778                        *       LD      20
003779  0E9F  6050
003782                        *       AND     =X'FE'
003783  0EA0  1493
003786                        *       OUT     2
003789  0EA1  FE32
003790                        *       GNB
003791                        *       WAIT
003792  0EA2  02C1
003793                        *       B       SNC1
003796  0EA3  E0C7
003797                        *       NOP
```

```
003798                          *
003799                          *          RECEIVE PROGRAM FOR CHARACTER SIZE TEST
003800                          *
003801                          *       ORG     X'200'
003802  0EA4  0000
003803        0EA5              CHLP3   EQU     $
003804                          *       LOC     RCP3
003805        0200              RCP3 EQU X'0200'
003806                          *       LD      24                  GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
003809  0EA5  5018
003810                          *       OUT     4                   OUTPUT IT
003813                          *       LD      20
003814  0EA6  3450
003817                          *       OUT     2                   OUTPUT LINE CONTROL
003820  0EA7  1432
003821                          *       RECV    0
003824                          *       WAIT
003825  0EA8  A001
003826                          *       LD      24                  SEE IF SYNC TO RECV.
003829  0EA9  5018
003830                          *       C       =X'F'
003833  0EAA  920F
003834                          *       BET     RCCH
003837  0EAB  E121
003838                          *       LOC     JHYY
003839        020E              JHYY EQU X'020E'
003840                          *       RECV    0                   INPUT. FIRST SYNC.
003843                          *       WAIT
003844  0EAC  A001
003845                          *       LD      24
003848  0EAD  5018
003849                          *       AND     28
003852  0EAE  531C
003853                          *       ST      59
003856  0EAF  513B
003857                          *       RECV    0                   INPUT SECOND SYNC.
003860                          *       C       59                  CHECK TO SEE IF TRUE SYNC.
003861  0EB0  A052
003864                          *       BET     JFYY
003865  0EB1  3BE1
003868                          *       LD      20                  SHUT OFF LINE
003869  0EB2  0B50
003872                          *       AND     =X'FD'
003873  0EB3  1493
003876                          *       OUT     2
003879  0EB4  FD32
003880                          *       LD      20                  SHUT LINE BACK ON
003883  0EB5  5014
003884                          *       OUT     2
```

```
003887                          *       B       JHYY
003888  0EB6  32E0
003891                          *       LOC     JFYY
003892        0225              JFYY EQU X'0225'
003893                          *       WAIT
003894  0EB7  EAC1
003895                          *       RECV    0                       INPUT THIRD SYNC.
003898                          *       LOC     RCKSY
003899        0227              RCKSY EQU X'0227'
003900                          *       WAIT
003901  0EB8  A001
003902                          *       RECV    0
003905                          *       C       =X'1F'                  SEE IF START OF DATA CHARACTER
003906  0EB9  A092
003909                          *       BEF     RCKSY
003910  0EBA  1FF1
003913                          *       WAIT
003914  0EBB  FB01
003915                          *       LOC     RCCH
003916        022E              RCCH EQU X'022E'
003917                          *       LD      2                       CHECK FOR FIVE BIT CHAR.
003920  0EBC  5002
003921                          *       BZT     NPRY
003924  0EBD  E208
003925                          *    RECV. CHAR WITH PARITY 6-7 BIT CHAR.
003926                          *       LOC     PRYT
003927        0232              PRYT EQU X'0232'
003928                          *       RECV    2
003931                          *       ST
003932  0EBE  A211
003933                          *       BLCT    SHFL
003936  0EBF  E30B
003937                          *       WAIT
003938                          *       B       PRYT
003939  0EC0  01E0
003942                          *    RECV. FIVE BIT CHAR. NO PARITY
003943                          *       LOC     NPRY
003944        0239              NPRY EQU X'0239'
003945                          *       RECV    0
003948  0EC1  FAA0
003949                          *       ST
003950                          *       BLCT    SHFL
003951  0EC2  11E3
003954                          *       WAIT
003955  0EC3  04C1
003956                          *       B       NPRY
003959  0EC4  E0FA
003960                          *     TURN OFF THE LINE
003961                          *       LOC     SHFL
```

```
003962          0240            SHFL EQU X'0240'
003963                          *       LD      =0
003966  0EC5    9000
003967                          *       OUT     2
003970                          *       GNB
003971  0EC6    3202
003972                          *       WAIT
003973                          *       B       RCP3
003974  0EC7    01E0
003977                          *       NOP
003978  0EC8    BA00
003979                          *
003980                          *          TRANSMIT PROGRAM FOR CHARACTER SIZE TEST
003981                          *       ORG     X'400'
003982          0EC9            CHLP3A EQU      $
003983                          *       LD      2                       GET LINE CONFIGURATION
003986  0EC9    5002
003987                          *       OUT     6                       OUTPUT IT
003990                          *       LD      =X'FA'                  LOAD FILL CHAR.
003991  0ECA    3690
003994                          *       OUT     4                       OUTPUT SYNC. FILL CHAR.
003997  0ECB    FA34
003998                          *       WAIT
003999                          *       LD      =X'1F'                  LOAD DEFAULT MASK TO STRIP 5 BITS
004000  0ECC    0190
004003                          *       ST      27
004004  0ECD    1F51
004007                          *       ST      28
004008  0ECE    1B51
004011                          *       LD      =X'FA'                  LOAD FILL CHAR.
004012  0ECF    1C90
004015                          *       OUT     4                       OUTPUT SYNC. CHAR.
004018  0ED0    FA34
004019                          *       LD      2
004022  0ED1    5002
004023                          *       BZT     XPYQ
004026  0ED2    E222
004027                          * SET MASK TO STRIP BITS FOR SYNC. CHAR. ON RECV. SIDE
004028                          *       C       =X'40'
004031  0ED3    9240
004032                          *       BEF     N61
004035  0ED4    F10A
004036                          *       LD      =X'3F'
004039  0ED5    903F
004040                          *       ST      28
004043  0ED6    511C
004044                          *       SR
004045                          *       ST      27
004046  0ED7    0751
```

```
004049                          *       B       XPYQ
004050  0ED8    1BE0
004053                          *       LOC     N61
004054          0269            N61 EQU X'0269'
004055                          *       C       =X'80'
004056  0ED9    1592
004059                          *       BEF     N71
004060  0EDA    80F1
004063                          *       LD      =X'7F'
004064  0EDB    0A90
004067                          *       ST      28
004068  0EDC    7F51
004071                          *       SR
004072  0EDD    1C07
004073                          *       ST      27
004076  0EDE    511B
004077                          *       B       XPYQ
004080  0EDF    E008
004081                          *       LOC     N71
004082          0276            N71 EQU X'0276'
004083                          *       LD      =X'FF'
004086  0EE0    90FF
004087                          *       ST      28
004090  0EE1    511C
004091                          *       SR
004092                          *       ST      27
004093  0EE2    0751
004096                          *       LOC     XPYQ
004097          027D            XPYQ EQU X'027D'
004098                          *       LD      24
004099  0EE3    1B50
004102                          *       C       =X'F'
004103  0EE4    1892
004106                          *       BET     XMNS            BRANCH ON ASYNC. LINE
004107  0EE5    0FE1
004110                          *       LD      =6
004111  0EE6    1590
004114                          *       LOC     XMZZ
004115          0285            XMZZ EQU X'0285'
004116                          *       ST      63
004117  0EE7    0651
004120                          *       LD      24              LOAD SYNC. CHAR
004121  0EE8    3F50
004124                          *       SEND    0               SEND IT
004127  0EE9    1860
004128                          *       LD      63              DO SIX TIMES
004131  0EEA    503F
004132                          *       DEC
004133                          *       BZT     SDZZ
```

```
004134  0EEB  05E2
004137                        *       WAIT
004138  0EEC  0401
004139                        *       B       XMZZ
004142  0EED  E0F4
004143                        *       LOC     SDZZ
004144        0292            SDZZ EQU X'0292'
004145                        *       WAIT
004146                        *       LD      =X'1F'
004147  0EEE  0190
004150                        *       SEND    0                       XMIT START OF DATA CHAR.
004153  0EEF  1F60
004154                        *       WAIT
004155                        *       LOC     XMNS
004156        0297            XMNS EQU X'0297'
004157                        *       LD      2
004158  0EF0  0150
004161                        *       BZT     XNPY
004162  0EF1  02E2
004165                        *       SEND DATA WITH PARITY
004166                        *       LOC     XPY
004167        029B            XPY EQU X'029B'
004168                        *       LD
004169  0EF2  0A10
004170                        *       AND     27
004173  0EF3  531B
004174                        *       SEND    2
004177                        *       BLCT    EDXT
004178  0EF4  62E3
004181                        *       WAIT
004182  0EF5  0D01
004183                        *       B       XPY
004186  0EF6  E0F8
004187                        *     SEND DATA FIVE BIT NO PARITY
004188                        *       LOC     XNPY
004189        02A4            XNPY EQU X'02A4'
004190                        *       LD
004191                        *       AND     27
004192  0EF7  1053
004195                        *       SEND    0
004198  0EF8  1B60
004199                        *       BLCT    EDXT
004202  0EF9  E304
004203                        *       WAIT
004204                        *       B       XNPY
004205  0EFA  01E0
004208                        *      XMIT DUMMY CHAR. TO SHIFT LAST DATA
004209                        *       LOC     EDXT
004210        02AD            EDXT EQU X'02AD'
```

```
004211                          *       WAIT
004212  0EFB  F801
004213                          *       LD      27
004216  0EFC  501B
004217                          *       SEND    0
004220                          *       BLBT    NXTB
004221  0EFD  60E4
004224                          *       WAIT
004225  0EFE  0401
004226                          *       B       XMNS
004229  0EFF  E0E2
004230                          *       LOC     NXTB
004231        02B6              NXTB EQU X'02B6'
004232                          *       LD      24
004235  0F00  5018
004236                          *       C       =X'F'
004239  0F01  920F
004240                          *       BET     EOP
004243  0F02  E105
004244                          *       WAIT
004245                          *       LD      27
004246  0F03  0150
004249                          *       SEND    0
004252  0F04  1B60
004253                          *       LOC     EOP
004254        02C0              EOP EQU X'02C0'
004255                          *       GNB
004256                          *       WAIT
004257  0F05  0201
004258                          *       B       EDXT
004261  0F06  E0EA
004262                          *       NOP
004263                          *
004264                          *          RECEIVE PROGRAM FOR OVERRUN TEST
004265                          *
004266                          *       ORG     X'200'
004267  0F07  0000
004268        0F08              CHLP4   EQU     $
004269                          *       LD      =10
004272  0F08  900A
004273                          *       ST      28
004276  0F09  511C
004277                          *       LD      2
004280  0F0A  5002
004281                          *       OUT     6
004284                          *       LD      24                  GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
004285  0F0B  3650
004288                          *       OUT     4                   OUTPUT IT
004291  0F0C  1834
```

```
004292                          *       LD      20
004295  0F0D  5014
004296                          *       OUT     2
004299                          *       RECV    0
004302  0F0E  32A0
004303                          *       WAIT
004304                          *
004305                          *       LOC     BEG
004306          020F            BEG EQU X'020F'
004307                          *       LD      24
004308  0F0F  0150
004311                          *       C       =X'F'
004312  0F10  1892
004315                          *       BET     RMCH
004316  0F11  0FE1
004319                          *       LOC     JHRR
004320          0215            JHRR EQU X'0215'
004321                          *       RECV    0               1ST SYNC.
004324  0F12  1BA0
004325                          *       WAIT
004326                          *       RECV    0               2ND SYNC.
004329  0F13  01A0
004330                          *       C       24              CHECK TO SEE IF TRUE SYNC.
004333  0F14  5218
004334                          *       BET     JFRR
004337  0F15  E10B
004338                          *       LD      20              SHUT OFF LINE
004341  0F16  5014
004342                          *       AND     =X'FD'
004345  0F17  93FD
004346                          *       OUT     2
004349                          *       LD      20              SHUT LINE BACK ON
004350  0F18  3250
004353                          *       OUT     2
004356  0F19  1432
004357                          *       B       JHRR
004360  0F1A  E0F0
004361                          *       LOC     JFRR
004362          0226            JFRR EQU X'0226'
004363                          *       WAIT
004364                          *       RECV    0               3RD SYNC.
004367  0F1B  01A0
004368                          *       LOC     NSCH
004369          0228            NSCH EQU X'0228'
004370                          *       WAIT
004371                          *       RECV    0
004374  0F1C  01A0
004375                          *       C       =X'FF'          SEE IF START DATA CHAR.
004378  0F1D  92FF
```

```
004379                          *       BEF     NSCH
004382  0F1E  F1FB
004383                          *       WAIT
004384                          *       LOC     RMCH
004385        022F              RMCH EQU X'022F'
004386                          *       RECV    0
004389  0F1F  01A0
004390                          *       ST
004391                          *       BLCT    REND
004392  0F20  11E3
004395                          *       WAIT
004396  0F21  0401
004397                          *       B       RMCH
004400  0F22  E0FA
004401                          *
004402                          *       LOC     REND
004403        0236              REND EQU X'0236'
004404                          *       LD      =0
004407  0F23  9000
004408                          *       OUT     2
004411                          *       GNB
004412  0F24  3202
004413                          *       WAIT
004414                          *       B       BEG
004415  0F25  01E0
004418                          *
004419                          *       LOC     DLAY
004420        023D              DLAY EQU X'023D'
004421                          *       LD      29
004422  0F26  D350
004425                          *       BZF     EDLY
004426  0F27  1DF2
004429                          *       RET
004430  0F28  0206
004431                          *       LOC     EDLY
004432        0242              EDLY EQU X'0242'
004433                          *       ST      30
004436  0F29  511E
004437                          *       LD      =94
004440  0F2A  905E
004441                          *       LOC     LESS
004442        0246              LESS EQU X'0246'
004443                          *       DEC
004444                          *       BZT     NXC
004445  0F2B  05E2
004448                          *       B       LESS
004449  0F2C  03E0
004452                          *       LOC     NXC
004453        024B              NXC EQU X'024B'
```

```
004454                          *       LD      30
004455  0F2D    FC50
004458                          *       DEC
004459  0F2E    1E05
004460                          *       BZF     EDLY
004463  0F2F    F2F3
004464                          *       RET
004465                          *       NOP
004466  0F30    0600
004467                          *
004468                          *       TRANSMIT PROGRAM FOR UNDERRUN TEST
004469                          *
004470                          *       ORG     X'260'
004471          0F31            CHLP4A  EQU     $
004472                          *       LOC     TOP
004473          0260            TOP     EQU X'0260'
004474                          *       WAIT
004475                          *       LD      24
004476  0F31    0150
004479                          *       C       =X'F'
004480  0F32    1892
004483                          *       BET     NSY
004484  0F33    0FE1
004487                          *       LD      =X'AA'                  LOAD FILL CHAR.
004488  0F34    3090
004491                          *       OUT     4
004494  0F35    AA34
004495                          *       LD      =6
004498  0F36    9006
004499                          *       LOC     XMRR
004500          026C            XMRR    EQU X'026C'
004501                          *       ST      63
004504  0F37    513F
004505                          *       LD      24                      LOAD SYNC. CHAR
004508  0F38    5018
004509                          *       SEND    0                       SEND IT
004512                          *       LD      63                      DO SIX TIMES
004513  0F39    6050
004516                          *       DEC
004517  0F3A    3F05
004518                          *       BZT     SDRR
004521  0F3B    E204
004522                          *       WAIT
004523                          *       B       XMRR
004524  0F3C    01E0
004527                          *       LOC     SDRR
004528          0279            SDRR    EQU X'0279'
004529                          *       WAIT
004530  0F3D    F401
```

```
004531                          *       LD      =X'FF'
004534  0F3E    90FF
004535                          *       SEND    0
004538                          *       WAIT
004539  0F3F    6001
004540                          *       LD      =2
004543  0F40    9002
004544                          *       ST      26
004547  0F41    511A
004548                          *       LOC     NZ7
004549          0282            NZ7 EQU X'0282'
004550                          *       LD
004551                          *       SEND    0
004554  0F42    1060
004555                          *       LD      26
004558  0F43    501A
004559                          *       DEC
004560                          *       ST      26
004561  0F44    0551
004564                          *       BZT     CDLY
004565  0F45    1AE2
004568                          *       WAIT
004569  0F46    0401
004570                          *       B       NZ7
004573  0F47    E0F5
004574                          *
004575                          *       LOC     CDLY
004576          028E            CDLY EQU X'028E'
004577                          *       BS      DLAY            GO TO CHAR. DELAY
004580  0F48    F0AE
004581                          *       IN      5               SEE IF UNDERRUN BIT SET
004584                          *       ST      60
004585  0F49    2551
004588                          *       WAIT
004589  0F4A    3CC1
004590                          *       B       NT3
004593  0F4B    E016
004594                          *
004595                          *       LOC     NSY
004596          0296            NSY EQU X'0296'
004597                          *       LD
004598                          *       SEND    0
004601  0F4C    1060
004602                          *       LD      28
004605  0F4D    501C
004606                          *       DEC
004607                          *       ST      28
004608  0F4E    0551
004611                          *       BZT     OVLY
```

```
004612  0F4F  1CE2
004615                        *       WAIT
004616  0F50  0401
004617                        *       B       NSY
004620  0F51  E0F5
004621                        *       LOC     OVLY
004622        02A2            OVLY EQU X'02A2'
004623                        *       BS      DLAY
004626  0F52  F09A
004627                        *       SEND    0
004630                        *       BS      DLAY
004631  0F53  60F0
004634                        *       IN      5
004637  0F54  9725
004638                        *       ST      60
004641  0F55  513C
004642                        *       WAIT
004643                        *       LOC     NT3
004644        02AB            NT3 EQU X'02AB'
004645                        *       LD
004646  0F56  0110
004647                        *       SEND    0
004650                        *       BLCT    EXM1
004651  0F57  60E3
004654                        *       WAIT
004655  0F58  0401
004656                        *       B       NT3
004659  0F59  E0FA
004660                        *
004661                        *       LOC     EXM1
004662        02B2            EXM1 EQU X'02B2'
004663                        *       WAIT
004664                        *       LD      =X'FF'
004665  0F5A  0190
004668                        *       SEND    0
004671  0F5B  FF60
004672                        *       BLBT    EXM
004675  0F5C  E404
004676                        *       WAIT
004677                        *       B       TOP
004678  0F5D  01E0
004681                        *       LOC     EXM
004682        02BB            EXM EQU X'02BB'
004683                        *       LD      24
004684  0F5E  A650
004687                        *       C       =X'F'
004688  0F5F  1892
004691                        *       BET     NSXM
004692  0F60  0FE1
```

```
004695                          *       WAIT
004696  0F61    0501
004697                          *       LD      =X'FF'
004700  0F62    90FF
004701                          *       SEND    0
004704                          *       LOC     NSXM
004705          02C5            NSXM EQU X'02C5'
004706                          *       GNB
004707  0F63    6002
004708                          *       LD      20
004711  0F64    5014
004712                          *       AND     =X'FE'
004715  0F65    93FE
004716                          *       ST      20
004719  0F66    5114
004720                          *       OUT     2
004723                          *       WAIT
004724  0F67    3201
004725                          *       B       EXM
004728  0F68    E0EC
004729                          *       NOP
004730                          *
004731                          *         CHANNEL PROGRAM TO LOOP AND CHECK DSC INTO DSS
004732                          *
004733                          *       ORG     X'200'
004734  0F69    0000
004735          0F6A            CHLP5   EQU     $
004736                          *       LOC     TOP1
004737          0200            TOP1 EQU X'0200'
004738                          *       LD      =7
004741  0F6A    9007
004742                          *       ST      30
004745  0F6B    511E
004746                          *       LD      29
004749  0F6C    501D
004750                          *       C       =X'F0'
004753  0F6D    92F0
004754                          *       BEF     TOP3
004757  0F6E    F109
004758                          *       LD      =16
004761  0F6F    9010
004762                          *       ST      58
004765  0F70    513A
004766                          *       LD      =2
004769  0F71    9002
004770                          *       B       TCMB1
004773  0F72    E00B
004774                          *       LOC     TOP3
004775          0212            TOP3 EQU X'0212'
```

```
004776                          *       C       =X'FC'
004779  0F73  92FC
004780                          *       BET     TOP2
004783  0F74  E105
004784                          *       LD      =8
004787  0F75  9008
004788                          *       ST      58
004791  0F76  513A
004792                          *       LOC     TOP2
004793        021A              TOP2 EQU X'021A'
004794                          *       LD      =0
004797  0F77  9000
004798                          *       LOC     TCMB1
004799        021C              TCMB1 EQU X'021C'
004800                          *       ST      25
004803  0F78  5119
004804                          *       C       =4
004807  0F79  9204
004808                          *       BEF     DSET
004811  0F7A  F109
004812                          *       LD      =X'80'
004815  0F7B  9080
004816                          *       OR      28
004819  0F7C  541C
004820                          *       ST      28
004823  0F7D  511C
004824                          *       LD      =4
004827  0F7E  9004
004828                          *       LOC     DSET
004829        022A              DSET EQU X'022A'
004830                          *       TLU     57
004833  0F7F  5639
004834                          *       OR      28
004837  0F80  541C
004838                          *       OUT     2
004841                          *##
004842                          *#
004843                          *       LD      =175            1 SEC TIME DELAY
004844  0F81  3290
004847                          *       ST      31
004848  0F82  AF51
004851                          *       LOC     LOOP2
004852        0233              LOOP2 EQU X'0233'
004853                          *       LD      =255
004854  0F83  1F90
004857                          *       LOC     LOOP1
004858        0235              LOOP1 EQU X'0235'
004859                          *       DEC
004860  0F84  FF05
```

```
004861                          *       CCH
004862                          *       CCH
004863  0F85  0404
004864                          *       BZF     LOOP1
004867  0F86  F2FC
004868                          *       LD      31
004871  0F87  501F
004872                          *       DEC
004873                          *       ST      31
004874  0F88  0551
004877                          *       BZF     LOOP2
004878  0F89  1FF2
004881                          *#
004882                          *##
004883                          *       IN      5
004886  0F8A  F325
004887                          *       ST
004888                          *       LD      25
004889  0F8B  1150
004892                          *       C       30
004893  0F8C  1952
004896                          *       BET     ENDTC
004897  0F8D  1EE1
004900                          *       XOR     =X'FF'
004901  0F8E  0895
004904                          *       DEC
004905  0F8F  FF05
004906                          *       XOR     =X'FF'
004909  0F90  95FF
004910                          *       B       TCMB1
004913  0F91  E0CD
004914                          *       LOC     ENDTC
004915          0250            ENDTC EQU X'0250'
004916                          *       GNB
004917                          *       WAIT
004918  0F92  0201
004919                          *       B       TOP1
004922  0F93  E0AD
004923                          *       NOP
004924                          *         CHANNEL PROGRAM FOR BROADBAND(RECV)
004925                          *    ADAPTER READY TEST & UNDERRUN OVERRUN TEST
004926                          *       ORG     X'200'
004927          0F94            CHLP6   EQU     $
004928                          *       LD      24                      GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
004929  0F94  0050
004932                          *       OUT     4                       OUTPUT IT
004935  0F95  1834
004936                          *       LD      20
004939  0F96  5014
```

```
004940                          *       OUT     2
004943                          *       WAIT
004944  0F97    3201
004945                          *       LOC     SINK
004946          025C            SINK EQU X'025C'
004947                          *       RECV    0               INPUT 1ST SYNC.
004950                          *       WAIT
004951  0F98    A001
004952                          *       RECV    0               INPUT 2ND SYNC.
004955                          *       C       24
004956  0F99    A052
004959                          *       BET     NSYK
004960  0F9A    18E1
004963                          *       LD      20
004964  0F9B    0B50
004967                          *       AND     =X'FD'
004968  0F9C    1493
004971                          *       OUT     2
004974  0F9D    FD32
004975                          *       LD      20
004978  0F9E    5014
004979                          *       OUT     2
004982                          *       B       SINK
004983  0F9F    32E0
004986                          *       LOC     NSYK
004987          026D            NSYK EQU X'026D'
004988                          *       WAIT
004989  0FA0    F001
004990                          *       RECV    0
004993                          *       WAIT
004994  0FA1    A001
004995                          *       LD      30
004998  0FA2    501E
004999                          *       BZF     UDOR
005002  0FA3    F22D
005003                          *       BARF    RERR
005006  0FA4    F50A
005007                          *       LD      20
005010  0FA5    5014
005011                          *       OUT     2
005014                          *       LOC     BUFE
005015          0279            BUFE EQU X'0279'
005016                          *       RECV    0
005019  0FA6    32A0
005020                          *       ST
005021                          *       BARF    BUFE
005022  0FA7    11F5
005025                          *       BART    CHRL
005026  0FA8    FDE5
```

```
005029                          *       LOC     RERR
005030          027F            RERR EQU X'027F'
005031                          *       LD      =X'FF'
005032  0FA9    0590
005035                          *       ST      61
005036  0FAA    FF51
005039                          *       LOC     CHRL
005040          0283            CHRL EQU X'0283'
005041                          *       LD      =X'33'
005042  0FAB    3D90
005045                          *       ST      62
005046  0FAC    3351
005049                          *       LD      =127
005050  0FAD    3E90
005053                          *       ST      59
005054  0FAE    7F51
005057                          *       LOC     BEPY3
005058          028B            BEPY3 EQU X'028B'
005059                          *       WAIT
005060  0FAF    3B01
005061                          *       RECV    0
005064                          *       ST
005065  0FB0    A011
005066                          *       LD      59
005069  0FB1    503B
005070                          *       DEC
005071                          *       BZT     STP3
005072  0FB2    05E2
005075                          *       ST      59
005076  0FB3    3651
005079                          *       IN      5               CHECK BUFFER EMPTY BIT
005082  0FB4    3B25
005083                          *       AND     =X'04'
005086  0FB5    9304
005087                          *       BZF     BEPY3
005090  0FB6    F2F2
005091                          *       LD      =0
005094  0FB7    9000
005095                          *       ST      62
005098  0FB8    513E
005099                          *       B       STP3
005102  0FB9    E029
005103                          *       LOC     UDOR
005104          02A0            UDOR EQU X'02A0'
005105                          *       C       =X'F'
005108  0FBA    920F
005109                          *       BET     UND
005112  0FBB    E106
005113                          *       RECV    0
```

```
005116                          *       ST
005117  0FBC  A011
005118                          *       WAIT
005119                          *       B       STP3
005120  0FBD  01E0
005123                          *
005124                          *       LOC     UND
005125        02A9              UND EQU X'02A9'
005126                          *       LD      =64
005127  0FBE  2090
005130                          *       LOC     FIFE
005131        02AB              FIFE EQU X'02AB'
005132                          *       ST      59
005133  0FBF  4051
005136                          *       BARF    FIF3
005137  0FC0  3BF5
005140                          *       RECV    0
005143  0FC1  08A0
005144                          *       ST
005145                          *       LD      59
005146  0FC2  1150
005149                          *       DEC
005150  0FC3  3B05
005151                          *       BZT     B1
005154  0FC4  E203
005155                          *       LOC     FIF3
005156        02B6              FIF3 EQU X'02B6'
005157                          *       B       FIFE
005160  0FC5  E0F4
005161                          *       LOC     B1
005162        02B8              B1 EQU X'02B8'
005163                          *       RECV    0
005166                          *       ST
005167  0FC6  A011
005168                          *       IN      5
005171                          *       ST      60
005172  0FC7  2551
005175                          *       LD      20
005176  0FC8  3C50
005179                          *       OR      =1
005180  0FC9  1494
005183                          *       ST      20
005184  0FCA  0151
005187                          *       OUT     2               TURN ON XMIT
005190  0FCB  1432
005191                          *       WAIT
005192                          *       RECV    0
005195  0FCC  01A0
005196                          *       ST
```

```
005197                          *       WAIT
005198  0FCD  1101
005199                          *       LOC     STP3
005200        02C8              STP3 EQU X'02C8'
005201                          *       LD      =0
005204  0FCE  9000
005205                          *       OUT     2
005208                          *       GNB
005209  0FCF  3202
005210                          *       WAIT
005211                          *       NOP
005212  0FD0  0100
005213                          *         CHANNEL PROOGRAM FOR BROADBAND LOOP(XMIT)
005214                          *     ADAPTER READY & UNDERRUN OVERRUN TEST
005215                          *
005216                          *       ORG     X'400'
005217        0FD1              CHLP6A EQU    $
005218                          *       LD      2                       GET LINE CONFIGURATION
005221  0FD1  5002
005222                          *       OUT     6                       OUTPUT IT
005225                          *       LD      24
005226  0FD2  3650
005229                          *       OUT     4
005232  0FD3  1834
005233                          *       WAIT
005234                          *       IN      5                       CHECK BUFFER EMPTY BIT
005237  0FD4  0125
005238                          *       AND     =X'04'
005241  0FD5  9304
005242                          *       BZT     BEPY
005245  0FD6  E205
005246                          *       LD      =X'11'
005249  0FD7  9011
005250                          *       ST      62
005253  0FD8  513E
005254                          *       LOC     BEPY
005255        0410              BEPY EQU X'0410'
005256                          *       LD      =3
005259  0FD9  9003
005260                          *       LOC     XMBB
005261        0412              XMBB EQU X'0412'
005262                          *       ST      59
005265  0FDA  513B
005266                          *       LD      24                      LOAD SYNC. CHAR
005269  0FDB  5018
005270                          *       SEND    0                       SEND IT
005273                          *       LD      59                      DO SIX TIMES
005274  0FDC  6050
005277                          *       DEC
```

```
005278  0FDD  3B05
005279                        *       BZT     SDBB
005282  0FDE  E204
005283                        *       WAIT
005284                        *       B       XMBB
005285  0FDF  01E0
005288                        *       LOC     SDBB
005289        041F            SDBB EQU X'041F'
005290                        *       WAIT
005291  0FE0  F401
005292                        *       LD      30
005295  0FE1  501E
005296                        *       BZF     ORUD
005299  0FE2  F237
005300                        *       LD      20
005303  0FE3  5014
005304                        *       AND     =X'FD'
005307  0FE4  93FD
005308                        *       OUT     2
005311                        *       LD      =58
005312  0FE5  3290
005315                        *       LOC     PGCX
005316        042B            PGCX EQU X'042B'
005317                        *       ST      59
005318  0FE6  3A51
005321                        *       LD
005322  0FE7  3B10
005323                        *       SEND    0
005326                        *       LD      59
005327  0FE8  6050
005330                        *       DEC
005331  0FE9  3B05
005332                        *       BZT     ALD
005335  0FEA  E203
005336                        *       B       PGCX
005339  0FEB  E0F6
005340                        *       LOC     ALD
005341        0436            ALD EQU X'0436'
005342                        *       IN      5               CHECK BUFFER EMPTY BIT
005345                        *       AND     =X'04'
005346  0FEC  2593
005349                        *       BZF     BEPY1
005350  0FED  04F2
005353                        *       LD      =X'22'
005354  0FEE  0590
005357                        *       ST      62
005358  0FEF  2251
005361                        *       LOC     BEPY1
005362        043F            BEPY1 EQU X'043F'
```

```
005363                          *       LD      =70
005364  0FF0    3E90
005367                          *       LOC     AL76
005368          0441            AL76 EQU X'0441'
005369                          *       ST      59
005370  0FF1    4651
005373                          *       LD
005374  0FF2    3B10
005375                          *       SEND    0
005378                          *       LD      59
005379  0FF3    6050
005382                          *       DEC
005383  0FF4    3B05
005384                          *       BZT     AL77
005387  0FF5    E206
005388                          *       BART    AL76
005391  0FF6    E5F6
005392                          *       WAIT
005393                          *       B       TRCV
005394  0FF7    01E0
005397                          *       LOC     AL77
005398          044F            AL77 EQU X'044F'
005399                          *       LD      =X'FF'
005400  0FF8    0590
005403                          *       ST      60
005404  0FF9    FF51
005407                          *       LOC     TRCV
005408          0453            TRCV EQU X'0453'
005409                          *       LD      20
005410  0FFA    3C50
005413                          *       AND     =X'FC'          TURN OFF XMIT AND RECV.
005414  0FFB    1493
005417                          *       OUT     2
005420  0FFC    FC32
005421                          *       B       ALD1
005424  0FFD    E029
005425                          *       LOC     ORUD
005426          045A            ORUD EQU X'045A'
005427                          *       C       =X'F'
005430  0FFE    920F
005431                          *       BEF     OVR
005434  0FFF    F10F
005435                          *       LD      20
005438  1000    5014
005439                          *       AND     =X'FE'
005442  1001    93FE
005443                          *       ST      20
005446  1002    5114
005447                          *       OUT     2               TURN OFF XMIT
```

```
005450                          *       WAIT
005451  1003  3201
005452                          *       LD
005453                          *       SEND    0
005456  1004  1060
005457                          *       GNB
005458                          *       WAIT
005459  1005  0201
005460                          *       B       ALD1
005463  1006  E017
005464                          *
005465                          *       LOC     OVR
005466        046C              OVR EQU X'046C'
005467                          *       LD
005468                          *       SEND    0
005471  1007  1060
005472                          *       BART    OVR
005475  1008  E5FD
005476                          *       LD      =74
005479  1009  904A
005480                          *       LOC     CNT
005481        0472              CNT EQU X'0472'
005482                          *       ST      30
005485  100A  511E
005486                          *       BARF    ROTR
005489  100B  F508
005490                          *       LD
005491                          *       SEND    0
005494  100C  1060
005495                          *       LD      30
005498  100D  501E
005499                          *       BZT     OVCK
005502  100E  E204
005503                          *       DEC
005504                          *       LOC     ROTR
005505        047D              ROTR EQU X'047D'
005506                          *       B       CNT
005507  100F  05E0
005510                          *       LOC     OVCK
005511        047F              OVCK EQU X'047F'
005512                          *       IN      5
005515  1010  F425
005516                          *       ST      60
005519  1011  513C
005520                          *
005521                          *       LOC     ALD1
005522        0482              ALD1 EQU X'0482'
005523                          *       LD      20
005526  1012  5014
```

```
005527                          *       AND     =X'FE'
005530  1013  93FE
005531                          *       ST      20
005534  1014  5114
005535                          *       OUT     2
005538                          *       GNB
005539  1015  3202
005540                          *       LOC     TAMPA
005541        048A              TAMPA EQU X'048A'
005542                          *       WAIT
005543                          *       B       TAMPA
005544  1016  01E0
005547                          *       NOP
005548  1017  FE00
005549                          *
005550                          *         CHANNEL PROGRAM FOR STOP I/O(BROADBAND RECV.)
005551                          *
005552                          *       ORG     X'200'
005553        1018              CHLP7   EQU     $
005554                          *       LD      24                      GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
005557  1018  5018
005558                          *       OUT     4                       OUTPUT IT
005561                          *       LD      20
005562  1019  3450
005565                          *       OUT     2
005568  101A  1432
005569                          *       WAIT
005570                          *       LOC     SINK1
005571        0207              SINK1 EQU X'0207'
005572                          *       RECV    0                       INPUT 1ST SYNC.
005575  101B  01A0
005576                          *       WAIT
005577                          *       RECV    0                       INPUT 2ND SYNC.
005580  101C  01A0
005581                          *       C       24
005584  101D  5218
005585                          *       BET     NSYK1
005588  101E  E10B
005589                          *       LD      20
005592  101F  5014
005593                          *       AND     =X'FD'
005596  1020  93FD
005597                          *       OUT     2
005600                          *       LD      20
005601  1021  3250
005604                          *       OUT     2
005607  1022  1432
005608                          *       B       SINK1
005611  1023  E0F0
```

```
005612                          *       LOC     NSYK1
005613          0218            NSYK1 EQU X'0218'
005614                          *       WAIT
005615                          *       RECV    0
005618  1024  01A0
005619                          *       LOC     LKB1
005620          021A            LKB1 EQU X'021A'
005621                          *       WAIT
005622                          *       RECV    0
005625  1025  01A0
005626                          *       C       =X'DB'
005629  1026  92DB
005630                          *       BEF     LKB1
005633  1027  F1FB
005634                          *       WAIT
005635                          *       LOC     T25A
005636          0221            T25A EQU X'0221'
005637                          *       RECV    0
005640  1028  01A0
005641                          *       ST
005642                          *       BLCT    KER
005643  1029  11E3
005646                          *       BART    T25A
005647  102A  06E5
005650                          *       WAIT
005651  102B  FB01
005652                          *       B       T25A
005655  102C  E0F8
005656                          *       LOC     KER
005657          022A            KER EQU X'022A'
005658                          *       LD      =0
005661  102D  9000
005662                          *       OUT     2
005665                          *       GNB
005666  102E  3202
005667                          *       WAIT
005668                          *       B       SINK1
005669  102F  01E0
005672                          *       NOP
005673  1030  D700
005674                          *
005675                          *
005676                          *         CHANNEL PROGRAM FOR STOP I/O (BROADBAND XMIT)
005677                          *
005678                          *       ORG     X'400'
005679          1031            CHLP7A EQU    $
005680                          *       LD      2                       GET LINE CONFIGURATION
005683  1031  5002
005684                          *       OUT     6                       OUTPUT IT
```

```
005687                          *       LD      =X'AA'
005688  1032  3690
005691                          *       OUT     4
005694  1033  AA34
005695                          *       WAIT
005696                          *       LD      =6
005697  1034  0190
005700                          *       LOC     XMWW
005701        0409              XMWW EQU X'0409'
005702                          *       ST      63
005703  1035  0651
005706                          *       LD      24              LOAD SYNC. CHAR
005707  1036  3F50
005710                          *       SEND    0               SEND IT
005713  1037  1860
005714                          *       LD      63              DO SIX TIMES
005717  1038  503F
005718                          *       DEC
005719                          *       BZT     SDWW
005720  1039  05E2
005723                          *       WAIT
005724  103A  0401
005725                          *       B       XMWW
005728  103B  E0F4
005729                          *       LOC     SDWW
005730        0416              SDWW EQU X'0416'
005731                          *       WAIT
005732                          *       LD      =X'DB'
005733  103C  0190
005736                          *       SEND    0
005739  103D  DB60
005740                          *       WAIT
005741                          *       LOC     T25B
005742        041B              T25B EQU X'041B'
005743                          *       LD
005744  103E  0110
005745                          *       SEND    0
005748                          *       BART    T25B
005749  103F  60E5
005752                          *       WAIT
005753  1040  FD01
005754                          *       LOC     HANG
005755        0420              HANG EQU X'0420'
005756                          *       LD
005757                          *       SEND    0
005760  1041  1060
005761                          *       BLCT    ERR
005764  1042  E304
005765                          *       WAIT
```

```
005766                          *       B       HANG
005767  1043  01E0
005770                          *       LOC     ERR
005771        0427              ERR EQU X'0427'
005772                          *       WAIT
005773  1044  FA01
005774                          *       LD      20
005777  1045  5014
005778                          *       AND     =X'FE'
005781  1046  93FE
005782                          *       OUT     2
005785                          *       GNB
005786  1047  3202
005787                          *       WAIT
005788                          *       B       T25B
005789  1048  01E0
005792                          *       NOP
005793  1049  EB00
005794                          *
005795                          *          CHANNEL PROGRAM FOR CHARACTER SIZE AND
005796                          *           CRC, PARITY TESTS (BROADBAND RECV.)
005797                          *
005798                          *       ORG     X'200'
005799        104A              CHLP8   EQU     $
005800                          *       LD      24
005803  104A  5018
005804                          *       OUT     4                       OUTPUT SYNC. CHARACTER
005807                          *       LD      20
005808  104B  3450
005811                          *       OUT     2                       OUTPUT LINE CONTROL
005814  104C  1432
005815                          *       WAIT
005816                          *       LOC     SYCH                    GET IN RECV. IN SYNC.
005817        0207              SYCH EQU X'0207'
005818                          *       RECV    0
005821  104D  01A0
005822                          *       BART    SCY2
005825  104E  E502
005826                          *       WAIT
005827                          *       LOC     SCY2
005828        020B              SCY2 EQU X'020B'
005829                          *       RECV    0
005832  104F  01A0
005833                          *       C       28
005836  1050  521C
005837                          *       BET     SYC
005840  1051  E104
005841                          *       SFS
005842                          *       B       SYCH
```

```
005843  1052  03E0
005846                        *       LOC     SYC
005847        0213            SYC EQU X'0213'
005848                        *       BART    SCY1
005849  1053  F5E5
005852                        *       WAIT
005853  1054  0201
005854                        *       LOC     SCY1
005855        0216            SCY1 EQU X'0216'
005856                        *       RECV    0
005859                        *       C       =X'19'          CHECK FOR START OF MESSAGE SHAR.
005860  1055  A092
005863                        *       BEF     SYC
005864  1056  19F1
005867                        *       LD      24
005868  1057  F950
005871                        *       C       =X'32'
005872  1058  1892
005875                        *       BET     CPRT            IF EQUAL GO TO CRC AND PARITY ROUTINES
005876  1059  32E1
005879                        *       LD      2               START OF CHAR. SIZE TEST
005880  105A  3550
005883                        *       BZT     FVBC            BRANCH TO FIVE BIT CHAR. OUTPUT LOOP
005884  105B  02E2
005887                        *       C       =X'40'
005888  105C  1E92
005891                        *       BEF     NFBA
005892  105D  40F1
005895                        *       LOC     SXBC
005896        0229            SXBC EQU X'0229'
005897                        *       BART    SXB1
005898  105E  0EE5
005901                        *       WAIT
005902  105F  0201
005903                        *       LOC     SXB1
005904        022C            SXB1 EQU X'022C'
005905                        *       RECV    2
005908                        *       C       30
005909  1060  A252
005912                        *       BET     SXBC
005913  1061  1EE1
005916                        *       ST
005917  1062  F911
005918                        *       BLCT    EXT
005921  1063  E347
005922                        *       B       SXBC
005925  1064  E0F4
005926                        *       LOC     NFBA
005927        0236            NFBA EQU X'0236'
```

```
005928                          *       BART    NFVB
005931  1065    E502
005932                          *       WAIT
005933                          *       LOC     NFVB
005934          0239            NFVB EQU X'0239'
005935                          *       RECV    2
005938  1066    01A2
005939                          *       ST
005940                          *       BLCT    EXT
005941  1067    11E3
005944                          *       B       NFBA
005945  1068    3EE0
005948                          *       LOC     FVBY
005949          023F            FVBY EQU X'023F'
005950                          *       BART    FVBC
005951  1069    F8E5
005954                          *       WAIT
005955  106A    0201
005956                          *       LOC     FVBC
005957          0242            FVBC EQU X'0242'
005958                          *       RECV    0
005961                          *       C       30
005962  106B    A052
005965                          *       BET     FVBY
005966  106C    1EE1
005969                          *       ST
005970  106D    F911
005971                          *       C       =X'1E'
005974  106E    921E
005975                          *       BET     TKF
005978  106F    E105
005979                          *       LOC     TKFN
005980          024C            TKFN EQU X'024C'
005981                          *       BLCT    EXT
005984  1070    E32D
005985                          *       B       FVBY
005988  1071    E0F0
005989                          *       LOC     TKF
005990          0250            TKF EQU X'0250'
005991                          *       LD      30
005994  1072    501E
005995                          *       ST
005996                          *       B       TKFN
005997  1073    11E0
006000                          *       LOC     CPRT
006001          0255            CPRT EQU X'0255'
006002                          *       LD      23
006003  1074    F850
006006                          *       C       =1
```

```
006007  1075  1792
006010                        *       BET     RE1A
006011  1076  01E1
006014                        *       C       =X'F0'
006015  1077  0E92
006018                        *       BET     RE2A
006019  1078  F0E1
006022                        *       LOC     RE3A
006023        025F            RE3A EQU X'025F'
006024                        *       BART    RE3
006025  1079  13E5
006028                        *       WAIT
006029  107A  02C1
006030                        *       LOC     RE3
006031        0262            RE3 EQU X'0262'
006032                        *       RECV    3
006035                        *       ST
006036  107B  A311
006037                        *       BLCT    EXT
006040  107C  E315
006041                        *       B       RE3A
006044  107D  E0F8
006045                        *       LOC     RE1A
006046        0268            RE1A EQU X'0268'
006047                        *       BART    RE1
006050  107E  E502
006051                        *       WAIT
006052                        *       LOC     RE1
006053        026B            RE1 EQU X'026B'
006054                        *       RECV    1
006057  107F  01A1
006058                        *       ST
006059                        *       BLCT    EXT
006060  1080  11E3
006063                        *       B       RE1A
006064  1081  0CE0
006067                        *       LOC     RE2A
006068        0271            RE2A EQU X'0271'
006069                        *       BART    RE2
006070  1082  F8E5
006073                        *       WAIT
006074  1083  0201
006075                        *       LOC     RE2
006076        0274            RE2 EQU X'0274'
006077                        *       RECV    2
006080                        *       ST
006081  1084  A211
006082                        *       BLCT    EXT
006085  1085  E303
```

```
006086                          *       B       RE2A
006089  1086  E0F8
006090                          *       LOC     EXT
006091        027A              EXT EQU X'027A'
006092                          *       LD      =0
006095  1087  9000
006096                          *       OUT     2
006099                          *       GNB
006100  1088  3202
006101                          *       WAIT
006102                          *       NOP
006103  1089  0100
006104                          *
006105                          *           CHANNEL PROGRAM FOR CHARACTER SIZE AND
006106                          *            CRC,PARITY TESTS (BROADBAND XMIT)
006107                          *
006108                          *       ORG     X'400'
006109        108A              CHLP8A EQU    $
006110                          *       LD      2                       GET LINE CONFIGURATION
006113  108A  5002
006114                          *       OUT     6                       OUTPUT IT
006117                          *       LD      =X'FF'
006118  108B  3690
006121                          *       OUT     4
006124  108C  FF34
006125                          *       ST      28
006128  108D  511C
006129                          *       WAIT
006130                          *       LD      24
006131  108E  0150
006134                          *       C       =X'32'
006135  108F  1892
006138                          *       BET     Z48
006139  1090  32E1
006142                          *       LD      =X'1F'
006143  1091  3290
006146                          *       ST      27
006147  1092  1F51
006150                          *       ST      28
006151  1093  1B51
006154                          *       LD      2
006155  1094  1C50
006158                          *       BZT     Z47
006159  1095  02E2
006162                          *       C       =X'40'
006163  1096  2292
006166                          *       BEF     N611
006167  1097  40F1
006170                          *       LD      =X'3F'
```

```
006171  1098  0A90
006174                        *       ST      28
006175  1099  3F51
006178                        *       SR
006179  109A  1C07
006180                        *       ST      27
006183  109B  511B
006184                        *       B       Z47
006187  109C  E015
006188                        *       LOC     N611
006189        0426            N611 EQU X'0426'
006190                        *       C       =X'80'
006193  109D  9280
006194                        *       BEF     N711
006197  109E  F10A
006198                        *       LD      =X'7F'
006201  109F  907F
006202                        *       ST      28
006205  10A0  511C
006206                        *       SR
006207                        *       ST      27
006208  10A1  0751
006211                        *       B       Z47
006212  10A2  1BE0
006215                        *       LOC     N711
006216        0433            N711 EQU X'0433'
006217                        *       LD      =X'FF'
006218  10A3  0890
006221                        *       ST      28
006222  10A4  FF51
006225                        *       SR
006226  10A5  1C07
006227                        *       ST      27
006230  10A6  511B
006231                        *       LOC     Z47
006232        043A            Z47 EQU X'043A'
006233                        *       LD      28
006236  10A7  501C
006237                        *       AND     =X'FF'
006240  10A8  93FF
006241                        *       ST      30
006244  10A9  511E
006245                        *       LOC     Z48
006246        0440            Z48 EQU X'0440'
006247                        *       LD      28
006250  10AA  501C
006251                        *       AND     24
006254  10AB  5318
006255                        *       ST      28
```

```
006258  10AC  511C
006259                        *       LD      =6
006262  10AD  9006
006263                        *       LOC     Z2Z
006264        0448            Z2Z EQU X'0448'
006265                        *       ST      63
006268  10AE  513F
006269                        *       LD      24
006272  10AF  5018
006273                        *       SEND    0
006276                        *       LD      63
006277  10B0  6050
006280                        *       DEC
006281  10B1  3F05
006282                        *       BZT     Z3Z
006285  10B2  E203
006286                        *       BART    Z2Z
006289  10B3  E5F5
006290                        *       LOC     Z3Z
006291        0454            Z3Z EQU X'0454'
006292                        *       LD      =X'19'
006295  10B4  9019
006296                        *       SEND    0
006299                        *       LD      24
006300  10B5  6050
006303                        *       C       =X'32'
006304  10B6  1892
006307                        *       BET     CPRTX
006308  10B7  32E1
006311                        *       LD      2
006312  10B8  1B50
006315                        *       BZT     ZXZ1
006316  10B9  02E2
006319                        *       LOC     XPYZ
006320        0461            XPYZ EQU X'0461'
006321                        *       LD
006322  10BA  0C10
006323                        *       AND     27
006326  10BB  531B
006327                        *       SEND    2
006330                        *       BLCT    ZEXT
006331  10BC  62E3
006334                        *       BART    XPYZ
006335  10BD  36E5
006338                        *       WAIT
006339  10BE  F901
006340                        *       B       XPYZ
006343  10BF  E0F6
006344                        *       LOC     ZXZ1
```

```
006345          046C                ZXZ1 EQU X'046C'
006346                              *       LD
006347                              *       AND     27
006348  10C0    1053
006351                              *       SEND    0
006354  10C1    1B60
006355                              *       BLCT    ZEXT
006358  10C2    E32B
006359                              *       BART    ZXZ1
006362  10C3    E5F9
006363                              *       WAIT
006364                              *       B       ZXZ1
006365  10C4    01E0
006368                              *       LOC     CPRTX
006369          0477                CPRTX EQU X'0477'
006370                              *       LD      23
006371  10C5    F650
006374                              *       C       =1
006375  10C6    1792
006378                              *       BET     BBCR
006379  10C7    01E1
006382                              *       C       =X'F0'
006383  10C8    0E92
006386                              *       BET     BBPR
006387  10C9    F0E1
006390                              *       LOC     BBPCR
006391          0481                BBPCR EQU X'0481'
006392                              *       LD
006393  10CA    1310
006394                              *       SEND    3
006397                              *       BLCT    ZEXT
006398  10CB    63E3
006401                              *       BART    BBPCR
006402  10CC    18E5
006405                              *       WAIT
006406  10CD    FB01
006407                              *       B       BBPCR
006410  10CE    E0F8
006411                              *       LOC     BBCR
006412          048A                BBCR EQU X'048A'
006413                              *       LD
006414                              *       SEND    1
006417  10CF    1061
006418                              *       BLCT    ZEXT
006421  10D0    E30F
006422                              *       BART    BBCR
006425  10D1    E5FB
006426                              *       WAIT
006427                              *       B       BBCR
```

```
006428  10D2  01E0
006431                        *       LOC     BBPR
006432        0493            BBPR EQU X'0493'
006433                        *       LD
006434  10D3  F810
006435                        *       SEND    2
006438                        *       BLCT    ZEXT
006439  10D4  62E3
006442                        *       BART    BBPR
006443  10D5  06E5
006446                        *       WAIT
006447  10D6  FB01
006448                        *       B       BBPR
006451  10D7  E0F8
006452                        *       LOC     ZEXT
006453        049C            ZEXT EQU X'049C'
006454                        *       LD      =0
006457  10D8  9000
006458                        *       SEND    0
006461                        *       SEND    0
006464  10D9  6060
006465                        *       LD      20
006468  10DA  5014
006469                        *       AND     =X'FE'
006472  10DB  93FE
006473                        *       OUT     2
006476                        *       GNB
006477  10DC  3202
006478                        *       WAIT
006479                        *       NOP
006480  10DD  0100
006481                        *         CHANNEL PROGRAM FOR MARK & SPACE (REC)
006482                        *
006483                        *       ORG     X'200'
006484        10DE            CHLP9   EQU     $
006485                        *       LD      2                       GET LINE CONFIG.
006488  10DE  5002
006489                        *       OUT     6                       OUTPUT IT
006492                        *       LD      =X'D'                   LOAD LINE SPEED
006493  10DF  3690
006496                        *       OUT     4                       OUTPUT IT
006499  10E0  0D34
006500                        *       LD      20                      GET LINE CONTROL
006503  10E1  5014
006504                        *       OUT     2
006507                        *       WAIT
006508  10E2  32C1
006509                        *       RECV    0
006512                        *       ST
```

```
006513  10E3  A011
006514                        *       WAIT
006515                        *       LOC     RAN
006516        020D            RAN EQU X'020D'
006517                        *       RECV    0
006520  10E4  01A0
006521                        *       ST
006522                        *       BZF     BINTA
006523  10E5  11F2
006526                        *       IN      5
006529  10E6  0A25
006530                        *       AND     =1
006533  10E7  9301
006534                        *       BZT     BINTA
006537  10E8  E205
006538                        *       LD      =X'FE'
006541  10E9  90FE
006542                        *       ST      60
006545  10EA  513C
006546                        *       LOC     BINTA
006547        021A            BINTA EQU X'021A'
006548                        *       BLCT    TONI
006551  10EB  E304
006552                        *       WAIT
006553                        *       B       RAN
006554  10EC  01E0
006557                        *       LOC     TONI
006558        021F            TONI EQU X'021F'
006559                        *       LD      =0
006560  10ED  EF90
006563                        *       OUT     2
006566  10EE  0032
006567                        *       GNB
006568                        *       WAIT
006569  10EF  0201
006570                        *       NOP
006571                        *
006572                        *         CHANNEL PROGRAM FOR MARK & SPACE (XMIT)
006573                        *
006574                        *       ORG     X'400'
006575  10F0  0000
006576        10F1            CHLP9A EQU    $
006577                        *       LOC     CLRK
006578        0400            CLRK EQU X'0400'
006579                        *       LD      GET             CHARACTER  TO SEND OUT
006580                        *       C       =5              CHECK TO SET MARK AND SPACE CONDITION
006581  10F1  1092
006584                        *       BEF     DENW
006585  10F2  05F1
```

```
006588                          *       LD      20              SET MARK OR SPACE CONDITION
006589  10F3    0A50
006592                          *       OR      26
006593  10F4    1454
006596                          *       ST      20
006597  10F5    1A51
006600                          *       OUT     2
006603  10F6    1432
006604                          *       LD      =5
006607  10F7    9005
006608                          *       LOC     DENW
006609          040E            DENW EQU X'040E'
006610                          *       C       =7              CHECK  TO RESET MARK AND SPACE CONDITION
006613  10F8    9207
006614                          *       BEF     BAMB
006617  10F9    F119
006618                          *       LD      20              RESET MARK OR SPACE CONDITION
006621  10FA    5014
006622                          *       XOR     26
006625  10FB    551A
006626                          *       ST      20
006629  10FC    5114
006630                          *       OUT     2
006633                          *       LD      26
006634  10FD    3250
006637                          *       C       =X'10'          IF SPACE CONDITION CHECK FOR FRAMING ERROR
006638  10FE    1A92
006641                          *       BEF     KUNTA
006642  10FF    10F1
006645                          *       IN      5
006648  1100    0A25
006649                          *       AND     =1              MASK OTHER BITS OUT
006652  1101    9301
006653                          *       BZT     KUNTA           BRANCH ON FRAMING ERROR SET
006656  1102    E205
006657                          *       LD      =X'EE'
006660  1103    90EE
006661                          *       ST      61
006664  1104    513D
006665                          *       LOC     KUNTA
006666          0428            KUNTA EQU X'0428'
006667                          *       LD      =7
006670  1105    9007
006671                          *       LOC     BAMB
006672          042A            BAMB EQU X'042A'
006673                          *       SEND    0
006676                          *       BLCT    RAY
006677  1106    60E3
006680                          *       WAIT
```

```
006681  1107  0401
006682                        *       B       CLRK
006685  1108  E0D1
006686                        *       LOC     RAY
006687        0430            RAY EQU X'0430'
006688                        *       WAIT
006689                        *       LD      =X'FF'
006690  1109  0190
006693                        *       SEND    0
006696  110A  FF60
006697                        *       GNB
006698                        *       LD      20
006699  110B  0250
006702                        *       AND     =X'FE'
006703  110C  1493
006706                        *       ST      20
006707  110D  FE51
006710                        *       WAIT
006711  110E  1401
006712                        *       NOP
006713        110F            CHLP10 EQU    $
006714                        *
006715                        *     ECHO BACK CCP &LCT TABLE
006716                        *
006717        110F            LOPCCP EQU    $
006718                        *       ORG     X'200'
006719  110F  0000
006720                        *
006721                        *
006722                        *     RECEIVE CCP
006723                        *
006724                        *
006725                        *       LD      24
006728  1110  5018
006729                        *       OUT     4               OUTPUT SYNC CHARACTER
006732                        *       LD      2
006733  1111  3450
006736                        *       OUT     6               OUTPUT CHARACTER LENGTH
006739  1112  0236
006740                        *       LD      =3              SET RECEIVER & XMITTER ON
006743  1113  9003
006744                        *       ST      20
006747  1114  5114
006748                        *       OUT     2
006751                        *       LOC     LOPWT
006752        020B            LOPWT EQU X'020B'
006753                        *       WAIT
006754  1115  3201
006755                        *       LD      17
```

```
006758 1116 5011
006759                  *       AND     =X'10'
006762 1117 9310
006763                  *       BZF     LOPDSS          BRANCH IF DATA SET STATUS CHANGE
006766 1118 F206
006767                  *       IN      1               GET & STORE RECEIVED CHARACTER
006770                  *       ST      24
006771 1119 2151
006774                  *       B       LOPWT
006775 111A 18E0
006778                  *       LOC     LOPDSS
006779      0217        LOPDSS  EQU X'0217'
006780                  *       LD      =0              DATA SET STATUS CHANGE PROCESSING
006781 111B F590
006784                  *       ST      17              CLEAR DATA SET STATUS CHANGE FLAG
006785 111C 0051
006788                  *       LD      =X'3A'          SET UP SYNC CHARACTER FOR XMIT
006789 111D 1190
006792                  *       ST      24
006793 111E 3A51
006796                  *       SFS     RESYNC          RECEIVER
006797 111F 1803
006798                  *       LD      =3              ECHO BACK DATA SET STATUS
006801 1120 9003
006802                  *       ST      25
006805 1121 5119
006806                  *       LD      14
006809 1122 500E
006810                  *       AND     =X'80'
006813 1123 9380
006814                  *       OR      25              SET UP DATA TERMINAL READY
006817 1124 5419
006818                  *       ST      25
006821 1125 5119
006822                  *       LD      14
006825 1126 500E
006826                  *       AND     =X'20'
006829 1127 9320
006830                  *       BZT     LOPDSO
006833 1128 E207
006834                  *       LD      =X'40'
006837 1129 9040
006838                  *       OR      25              SET REQUEST TO SEND
006841 112A 5419
006842                  *       ST      25
006845 112B 5119
006846                  *       LOC     LOPDSO
006847      0238        LOPDSO  EQU X'0238'
006848                  *       LD      25              OUTPUT NEW DATA SET STATUS
```

```
006851  112C  5019
006852                        *       ST      20
006855  112D  5114
006856                        *       OUT     2
006859                        *       B       LOPWT
006860  112E  32E0
006863                        *
006864                        *
006865                        *     XMIT CCP
006866                        *
006867                        *
006868                        *       LOC     LOPXMT
006869        023F            LOPXMT  EQU X'023F'
006870                        *       LD      24                      GET CHARACTER TO SEND
006871  112F  CD50
006874                        *       OUT     1
006877  1130  1831
006878                        *       WAIT
006879                        *       B       LOPXMT
006880  1131  01E0
006885  1132  FB00
006886        1133            LOPCCE  EQU     $
006887  1133  0048            LOPSZE  DC      (LOPCCE-LOPCCP)*2       CCP SIZE (BYTES)
```

```
006888                          /
006889                          *
006890                          *
006891                          *     LCT TABLE
006892                          *
006893                          *
006894  1134  0206              LOPLCT DC    • X'206'              RECEIVE CCP START ADDRESS
006895  1135  0226                     DC      AND(LOPXMT,Z'FF00')+X'0026' XMIT CCP START ADDRESS
006896  1136  3F27
006897  1137  C002                     DC      Z'C002'               8 BIT CHARACTERS
006898  1138  C022                     DC      Z'C022'
006899  1139  D008                     DC      Z'D008'               START UP RECEIVE CCP ON DATA SET STATUS
006900  113A  A00F                     DC      Z'A00F'               DSS MASK
006901  113B  3A18                     DC      Z'3A18'               SYNC CHARACTER (LCT 24)
006902  113C  0000                     DC      0                     END OF TABLE
006903                          *=============================================================================
006904                          *                        I/O FUNCTION CODES
006905                          *
006906  113D  0001              FOMC   DC      X'1'                  OUTPUT MLCP CONTROL
006907  113E  0003              FOIC   DC      X'3'                  OUTPUT INTERRUPT CONTROL
006908  113F  0005              FOCC   DC      X'5'                  OUTPUT CHANNEL CONTROL
006909  1140  0009              FOLD   DC      X'9'                  OUTPUT ADDRESS AND RANGE
006910  1141  000B              FOBL   DC      X'B'                  OUTPUT BYTE INTO LCT
006911  1142  000F              FOCB   DC      X'F'                  OUTPUT CCB CONTROL
006912  1143  000C              FIR    DC      X'C'                  INPUT RANGE
006913  1144  0018              FIS    DC      X'18'                 INPUT STATUS
006914  1145  001A              FINS   DC      X'1A'                 INPUT NEXT STATUS
006915  1146  001C              FIDSS  DC      X'1C'                 INPUT DATA SET STATUS
006916  1147  001E              FILS   DC      X'1E'                 INPUT LCT ATATUS
006917  1148  0026              FID    DC      X'26'                 INPUT ID
006918                          *
006919                          *=============================================================================
006920                          *                        STORAGE CONSTANTS
006921                          *
006922                          *##
006923                          *#
006924  1149  0000              TEST   DC      0                     TEST COMMAND CODE SAVE AREA
006925  114A  0000              MLXCT  DC      0                     MODEM LOOPBACK XMIT BLOCK CTR
006926  114B  0000              MLECT  DC      0                     MODEM LOOPBACK XMIT ERROR CTR
006927  114C  0000              COUNT1 DC      0                     DATA BLOCK COUNTER
006928                          *#
006929                          *##
006930  114D  0000              ARFG   DC      0                     ADAPTER READY TEST FLAG
006931  114E  0000              BYTP   DC      0                     BYTE OFFSET FOR V$CALLS
006932  114F  0000              CHANL  DC      0
006933  1150  0000              CHSZFG DC      0                     CHARACTER SIZE TEST FLAG
006934  1151  0000              CHNZ   DC      0
006935  1152  0000              COUNT  DC      0
```

```
006936  1153  0000              CS      DC      0                       CHARACTER SIZE
006937  1154  0000              CSB     DC      0                       CHARACTER SIZE INCLUDING START AND STOP BITS
006938  1155  0000              CRCAC   DC      0                       CRC ACUMMULATOR
006939  1156  0000              CRCFG   DC      0
006940  1157  0000              DIV1    DC      0                       DIVISORS FOR TICKS TO MILLSEC SUB-ROUTINE
006941  1158  0000              DIV2    DC      0
006942  1159  0000              DIV3    DC      0
006943  115A  0000              DIV4    DC      0
006944  115B  0034              ERF     DC      52
006945  115C  0000              FBRG    DC      0
006946  115D  0000              FWFG    DC      0                       FIRMWARE PRINT FLAG
006947  115E  0000              HGSPD   DC      0                       HIGH SPEED FLAG
006948  115F  003C              HRTZ    DC      60                      LINE FREQUENCY
006949  1160  0000              MASK    DC      0
006950  1161  0000              MSBS    RESV    2,0                     BITS/SEC
006951  1163  0000              PARFG   DC      0                       PARITY ERROR TEST FLAG
006952  1164  0000              PTTY    DC      0                       PARITY TYPE FLAG
006953  1165  0000              PSFG    DC      0                       SYNC. PRINT SPEED FLAG
006954  1166  0000              RANGE   DC      0
006955  1167  0000              RCRC    DC      0                       RECEIVE CRC RESIDUE
006956  1168  0000              RLCTAD  DC      0                       RAM ADDRESS TO READ
006957  1169  0000              RLCTRG  DC      0                       RANGE FOR READ FROM RAM
006958  116A  0000              RUNFG   DC      0                       OVERRUN AND UNDERRUN TEST FLAG
006959  116B  0000              TLOC    DC      0                       TEMPORARY LOCATION
006960  116C  0000              TP1     DC      0
006961  116D  0000              TP2     DC      0
006962  116E  0000              SBTFG   DC      0                       STOP BIT TIME TEST FLAG
006963  116F  0000              SHORT   DC      0                       TEST FLAG
006964  1170  0000              STPFG   DC      0                       STOP I/O AND CHANNEL RESET FLAG
006965  1171  0000              WDBFG   DC      0                       BROADBAND TEST FLAG
006966  1172  0000              XCRC    DC      0                       TRANSMIT CRC RESIDUE
006967  1173  FFFF              LCTV    RESV    3,-1
006968  1176  0000              ERAR    RESV    4,0                     ERROR ARRAY FOR ZV$C
006969                          *
006970  117A                    SAV1    RESV    9+7*$AF
006971  118A                    SAV2    RESV    9+7*$AF
006972  119A                    SAV3    RESV    9+7*$AF
006973  11AA                    SAV4    RESV    9+7*$AF
006974                          *
006975                          *=============================================================================
006976                          *
006977                          *                       DEFINED CONSTANTS
006978                          *
006979  11BA  2108              ASYN    DC      X'2108'                 ASYNC. ID
006980  11BB  2118              ASYN1   DC      X'2118'                 ASYNC. ID NEW BAUD RATE
006981  11BC  2110              ASYNC   DC      X'2110'                 CURRENT LOOP ASYNC.
006982  11BD  2158              SYNC    DC      X'2158'                 SYNC.  ID
006983  11BE  2150              BISYNC  DC      X'2150'                 BISYNC. ID
006984  11BF  2160              SYN188  DC      X'2160'                 SYNC. MIL 188 ID
```

```
006985  11C0  2138                WDB3    DC      X'2138'             WIDEBAND WITH 301/303
006986  11C1  2130                WDB3B   DC      X'2130'             BI-SYNC. WIDEBAND WITH 301/303
006987  11C2  2168                WDBV35  DC      X'2168'             WIDEBAND WITH V35
006988  11C3  2128                WDBV5B  DC      X'2128'             BI-SYNC. WIDEBAND WITH V35
006989                            *
006990  11C4  8000                BIT0    DC      Z'8000'
006991  11C5  4000                BIT1    DC      Z'4000'
006992  11C6  2000                BIT2    DC      X'2000'
006993  11C7  0800                BIT4    DC      X'800'
006994  11C8  0200                BIT6    DC      X'200'
006995  11C9  0100                BIT7    DC      X'100'
006996                            *
006997  11CA  A001                CRC16   DC      Z'A001'
006998                            *
006999  11CB  0000                ATLT    RESV    8,0                 ACTIVE LINE TABLE
007000                            *
007001  11D3  0000                ADDRS   RESV    9,0                 LINE ADDRESS TABLE
007002                            *
007003  11DC  0117                CPFLG   DC      X'117'              LCT FLAG FOR CRC ONLY
007004  11DD  F017                        DC      Z'F017'             PARITY ONLY
007005  11DE  0017                        DC      X'017'              CRC AND PARITY
007006  11DF  0000                        DC      0
007007                            *
007008  11E0  805C                CRLF    DC      Z'805C'
007009  11E1  2C00                COMMA   DC      X'2C00'
007010  11E2  005C                BKLS    DC      X'5C'
007011                            *
007012  11E3  0001 0203 0405      DOUT    TEXT    Z'000102030405060708090A0B0C0D0E0F'
        11E6  0607 0809 0A0B
              0C0D 0E0F
007013  11E8  1011 1213 1415              TEXT    Z'101112131415161718191A1B1C1D1E1F'
        11EE  1617 1819 1A1B
              1C1D 1E1F
007014  11F3  2021 2223 2425      DOUT1   TEXT    Z'202122232425262728292A2B2C2D2E2F'
        11F6  2627 2829 2A2B
              2C2D 2E2F
007015  11F8  3031 3233 3435              TEXT    Z'303132333435363738393A3B3C3D3E3F'
        11FE  3637 3839 3A3B
              3C3D 3E3F
007016  1203  4041 4243 4445              TEXT    Z'404142434445464748494A4B4C4D4E4F'
        1206  4647 4849 4A4B
              4C4D 4E4F
007017  120B  5051 5253 5455              TEXT    Z'505152535455565758595A5B5C5D5E5F'
        120E  5657 5859 5A5B
              5C5D 5E5F
007018  1213  6061 6263 6465              TEXT    Z'606162636465666768696A6B6C6D6E6F'
        1216  6667 6869 6A6B
              6C6D 6E6F
007019  1218  7071 7273 7475              TEXT    Z'707172737475767778797A7B7C7D'
```

```
         121E  7677 7879 7A7B
               7C7D
007020   1222  5555            DFLT   DC     X'5555'          DEFAULT CHARACTER
007021                         *
007022   1223  0000            FIVSEC DC     0                5 SECONDS OF TICK ON RTC
007023   1224  0000            SECL   DC     0
007024   1225  0000            SECH   DC     0
007025                         *
007026   1226  0000            FRIMRV DC     0                FIRMWARE REV NUMBER
007027                         *
007028                         *    LINE SPEED IS IN SECOND WORD FOR LOW AND MEDIUM
007029                         *    SPEED LINES.  FOR HIGH SPEED LINES THE TWO
007030                         *    WORDS ARE PUT TOGETHER FORMING A SINGLE HEX WORD
007031                         *    CONSISTING OF 32 BITS.
007032                         *
007033   1227  0000            SYLSP0 RESV   2,0              SYNC. LINE 0 SPEED
007034   1229  0000            SYLSP1 RESV   2,0              SYNC. LINE 1 SPEED
007035   122B  0000            SYLSP2 RESV   2,0              SYNC. LINE 2 SPEED
007036   122D  0000            SYLSP3 RESV   2,0              SYNC. LINE 3 SPEED
007037   122F  0000            SYLSP4 RESV   2,0              SYNC. LINE 4 SPEED
007038   1231  0000            SYLSP5 RESV   2,0              SYNC. LINE 5 SPEED
007039   1233  0000            SYLSP6 RESV   2,0              SYNC. LINE 6 SPEED
007040   1235  0000            SYLSP7 RESV   2,0              SYNC. LINE 7 SPEED
007041                         *
007042   1237  000A            NSPTBL DC     10
007043   1238  000F                   DC     15
007044   1239  0016                   DC     22
007045   123A  001B                   DC     27
007046   123B  001E                   DC     30
007047   123C  0028                   DC     40
007048   123D  003C                   DC     60
007049   123E  0078                   DC     120
007050   123F  00D2                   DC     210
007051   1240  00F0                   DC     240
007052   1241  0168                   DC     360
007053   1242  0190                   DC     400
007054   1243  01E0                   DC     480
007055   1244  03C0                   DC     960
007056   1245  0780                   DC     1920
007057   1246  0F00                   DC     3840
007058                         *
007059   1247  000A            OSPTBL DC     10
007060   1248  000F                   DC     15
007061   1249  0016                   DC     22
007062   124A  001B                   DC     27
007063   124B  001E                   DC     30
007064   124C  003C                   DC     60
007065   124D  0078                   DC     120
007066   124E  00B4                   DC     180
```

```
007067  124F  00F0                    DC    240
007068  1250  0168                    DC    360
007069  1251  01E0                    DC    480
007070  1252  02D0                    DC    720
007071  1253  03C0                    DC    960
007072  1254  05A0                    DC    1440
007073  1255  0780                    DC    1920
007074  1256  0000                    DC    0
007075                        *
007076  1257  0032            ALSPBL  DC    50          OLD BAUD RATE LINE SPEED TABLE
007077  1258  004B                    DC    75
007078  1259  006E                    DC    110
007079  125A  0087                    DC    135
007080  125B  0096                    DC    150
007081  125C  012C                    DC    300
007082  125D  0258                    DC    600
007083  125E  0384                    DC    900
007084  125F  04B0                    DC    1200
007085  1260  0708                    DC    1800
007086  1261  0960                    DC    2400
007087  1262  0E10                    DC    3600
007088  1263  12C0                    DC    4800
007089  1264  1C20                    DC    7200
007090  1265  2580                    DC    9600
007091                        *
007092  1266  0032            CMSPBL  DC    50          NEW BAUD RATE LINE SPEED TABLE
007093  1267  004B                    DC    75
007094  1268  006E                    DC    110
007095  1269  0087                    DC    135
007096  126A  0096                    DC    150
007097  126B  00C8                    DC    200
007098  126C  012C                    DC    300
007099  126D  0258                    DC    600
007100  126E  041A                    DC    1050
007101  126F  04B0                    DC    1200
007102  1270  0708                    DC    1800
007103  1271  07D0                    DC    2000
007104  1272  0960                    DC    2400
007105  1273  12C0                    DC    4800
007106  1274  2580                    DC    9600
007107  1275  4B00                    DC    19200
007108                        *
007109  1276  0021            SBITBL  DC    33
007110  1277  001E                    DC    30
007111  1278  0020                    DC    32
007112  1279  001E                    DC    30
007113                        *
007114  127A  0018            TLTBLA  DC    X'18'       SHOULD BE TABLE FOR DSS TO DSC LOOP ASYC.
007115  127B  6078                    DC    X'6078'
```

```
007116  127C  8098                    DC      Z'8098'
007117  127D  E0F8                    DC      Z'E0F8'
007118                          *
007119  127E  B0B0            TLTBLC  DC      Z'B0B0'         SHOULD BE TABLE FOR CURRENT LOOP ADAPT.
007120  127F  F0F0                    DC      Z'F0F0'
007121  1280  B0B0                    DC      Z'B0B0'
007122  1281  F0F0                    DC      Z'F0F0'
007123                          *
007124  1282  0010            TLTBLS  DC      X'10'           SHOULD BE TABLE FOR DSS TO DSC LOOP SYNC.
007125  1283  6070                    DC      X'6070'
007126  1284  8090                    DC      Z'8090'
007127  1285  E0F0                    DC      Z'E0F0'
007128                          *##
007129                          *#
007130  1286  2020            TLTBLM  DC      X'2020'         SHOULD BE TABLE FOR MODEM DSS TO DSC LOO
007131  1287  2020                    DC      X'2020'
007132  1288  A0A0                    DC      Z'A0A0'
007133  1289  E0E0                    DC      Z'E0E0'
007134                          *#
007135                          *##
007136                          *
007137  128A  00E0            TLUWB3  DC      X'E0'           SHOULD BE TABLE FOR BROADBAND 301
007138  128B  10F0                    DC      X'10F0'
007139                          *
007140  128C  0060            TLUV35  DC      X'60'           SHOULD BE TABLE FOR BROADBAND V35
007141  128D  90F0                    DC      Z'90F0'
007142                          *
007143  128E  2020            SPACE   DC      Z'2020'
007144                          *
007145  128F  0020            TLUTB2  DC      X'20'
007146  1290  4060                    DC      X'4060'
007147  1291  0020                    DC      X'20'
007148  1292  4060                    DC      X'4060'
007149  1293  0010                    DC      X'10'
007150  1294  4050                    DC      X'4050'
007151  1295  0010                    DC      X'10'
007152  1296  4050                    DC      X'4050'
007153  1297  0000                    DC      0
007154  1298  0040                    DC      X'40'
007155  1299  0040                    DC      X'40'
007156                          *
007157  129A  0001            XMSB    DC      1               XMIT MARK SHOULD BE
007158  129B  0203                    DC      X'0203'
007159  129C  FF07                    DC      Z'FF07'
007160  129D  0809                    DC      X'0809'
007161                          *
007162  129E  0001            XSSB    DC      1               XMIT SPACE SHOULD BE
007163  129F  0203                    DC      X'0203'
007164  12A0  0007                    DC      X'0007'
```

```
007165  12A1  0809                      DC      X'0809'
007166                          *
007167  12A2  0002              C2      DC      2
007168  12A3  0004              C4      DC      4
007169  12A4  0008              C8      DC      8
007170  12A5  0010              C16     DC      16
007171  12A6  0200              HX200   DC      X'200'
007172  12A7  0400              HX400   DC      X'400'
007173                          *
007174                          *=============================================================================
007175                          *
007176                          *                          TRAP SAVE AREAS
007177                          *
007178  12A8  12B0              TSA1    DC      <TSA2
007179  12A9  0000                      RESV    4+3*$AF,0
007180  12B0  12B8              TSA2    DC      <TSA3
007181  12B1  0000                      RESV    4+3*$AF,0
007182  12B8  0000              TSA3    RESV    8,0
007183                          *
007184                          *=============================================================================
007185                          *
007186                          *                  INTERRUPT SAVE AREAS
007187                          *
007188  12C0  0000              ISA1    RESV    3+$AF,0                 INTERRUPT SAVE AREA FOR LEVEL 1 (RTC)
007189  12C4  048C              ISA1P   DC      <NOSPD
007190  12C5  6000                      DC      X'6000'
007191                          *
007192  12C6  0000              ISA4    RESV    3+$AF,0
007193  12CA  0368              ISA4P   DC      <SPSYT1
007194  12CB  6000                      DC      X'6000'
007195                          *
007196                          *=============================================================================
007197                          *
007198                          *                          LINE CONTROL TABLES
007199                          *
007200  12CC  0206              LCT     DC      X'206'                  STARTING ADDRESS OF CHANNEL PROG.
007201  12CD  0414                      DC      X'414'                  RECV. LINE CONTROL SET TEST MODE
007202  12CE  0000                      DC      0                       TABLE END
007203                          *
007204  12CF  0206              LCT1    DC      X'206'
007205  12D0  0426                      DC      X'426'
007206  12D1  C802                      DC      Z'C802'
007207  12D2  C822                      DC      Z'C822'
007208  12D3  0F18              LCT1S   DC      X'F18'
007209  12D4  C714              LC1LCR  DC      Z'C714'                 LINE CONTROL
007210  12D5  0000              LC1SYF  DC      0                       SYNC. FLAG
007211  12D6  0000                      DC      0                       END OF TABLE
007212                          *
007213  12D7  02C6              LCT2    DC      X'206'
```

```
007214  12D8  0426                      DC      X'426'
007215  12D9  C802              LC2CFR  DC      Z'C802'
007216  12DA  C822              LC2CFX  DC      Z'C822'
007217  12DB  0F18              LCT2S   DC      X'F18'
007218  12DC  C714                      DC      Z'C714'
007219  12DD  0117              LCT2FG  DC      X'117'          FLAG FOR TYPE OF SEND AND RECV
007220  12DE  0000                      DC      0
007221                          *
007222  12DF  0206              LCT3    DC      X'206'          RAM STARTING ADDRESS (RECV)
007223  12E0  0426                      DC      X'426'          RAM STARTING ADDRESS (XMIT)
007224  12E1  C002              LC3CFR  DC      Z'C002'         CONFIGURATION (RECV)
007225  12E2  C022              LC3CFX  DC      Z'C022'         CONFIGURATION (XMIT)
007226  12E3  0000              LCT3S   DC      0               SPEED IN SCRATCH LOCATION
007227  12E4  C714                      DC      Z'C714'         LINE CONTROL (RECV)
007228  12E5  0000              LC3SYF  DC      0               SYNC FLAG
007229  12E6  0000                      RESV    1,0
007230                          *
007231  12E7  0206              LCT4    DC      X'206'
007232  12E8  0426                      DC      X'426'
007233  12E9  0002              LC4CFR  DC      2
007234  12EA  0022              LC4CFX  DC      X'22'
007235  12EB  0000              LCT4S   DC      0
007236  12EC  C714                      DC      Z'C714'
007237  12ED  0000                      RESV    1,0
007238                          *
007239  12EE  0206              LCT5    DC      X'206'
007240  12EF  0426                      DC      X'426'
007241  12F0  C802                      DC      Z'C802'
007242  12F1  C822                      DC      Z'C822'
007243  12F2  0F18              LCT5S   DC      X'F18'
007244  12F3  C714                      DC      Z'C714'
007245  12F4  0F10                      DC      X'F10'          LCT STATUS BYTES
007246  12F5  0F11                      DC      X'F11'          "
007247  12F6  0000              LC5SYF  DC      0               SYNC. LINE FLAG
007248  12F7  0000                      DC      0
007249                          *
007250  12F8  0206              LCT6    DC      X'206'
007251  12F9  0226                      DC      X'226'
007252  12FA  6027                      DC      X'6027'
007253  12FB  C802                      DC      Z'C802'
007254  12FC  C822                      DC      Z'C822'
007255  12FD  0F18              LCT6S   DC      X'F18'
007256  12FE  C714                      DC      Z'C714'
007257  12FF  021D              LCT6TF  DC      X'21D'
007258  1300  0125                      DC      X'125'          DISABLE PAUSE
007259  1301  0000                      DC      0
007260                          *
007261  1302  0206              LCT7    DC      X'206'
007262  1303  0226                      DC      X'226'
```

```
007263  1304  C802                      DC      Z'C802'
007264  1305  C822                      DC      Z'C822'
007265  1306  FC1D            LC7SM     DC      Z'FC1D'          MASK TO CHECK BITS(FC-ASY.  F8-SYC.)
007266  1307  0439                      DC      X'439'           TLU POINTER TABLE  1
007267  1308  0105                      DC      X'105'
007268  1309  0125                      DC      X'125'
007269  130A  0000                      RESV    1,0
007270                          *
007271  130B  0206            LCT8      DC      X'206'
007272  130C  0426                      DC      X'426'
007273  130D  C802                      DC      Z'C802'
007274  130E  C822                      DC      Z'C822'
007275  130F  C518                      DC      Z'C518'
007276  1310  C714                      DC      Z'C714'
007277  1311  0105                      DC      X'105'           DISABLE PAUSE
007278  1312  0125                      DC      X'125'           DISABLE PAUSE
007279  1313  0000            LCT8FG    DC      0
007280  1314  0000                      RESV    1,0
007281                          *
007282  1315  0206            LCT9      DC      X'206'
007283  1316  0426                      DC      X'426'
007284  1317  C802                      DC      Z'C802'
007285  1318  C822                      DC      Z'C822'
007286  1319  0000            LCT9MS    DC      0
007287  131A  C314                      DC      Z'C314'
007288  131B  0000                      DC      0
007289                          *
007290                          *
007291                          *=================================================================================
007292                          *
007293                          *                          FORMAT  FOR DRIVER
007294                          *
007295  131C  0006            WCP       DC      6                RANGE OF PROGRAM IN BYTES
007296                          *
007297  131D  0E17            WCP1      DC      <CHLP1           CHANNEL PROGRAM ADDRESS
007298  131E  0E2F                      DC      <CHLP1A          XMIT CHANNEL PROGRAM ADDRESS
007299  131F  0030            RAG1      DC      (CHLP1A-CHLP1)*2 RANGE OF CHANNEL PROGRAM
007300  1320  003C            RAG2      DC      (CHLP2-CHLP1A)*2 RANGE OF PROGRAM IN BYTES
007301  1321  0400                      DC      X'400'           RAM ADDRESS
007302                          *
007303  1322  0E4D            WCP2      DC      <CHLP2           CHANNEL PROGRAM ADDRESS
007304  1323  0E75                      DC      <CHLP2A          XMIT CHANNEL PROGRAM ADDRESS
007305  1324  0050                      DC      (CHLP2A-CHLP2)*2 RANGE OF CHANNEL PROGRAM
007306  1325  0060                      DC      (CHLP3-CHLP2A)*2 RANGE OF CHANNEL PROGRAM
007307  1326  0400                      DC      X'400'           RAM STARTING ADDRESS
007308                          *
007309  1327  0EA5            WCP3      DC      <CHLP3
007310  1328  0EC9                      DC      <CHLP3A
007311  1329  0048                      DC      (CHLP3A-CHLP3)*2
```

```
007312  132A  007E                    DC      (CHLP4-CHLP3A)*2
007313  132B  0400                    DC      X'400'
007314                          *
007315  132C  0F08              WCP4  DC      <CHLP4
007316  132D  0F31                    DC      <CHLP4A
007317  132E  0052                    DC      (CHLP4A-CHLP4)*2
007318  132F  0072                    DC      (CHLP5-CHLP4A)*2
007319  1330  0260                    DC      X'260'
007320                          *
007321  1331  0054              WCP5  DC      (CHLP6-CHLP5)*2
007322  1332  0016              WCP5B DC      22
007323                          *
007324  1333  0F94              WCP6  DC      <CHLP6
007325  1334  0FD1                    DC      <CHLP6A
007326  1335  007A                    DC      (CHLP6A-CHLP6)*2
007327  1336  008E                    DC      (CHLP7-CHLP6A)*2
007328  1337  0400                    DC      X'400'
007329                          *
007330  1338  1018              WCP7  DC      <CHLP7
007331  1339  1031                    DC      <CHLP7A
007332  133A  0032                    DC      (CHLP7A-CHLP7)*2
007333  133B  0032                    DC      (CHLP8-CHLP7A)*2
007334  133C  0400                    DC      X'400'
007335  133D  0000                    RESV    2,0
007336                          *
007337  133F  104A              WCP8  DC      <CHLP8
007338  1340  108A                    DC      <CHLP8A
007339  1341  0080                    DC      (CHLP8A-CHLP8)*2
007340  1342  00A8                    DC      (CHLP9-CHLP8A)*2
007341  1343  0400                    DC      X'400'
007342                          *
007343  1344  10DE              WCP9  DC      <CHLP9
007344  1345  10F1                    DC      <CHLP9A
007345  1346  0026                    DC      (CHLP9A-CHLP9)*2
007346  1347  003C                    DC      (CHLP10-CHLP9A)*2
007347  1348  0400                    DC      X'400'
007348                          *
007349                          *=================================================================
007350                          *
007351                          *                       CONSOLE MESSAGES
007352                          *
007353  1349  434C 4120 5445    MESG  TEXT    'CLA TEST'
        134C  5354
007354                                  IFZ     ($AF-2),PLAF
007355  134D  2020 4443 4D56           TEXT    '  DCMVA,SAF-A '
        1350  412C 5341 462D
              4120
007359  1354  2020 2041 5547    SLAF  TEXT    '   AUG 1 1978$'
        1357  2031 2031 3937
```

```
                3824
007360  135B    4153 594E 4348          MESG2   TEXT    'ASYNCHRONOUS LINE$'
        135E    524F 4E4F 5553
                204C 494E 4524
007361  1364    2053 5045 4544          MESG4   TEXT    ' SPEED IS  $'
        1367    2049 5320 2024
007362  136A    2053 484F 554C          MESG5   TEXT    ' SHOULD BE $'
        136D    4420 4245 2024
007363  1370    4E45 5854 2024          MESG6   TEXT    'NEXT $'
007364  1373    434C 4120 4348          MESG7   TEXT    'CLA CHANNEL(S) $'
        1376    414E 4E45 4C28
                5329 2024
007365  137B    5041 5353 2400          MESG8   TEXT    'PASS$'
007366  137E    504F 5745 5220          MESG9   TEXT    'POWER FREQ(HZ) $'
        1381    4652 4551 2848
                5A29 2024
007367  1386    434F 5059 5249                  TEXT    'COPYRIGHT 1978 BY HONEYWELL  INFORMATION SYSTEMS INC.'
        1389    4748 5420 3139
                3738 2042 5920
                484F 4E45 5957
                454C 4C20 2049
                4E46 4F52 4D41
                5449 4F4E 2053
                5953 5445 4D53
                2049 4E43 2E00
007368  13A1    4649 524D 5741          FREV    TEXT    'FIRMWARE REV.$'
        13A4    5245 2052 4556
                2E24
007369  13A8    2020 2020 2024          ERMG    TEXT    '     $'
007370  13AB    434C 4120 4E4F          ERMG1   TEXT    'CLA NOT FOUND ON THIS CHANNEL  $'
        13AE    5420 464F 554E
                4420 4F4E 2054
                4849 5320 4348
                414E 4E45 4C20
                2024
007371  13BB    494E 434F 5252          ERMG2   TEXT    'INCORRECT ID ON THIS CHANNEL  $'
        13BE    4543 5420 4944
                204F 4E20 5448
                4953 2043 4841
                4E4E 454C 2020
                2400
007372                                  *##
007373                                  *#
007374  13CB    2042 4C4F 434B          MLMSG1  TEXT    ' BLOCKS TRANSMITTED WITH$'
        13CE    5320 5452 414E
                534D 4954 5445
                4420 5749 5448
                2400
007375  13D8    2045 5252 4F52          MLMSG2  TEXT    ' ERRORS$'
```

```
                13DB  5324
     007376     13DC  8020          MLCTRL DC      Z'8020'
     007377                         *#
     007378                         *##
     007379           13DD          V$LB   EQU     $
     007380     1C00                ORG     ZERO+X'1C00'
     007383           1C00          DIN    EQU     $
     007384     1D00                ORG     DIN+X'100'
     007385           1D00          DIN1   EQU     $
     007386     13DD                ORG     V$LB
     007387                         *
     007388     13DD  0100                 END     DCMVA,STRT
0000  ERR COUNT
```

```
title    dcmva,'78072500'    mlcp line adapter test

$af       775    776c    784    785c    880    1187    1204    1418    1435    1469
         1485    1654    1671   2151c   2153c  2171c   2188c   2207c   2209c   2221c
         2239   2334c    2367   2387c   2389c  2390c   2392c   2394c    2403   2427c
         2430   2680c    3014   3015c    3019   3020    3024    3030    3031    3035
         3037   3042c    3045    3051    3059   3060    6970    6971    6972    6973
         7179    7181    7188    7192    7354   7356    7381
$b1       408     414     596     607     621    635     648     658     695     705
          712     723     733     740     750    797     804     811     833     838
          843     848     853     858     863   877c    1041    1051    1058    1068
         1118    1129    1139    1150    1160   1167    1178    1181   1182c   1183c
         1184    1194    1201    1211    1260   1349    1360    1370    1381    1391
         1398    1409    1412   1413c   1414c   1415    1425    1432    1442    1495
         1500    1504    1538    1585    1596   1606    1617    1627    1634    1645
         1648   1649c   1650c    1651    1661   1668    1678    1725    1736    1746
         1759    1769    1776    1788    1798   1805    1815    1956    1967    1977
         1991    2001    2008    2022    2032   2039    2049    2150   2151c    2152
        2153c    2164    2167    2170   2171c   2172    2175    2183   2188c    2191
         2194    2197    2206   2207c    2208   2209c   2210    2215    2220   2221c
         2222    2227   2236c    2301    2310   2312    2312   2364c   2655c    2865
         2868    2895    2896    3030    3035  3036c   3064c
$b2       366     371     407     410    488b    489b    490b    491b    492b    493b
         494b    495b    496b    497b    532b    533b    534b    535b    536b    553b
         556b    562b    578b    755b    869b   1073b   1229b    1237    1238    1254
         1263    1280    1307   1458b   1509b   1684b   1821b   2064b   2229b   2321b
         2897    2900    2976    2982    3019   3023c    3024   3026c   3050c   3054c
         3060   3063c   3078b
$b3      588b    597b    608b    636b    649b    696b    713b    724b    741b    763b
         799b    806b    813b    834b    839b    844b    849b    854b    859b    864b
         881b   1034b   1042b   1059b   1101b   1119b   1130b   1151b   1168b   1185b
        1202b    1259    1306    1310   1334b   1350b   1361b   1382b   1399b   1416b
        1433b   1466b   1497b   1501b   1505b   1570b   1586b   1597b   1618b   1635b
        1652b   1669b   1714b   1726b   1737b   1760b   1777b   1789b   1806b   1943b
        1957b   1968b   1992b   2009b   2023b   2040b   2148b   2165b   2168b   2173b
        2192b   2195b   2198b   2211b   2216b   2223b   2228b   2240b   2288b   2302b
        2368b    2375    2375    2377    2399    2420    2424    2424    2425   2543b
         2587    2597    2621    3029   3036c    3041   3042c   3064c
$b4       367    368c     369     370c    440b    509b    517b     773     774c     775
          776c     781    782c     784    785c     819    820c    890b    891b    894b
          934b    943b    952b    954b    978b    996b   1016b   1269b   1313b    1467
         1468c    1469   1470c    1482   1483c    1485   1486c   1690b   1698b   1829b
         1831b   1835b   1843b   1844b   1847b   1850b   1863b   1871b   2079b   2090b
         2241b   2260b   2266b   2374b   2376b   2382b    2386   2387c    2388   2389c
         2406b   2414b   2442b   2451b   2465b   2472b   2602b   2658b    2668   2676b
          2679   2680c   2682b   2692b   2699b   2808b   2817b   2823b   2850b   2855b
         2875b   2937b   2961b   2980b   2986b    3018    3025    3050c    3062   3063c
$b5       425     428    430c     457b    506b    514b    523b    879b    902b    907b
          916b    924b    938b    1006   1007c    1022   1023c    1279   1325b    1557
         1558c   1857b   2238b   2251b   2255b   2264b    2329    2333   2334c   2366b
          2377    2386    2388   2390c    2391    2391   2392c    2393    2393   2394c
          2399   2400c   2402b    2425   2426c   2427c   2429b   2434b   2446b   2455b
         2481b   2687b   2697b   2726b   2767b   2779b   2781b   2821b    2865   2872b
          2895    2904    2904   2906b   2925b    2976    2977   2992b    3022   3023c
$b6       441b     798     805     812     817     905   1078b   1082b    1108    1109
         1216    1225    1226   1282b   1288b   1296b    1304    1496   1878b   1884b
         1892b   2072b   2098b    2294    2310   2346b   2350b   2552b    2581    2734
         2796    2800    2802    2940    2942   2946b    3014   3015c    3017   3026c
         3053   3054c
```

$b7	520b	525b	937b	951b	957b	977b	1088b	1091b	1277b	1319b
	1323b	1517b	1520b	1556b	1693b	1696b	1701b	1704b	1834b	1837b
	1853b	1867b	1875b	1890b	1898b	1919b	1929b	2082b	2085b	2093b
	2097b	2106b	2112b	2263b	2269b	2276b	2341b	2357b	2410b	2418b
	2445b	2454b	2469b	2475b	2507b	2522b	2695b	2820b	2853b	2990b
	3014	3058b								
$r1	330	331	371	372b	373	374	377	378	383	387
	388c	393	394c	406c	410	411	418c	419	420	424
	427	429	430c	883c	919c	941c	960b	1017c	1018b	1020b
	1899	1900c	1901	1902	1903	1904c	1905c	1909	1913	1917
	1920c	1921	1926	1927	1930c	1931	2582c	2590c	2592	2594b
	2616	2617c	2707c	2710	2715	2720	2734	2735	2737	2738
	2745	2756	2762	2866c	2868	2869c	2915	2916c	2917	2919
	3012	3055b								
$r2	365c	371	392c	393	409	414	416b	423	428	430c
	431b	884c	905	982c	985	1527c	1530	1536c	1715	2300
	2312	2584c	2592	2593	2595	2600c	2619c	2621	2655c	2734
	2802	2832	2868	2870b	2876	2896	2900	2977	2987	
$r3	383	384	385	386c	439	505c	876c	906c	910c	932c
	939c	988	1697c	1828c	1838b	1839c	1869c	2071c	2086c	2237c
	2278	2365c	2412c	2433c	2439c	2448c	2463c	2471c	2526c	2549
	2583c	2587	2597	2667	2751	2774	2775c	2802	2803c	2894
	2898	2900	2902c	2903	2958	2959	2960	3008	3009	3010
	3011c	3013c	3029	3032	3040c	3041	3049c	3051		
$r4	379	386c	410	413b	414	426	428	429	446	447c
	458	459	462c	465	467	471	473	475	477	481
	486	487c	510	511	512c	516	518	529	530c	531c
	545	546c	548	549c	554	555c	568	569	584	585c
	586	587c	619	620c	621	622	628	629c	646	647c
	658	659	663	664	668	669	670	674	675c	676
	677c	678	679c	683	684	686	705	706	733	734
	750	751	761	762c	764	765	766	767c	768	769c
	770	771	772c	778	779c	789	790c	796	803	810
	815	816c	823	824c	825	826c	827	885c	887	888
	892	896	897	898c	905	911b	912c	913	915	920
	921c	933	935	942	949	953	955	963	964	971
	973	980	981c	983	984	997	998c	1021	1032	1033c
	1051	1052	1068	1069	1099	1100c	1104	1105c	1106	1107c
	1109	1110c	1139	1140	1160	1161	1194	1195	1211	1212
	1216	1217b	1218	1219	1221	1222c	1223	1224c	1226	1238
	1239	1240	1241c	1245	1247	1249	1251	1252c	1254	1255
	1263	1264	1265	1266	1267c	1292	1293	1294c	1300	1301
	1302c	1307	1308	1332	1333c	1336	1337c	1338	1339c	1340
	1341c	1343	1344c	1370	1371	1391	1392	1425	1426	1442
	1443	1448	1449	1451c	1452	1453c	1464	1465c	1472	1473
	1474	1475c	1476	1477c	1479	1480c	1490	1491c	1492	1493c
	1494	1499	1503	1515	1518	1532	1533c	1534	1535c	1537
	1541	1544	1545c	1546	1547c	1568	1569c	1572	1573c	1574
	1575c	1576	1577c	1606	1607	1627	1628	1661	1662	1678
	1679	1712	1713c	1718	1719c	1746	1747	1751	1752c	1753
	1754c	1769	1770	1798	1799	1815	1816	1849	1851	1862
	1864	1870	1872	1910	1911	1912	1913	1914	1915c	1916
	1917	1925	1927	1941	1942c	1947	1948c	1950	1951c	1977
	1978	2001	2002	2020	2021c	2032	2033	2049	2050	2058
	2059c	2078	2080	2089	2091	2146	2147c	2154	2155c	2156
	2157c	2178	2179	2181	2184	2185c	2189	2190c	2202	2203c
	2204	2205c	2245	2246	2247c	2254	2259	2261	2286	2287c
	2289	2290c	2292	2295c	2315	2316	2319	2320c	2330	2331
	2343	2344	2371	2372	2380	2383	2384	2385c	2405	2407
	2413	2415	2420	2421c	2422	2441	2443	2450	2452	2581

	6983	bisync	706	833	1161	1392	1628	1770	2002	2191		
	6990	bit0	408	2677	2818							
	6991	bit1	518	935	949	2261	2443	2452	2693			
	6992	bit2	1715									
n	6993	bit4										
n	6994	bit6										
	6995	bit7	1864	1872	2407	2415						
	7010	bkls	2791	2793	3022	3025						
	2732	bps	2781b									
	2739	bps1	2736b									
	2750	bps2	2748b									
	2749	bps2a	2747b									
	5015	bufe	5023	5024	5026							
	6931	bytp	792c	821c	2776							
	7170	c16	361									
	7167	c2	2206									
	7168	c4	2152	2335								
	7169	c8	364									
	541	cablp	468b	472b	474b	575b	579b					
	409	car	415b									
	4576	cdly	4566	4567	4569							
	2958	cgsch	517b	934b	943b	954b	1850b	1863b	1871b	2079b	2090b	2260b
			2406b	2414b	2442b	2451b	2676b	2682b	2692b	2817b	2850b	2875b
			2937b	2980b	2986b							
	6932	chanl	362c	388c	420	434						
	3087	chlp	2668									
	3118	chlp1	899	7297	7299							
	6713	chlp10	7346									
	3241	chlp1a	900	7298	7299	7300						
	3386	chlp2	7300	7303	7305							
	3581	chlp2a	7304	7305	7306							
	3803	chlp3	7306	7309	7311							
	3982	chlp3a	7310	7311	7312							
	4268	chlp4	7312	7315	7317							
	4471	chlp4a	7316	7317	7318							
	4735	chlp5	2248	7318	7321							
	4927	chlp6	7321	7324	7326							
	5217	chlp6a	7325	7326	7327							
	5553	chlp7	7327	7330	7332							
	5679	chlp7a	7331	7332	7333							
	5799	chlp8	7333	7337	7339							
	6109	chlp8a	7338	7339	7340							
	6484	chlp9	7340	7343	7345							
	6576	chlp9a	7344	7345	7346							
	3102	chlps	2679									
	6934	chnz	512c	513	898c	899	900	2247c	2248	2249	2385c	2396
			2397	2684c	2686	2939c	2944					
	1251	chrf1	1248b									
	5040	chrl	5027	5028	5032							
	1456	chszde	1450b									
	6933	chszfg	1335c	1457c	2509	2541c	2622					
	1342	chszt1	1454b									
	1332	chsztd	492b									
	1282	ckcrc	1256b									
	3171	ckff	3162	3163	3165							
	1311	ckpcr	1309b									
	1279	ckpty2	1273b									
	2535	clfg	588b	763b	1034b	1101b	1334b	1466b	1570b	1714b	1943b	2148b
			2288b									
	6578	clrk	6683	6684	6685							

	2412	dt2	2408b								
	2420	dt3	2416b								
	2375	dtal	2373b								
	2363	dtalop	597b	608b	636b	649b	696b	713b	724b	741b	1042b	1059b
			1119b	1130b	1151b	1168b	1185b	1202b	1350b	1361b	1382b	1399b
			1416b	1433b	1586b	1597b	1618b	1635b	1652b	1669b	1726b	1737b
			1760b	1777b	1789b	1806b	1957b	1968b	1992b	2009b	2023b	2040b
			2302b								
	1324	edck	1281b	1322b							
	361	edlint	326b	396b							
	4432	edly	4427	4428	4430	4461	4462	4463			
	3074	edpck	497b	536b	578b						
	4210	edxt	4179	4180	4182	4200	4201	4202	4259	4260	4261	
	4915	endtc	4898	4899	4901						
	4254	eop	4241	4242	4243						
	3026	er11	3020								
	6968	erar	1271	1272	1274	1275	1276	2270	2271	2273	2274	2275
			2335	2336	2338	2339	2340	2488	2494	2504	2505	2506
			2516	2517	2519	2520	2521					
	3003	erdb	977b	1088b	1091b	1319b	1323b	1517b	1520b	1556b	1834b	1837b
			1890b	1898b	1919b	1929b	2085b	2097b	2106b	2112b	2276b	2341b
			2357b	2469b	2475b	2507b	2522b					
	6944	erf	975	976c	979c	3027						
	2998	ermb	520b	525b	937b	951b	957b	1277b	1693b	1696b	1701b	1704b
			1853b	1867b	1875b	2082b	2093b	2263b	2269b	2410b	2418b	2445b
			2454b	2695b	2820b	2853b	2990b					
	7369	ermg	585c	762c	1033c	1100c	1333c	1465c	1569c	1713c	1942c	2147c
			2287c	3000c	3005c	3007	3010	3011c	3016			
	7370	ermg1	391									
	7371	ermg2	417									
	5771	err	5762	5763	5764							
	3016	err1	3015c									
	3029	err2	3055b									
	3036	err21	3031	3034b								
n	3035	err21a										
	3044	err3	3019	3030	3037	3039b	3042c	3060	3065b			
	3043	err31	3024	3035								
	3049	err34	3045	3047b								
	3055	err4	3052b									
	3057	err5	3028b									
	3007	error	3001b									
	4682	exm	4673	4674	4675	4726	4727	4728				
	4662	exm1	4652	4653	4655							
	6091	ext	5919	5920	5921	5942	5943	5945	5982	5983	5984	6038
			6039	6040	6061	6062	6064	6083	6084	6085		
	6945	fbrg	1900c	1931								
	6917	fid	425									
n	6915	fidss										
	5156	fif3	5138	5139	5143							
	5131	fife	5158	5159	5160							
n	6916	fils										
	6914	fins	2681	2840								
	6912	fir	2849	2936								
	6913	fis	953	1849	2078	2089						
	7022	fivsec	772c	944	963	1021						
	2893	fln	506b	879b	2238b	2366b						
	2898	fln2	2901b									
	2905	fln3	2899b									
	6910	fobl	2874									
	6911	focb	2985									

6908	focc	516	933	942	1862	1870	2259	2405	2413	2441	2450
		2675	2691	2833							
n 6907	foic										
6909	fold	2979									
6906	fomc	2816									
7368	frev	448									
7026	frimrv	447c									
5957	fvbc	5885	5886	5888	5952	5953	5955				
5949	fvby	5967	5968	5970	5986	5987	5988				
6946	fwfg	325	450c								
2814	genitz	440b	509b	891b	2241b	2376b					
3190	getc	3146	3147	3148	3205	3206	3210	3216	3217	3218	
3201	getc1	3177	3178	3179							
3220	getx	3211	3212	3214							
5755	hang	5768	5769	5773							
6947	hgspd	433c	1179	1410	1646	2749c	2782				
3120	hlp1	3233	3234	3236							
3208	hlp1a	3197	3198	3199							
6948	hrtz	334	340	352	764	770	1472				
7171	hx200	513	899	2248	2370	2396	2686				
7172	hx400	900	2249								
2848	inrg	1016b	1835b	1844b							
454	invd	435b	482b	501b							
2839	inxt	952b	1690b	1698b	1831b	1843b	2266b	2465b	2472b		
1015	irrg	819	1025b								
1019	irrg1	1020b									
1021	irrg2	1018b									
7188	isa1	784	1485								
7189	isa1p	782c	820c	1483c							
7192	isa4	775	1469								
7193	isa4p	774c	1007c	1023c	1468c	1558c					
3461	jfqq	3435	3436	3438							
4362	jfrr	4335	4336	4337							
3892	jfyy	3866	3867	3869							
3420	jhqq	3458	3459	3463							
4320	jhrr	4358	4359	4360							
3839	jhyy	3889	3890	3894							
2455	jp2	2453b									
2459	jp2a	2457b									
2465	jp2b	2460b									
2471	jp3	2468b									
2488	jp4	1933b	2427c								
2476	jp4a	2474b									
2483	jp4c	2480b									
2509	jp5	1279	2495b	2501b							
2526	jp5a	1092b	1705b	2113b	2359b	2510b	2513b				
2524	jp5b	2518b									
952	jpt1	950b	960b								
958	jpt2	956b									
879	jpt3	990b									
969	jpt4	1006									
5657	ker	5644	5645	5647							
6666	kunta	6643	6644	6648	6654	6655	6656				
7209	lc1lcr	487c	530c	549c	663	675c					
7210	lc1syf	590c	679c								
7215	lc2cfr	1105c	1218	1222c	1573c						
7216	lc2cfx	1107c	1224c	1575c							
7224	lc3cfr	1491c	1521	1524c	1533c	1545c					
7225	lc3cfx	1493c	1526c	1535c	1547c						
7228	lc3syf	791c	824c	969	1488c	2794					

7233	lc4cfr	1337c	1448	1451c						
7234	lc4cfx	1339c	1453c							
7247	lc5syf	1717c	1754c	1907						
7265	lc7sm	2155c	2190c	2203c						
3735	lcb1	3693	3694	3695	3710	3711	3713	3726	3727	3728
3754	lcb2	3747	3748	3750						
7200	lct	2688								
7204	lct1	600	611	639	652	699	716	727	744	
7208	lct1s	587c	620c	629c	647c	677c				
7213	lct2	1122	1133	1154	1171	1188	1205	1589	1600	1621
		1655	1672							
7219	lct2fg	1110c	1577c							
7217	lct2s	1112c	1145c	1579c	1612c					
7222	lct3	903								
2876	lct3a	2877b								
7226	lct3s	885c	983							
7231	lct4	1353	1364	1385	1402	1419	1436			
2868	lct4a	2878b								
7235	lct4s	1344c	1376c							
7239	lct5	1729	1740	1763	1780	1792	1809			
2874	lct5a	2870b								
7243	lct5s	1719c	1752c							
7250	lct6	1960	1971	1995	2012					
7255	lct6s	1951c	1986c							
7257	lct6tf	1948c	1984c							
7261	lct7	2252								
7271	lct8	1045	1062	2026	2043					
7279	lct8fg	1035c	1944c	2021c	2059c	2063c				
7282	lct9	2305								
7286	lct9ms	2295c	2316	2330	2344					
6967	lctv	445c	446	449	1086	1089	1292	1300	1888	1896
		2110	2355	2944						
4442	less	4450	4451	4455						
5620	lkb1	5631	5632	5633						
4858	loop1	4865	4866	4867						
4852	loop2	4879	4880	4886						
521	loopb1	519b	524b							
504	loopbk	476b								
525	looper	508b								
6886	lopcce	6887								
6717	lopccp	513	6887							
2238	lopd	2279b								
2239	lopd1	2236c								
2264	lopd2	2262b								
2278	lopd3	2272b								
2270	lopd7	2151c	2153c	2171c	2188c	2207c	2209c	2221c	2268b	
2236	lopds	2165b	2168b	2173b	2192b	2195b	2198b	2211b	2216b	2223b
6847	lopdso	6831	6832	6833						
6779	lopdss	6764	6765	6766						
6894	loplct	515								
6887	lopsze	513								
6752	lopwt	6776	6777	6781	6861	6862	6871			
6869	lopxmt	6881	6882	6885	6895	6896				
3243	lp1a	3378	3379	3381						
3325	lp1aa	3350	3351	3355						
3333	lp1ab	3322	3323	3327						
3314	lp1abd	3266	3267	3269	3329	3330	3331			
3353	lp1ac	3344	3345	3346						
3372	lp1ac1	3360	3361	3363						
2184	lpcb1	2180b								

3686	snc3	3699	3700	3704							
7143	space	3018	3062								
3066	space1	3048									
875	spdtlp	799b	806b	813b	834b	839b	844b	849b	854b	859b	864b
		1497b	1501b	1505b							
778	spsyt1	773	7193								
761	spsytc	489b	533b								
880	st1	877c	887	2798							
896	st2	893b									
955	st3	959b									
1464	stbite	493b									
2831	stcr	1829b	1847b								
1712	stiog	495b									
1828	stiq	2458b									
1829	stiq1	1858b									
1856	stiq2	1842b									
1844	stiq3	1846b									
1854	stiq4	1852b									
1832	stiq5	1854b									
2666	stmd	894b	2382b								
2697	stmd1	2694b									
2680	stmd2	2669b									
2686	stmd3	2680c									
314	stop	3074c	3076b								
324	stp	319b									
5200	stp3	5073	5074	5076	5100	5101	5102	5121	5122	5127	
6964	stpfg	1716c	1820c	2456	2538c						
318	strt	7388									
	strtd2	522	523b								
815	stsp1	788b									
848	stsp2	829b									
3067	swt	3021c	3038	3046	3061c						
5904	sxb1	5899	5900	5902							
5896	sxbc	5914	5915	5917	5923	5924	5925				
5847	syc	5838	5839	5840	5865	5866	5868				
5817	sych	5844	5845	5849							
947	syct	948b									
7033	sylsp0	2751									
n 7034	sylsp1										
n 7035	sylsp2										
n 7036	sylsp3										
n 7037	sylsp4										
n 7038	sylsp5										
n 7039	sylsp6										
n 7040	sylsp7										
6984	syn188	712	843	1167	1398	1634	1776	2008	2197		
6982	sync	507	695	838	1150	1381	1617	1759	1991	2194	
992	synsp	970b	1022								
997	synsp1	993b	995b								
607	t1	598b									
1160	t10	1152b									
1167	t12	1162b									
1178	t13	1169b									
1182	t13x	1180b									
1187	t13y	1183c									
1202	t13yy	1214b									
1204	t13z	1182c									
1185	t13zz	1197b									
1360	t14	1351b									
1361	t14a	1373b									

1370	t15	1362b						
1374	t15a	1372b						
1391	t16	1383b						
1398	t17	1393b						
1409	t18	1400b						
1413	t18x	1411b						
1418	t18y	1414c						
1416	t18yy	1428b						
1435	t18z	1413c						
1433	t18zz	1445b						
1736	t19	1727b						
1737	t19a	1749b						
608	t1a	624b						
621	t2	618b						
1746	t20	1738b						
1750	t20a	1748b						
1769	t21	1761b						
1776	t22	1771b						
1786	t23	1778b						
1789	t23a	1801b						
1967	t24	1958b						
1968	t24z	1980b						
1977	t25	1969b						
5636	t25a	5648	5649	5651	5653	5654	5655	
5742	t25b	5750	5751	5753	5790	5791	5793	
1981	t25z	1979b						
2001	t26	1993b						
2008	t27	2003b						
2019	t28	2010b						
740	t29	735b						
733	t29a	725b						
741	t29z	753b						
625	t2a	623b						
617	t2aa	609b						
646	t3	637b						
755	t30	752b						
750	t30a	742b						
2039	t31	2034b						
2032	t31a	2024b						
2040	t31z	2052b						
2054	t32	2051b						
2049	t32a	2041b						
1805	t33	1800b						
1798	t33a	1790b						
1806	t33z	1818b						
1820	t34	1817b						
1815	t34a	1807b						
1432	t35	1427b						
1425	t35a	1417b						
1447	t36	1444b						
1442	t36a	1434b						
1201	t37	1196b						
1194	t37a	1186b						
1216	t38	1213b						
1211	t38a	1203b						
1596	t39	1587b						
1597	t39a	1609b						
649	t3a	661b						
658	t4	650b						
1606	t40	1598b						

```
6961  tp2       455   1877c  1911   1915c  1921   2157c  2205c  2254   2744
5408  trcv     5395   5396   5400
3150  trnx     3167   3168   3169   3184   3185   3186
 975  tryh      972b
7178  tsa1      367
7180  tsa2     7178
7182  tsa3     7180
 988  tsnl      911b   999b  1548b
1899  tstf2    1897b
1885  tstf2b   1880c
1835  tstf3    1833b
1838  tstf4    1836b
1862  tstf4a   1838b
1925  tstf4c   1908b  1922b
1927  tstf4d   1932b
1930  tstf4e   1928b
1920  tstf5    1918b
1917  tstf5a   1923b
1892  tstf7k   1889b
1893  tstf8b   1882c
 982  tyxl      974b
2309  u1       2303b
2294  u2       2317b
2302  u3       2313b
2315  u5       2311b
5104  udor     5000   5001   5002
5125  und      5110   5111   5112
1941  uornth    496b
7379  v$lb     7386
7295  wcp      2686
7297  wcp1      599    610    638    651    698    715    726    743   1728   1739
               1762   1779
7303  wcp2     1121   1132   1153   1170   1178   1588   1599   1620   1637   1645
7309  wcp3     1352   1363   1384   1401   1409
7315  wcp4     1959   1970   1994   2011
7321  wcp5     2248
7322  wcp5b    2249
7324  wcp6     1044   1061   2025   2042
7330  wcp7     1791   1808
7337  wcp8     1181   1412   1648
7343  wcp9     2304
6985  wdb3      723    848   1041   1184   1415   1651   1788   2022   2210
6986  wdb3b     734    853   1052   1195   1426   1662   1799   2033   2215
6965  wdbfg    1786c  1841   1945c  1982c  2019c  2054   2056c  2057c  2061c  2076
               2087   2102   2107   2539c
6987  wdbv35    740    858   1058   1201   1432   1668   1805   2039   2222
6988  wdbv5b    751    863   1069   1212   1443   1679   1816   2050   2227
6966  xcrc     1302c  1320
5261  xmbb     5286   5287   5291
2286  xmmsl     562b
4156  xmns     4108   4109   4111   4227   4228   4229
4500  xmrr     4525   4526   4530
7157  xmsb     2329
2327  xmsck    2464b
2334  xmsck1   2332b
2335  xmsck2   2334c
2343  xmsck3   2337b
2351  xmsck4   2348c
2358  xmsck5   2345b  2356b
3273  xmsy     3298   3299   3300
```

```
3623  xmvv      3648    3649    3653
5701  xmww      5726    5727    5728
4115  xmzz      4140    4141    4142
4189  xnpy      4163    4164    4169    4206    4207    4212
4167  xpy       4184    4185    4186
4097  xpyq      4024    4025    4026    4051    4052    4056    4078    4079    4080
6320  xpyz      6336    6337    6339    6341    6342    6343
7162  xssb      2333
6264  z2z       6287    6288    6289
6291  z3z       6283    6284    6285
6232  z47       6160    6161    6163    6185    6186    6187    6213    6214    6218
6246  z48       6140    6141    6143
 307  zero       313     320    7380    7382
6453  zext      6332    6333    6335    6356    6357    6358    6399    6400    6402    6419
                6420    6421    6440    6441    6443
      zhcomm     312
      zhiafb     309
      zhisaz     309     776c    785c   1470c   1486c
      zhntsa     308     368c
      zhpfr      312
      zhrtcc     310     945c     947     964   2916c   2919
      zhrtci     311
      zhrtcl     311     779c   1480c
      zhth15     308     370c
      zhwdtc     310
      zv$c      1271b   2270b   2335b   2488b   2516b
      zv$co     1278b   2277b   2342b   2508b   2523b
      zv$er     3016b
      zv$f       361b   2370b   2515b
      zv$hr      308
      zv$ia      455b
      zv$id      334b
      zv$ih      364b
      zv$iz      321b
      zv$mlr    2944b
      zv$mlw     513b    899b    900b   2248b   2249b   2396b   2397b   2686b
      zv$pch     500b
      zv$qc      333b    363b    454b
      zv$rd      321b    324b    867b   1507b
      zv$t       572b    574b   2778b   2792b   2801b   3048b
      zv$tc      391b    417b    448b   2776b   3077b
      zv$td      449b    571b    573b   2777b   2791b   2793b   2804b
      zv$th      395b    571b    573b   3043b
      zv$thz    3044b
      zv$tty     310     322c    3075
6345  zxz1      6317    6318    6322    6360    6361    6362    6366    6367    6371
```

```
 769 labels
3710 references
7388 records
   C u flags
   C m flags
  13 n flags
```

L6 Cross Ref Version L - 24 Sept, 1976

```
HRS Linker Version 5.05 07/28/78  1737.9 edt Fri
 LINK MAP for DCMVA

  START          0100
  LOW            0000
  HIGH           1CFF
  CURRENT        1CF1


**LOC DEFS
P  ZHCOMM        0000
  *DCMVA         0000   78072500
   ZHPFR         0000
   ZHTSA         0002
   ZHNTSA        0010
   ZHRTCI        0014
   ZHRTCC        0015
   ZHRTCL        0016
   ZHWDTC        0017
   ZHMERC        001F
   ZHIAFB        0020
   ZHTH29        0063
   ZHTH28        0064
   ZHTH27        0065
   ZHTH26        0066
   ZHTH25        0067
   ZHTH24        0068
   ZHTH23        0069
   ZHTH22        006A
   ZHTH21        006B
   ZHTH20        006C
   ZHTH19        006D
   ZHTH18        006E
   ZHTH17        006F
   ZHMEMP        006F
   ZHTH16        0070
   ZHLERR        0070
   ZHTH15        0071
   ZHNRES        0071
   ZHTH14        0072
   ZHPMEM        0072
   ZHTH13        0073
   ZHP_OP        0073
   ZHTH12        0074
   ZHTH11        0075
   ZHTH10        0076
   ZHTH9         0077
   ZHTH8         0078
   ZHTH7         0079
   ZHTH6         007A
   ZHOVFL        007A
   ZHTH5         007B
   ZHOP_N        007B
   ZHTH4         007C
   ZHTH3         007D
   ZHSC_N        007D
   ZHTH2         007E
   ZHTRC         007E
```

```
 ZHTH1          007F
 ZHMCL          007F
 ZHISAZ         0080
 ZHIVBS         0080
 ZHTVBS         0080
*ZV$F           13DD
 ZV$F           13DD
*ZV$IH          13EB
 ZV$ID          13F0
 ZV$IH          13EB
 ZV$IAD         13F5
 ZV$__2         140D
 ZV$__3         141F
*ZV$TH          1484
 ZV$TH          1484
 ZV$TD          14B9
 ZV$THZ         14AC
*ZV$IA          14D4   REV. 7
 ZV$IA          14D7
 ZV$ARG         1586
 ZV$ABF         1588
 ZV$__1         1543
 ZV$IAV         14D5
*ZV$PCH         1593
 ZV$PCH         1593
*ZV$MLW         1695   REV. 0
 ZV$MLW         1695
 ZV$MLR         16C4
*ZV$T           16DC   REV. 5.0
 ZV$QC          16F9
 ZV$TC          16E5
 ZV$T           16DC
 ZV$Q           16EE
*ZV$C           170D   REV. 5
 ZV$C           170D
 ZV$CO          1730
*ZV$ER          1741   REV. 5.0
 ZV$ER          1741
 ZV$TA          176D
 ZV$__0         1754
*ZV$GP          17B1
 ZV$GP          17B1
 ZV$__4         17D1
*ZV$HA          17DD
 ZV$HA          17DD
 ZV$HZ          17E7
 ZV$HS          17E2
*ZV$HD          1816
 ZV$HD          1816
*ZV$BRK         1848
 ZV$BRK         1848
*ZV$RD          1862   REV. 7
 ZV$IZ          189C
 ZV$TTY         1875
 ZV$RD          1862
 ZV$SV1         1A37
 ZV$SV3         1A57
 ZV$AF          1873
 ZV$SV2         1A47
 ZV$TID         1874
```

```
   ZV$CF2       187E
   ZV$TK        187A
   ZV$RAR       187B
   ZV$ST1       187F
   ZV$RCC       1880
   ZV$BUD       1876
   ZV$OLB       1882
   ZV$RCB       1883
   ZV$NSR       1887
   ZV$STR       1885
   ZV$BKF       188A
   ZV$OTP       1909
   ZV$BKS       1889
   ZV$HR        1891
   ZV$LR        188E
   ZV$DAT       1871
   ZV$HM        18D8
   ZV$HRU       188B
   ZV$HRL       188C
   ZV$LRU       188D
   ZV$LRL       188E
   ZV$HBD       188F
   ZV$CF1       187D
   ZV$__5       1894
   ZV$RMD       1872
   ZV$MCP       1890
   HIBAUD       188F
   ZV$RAW       187C
   ZV$RDT       1A93
   ZV$CTL       1879
   ZV$B1        19B4
   ZV$TST       1AE9
   ZV$MDC       1ABD
   ZV$R99       1CBB
   ZV$ISA       1897
   ZV$UIH       1892
   ZV$ZRO       1916
   ZV$BSH       1918
   ZV$CPU       1878
   ZV$R50       18F6
   ZV$R60       1901
   ZV$RT        1BF8
   ZV$ALL       1877
  *MLCHPG       1CC0   T+V
   MLCHPG       1CC0
   ENDCHP       1CF1

**UNDEF
   STRTD2


**UNLINK MODULE(S)
  ZV$QC
  ZV$ID
  ZV$TC
  ZV$TD
  ZV$CO
  ZV$MLR
```

ZV$THZ