```
000001                          TITLE DCMC2, 'REV A'     CLA TEST (SAF)
000002                         *
000003                         *
000004                         * DESCRIPTION:
000005                         *
000006                         *   THIS T&V PROGRAM VERIFIES PROPER OPERATION OF THE LEVEL 6 COMMUNICATION LINE
000007                         *   ADAPTERS, WHICH ARE ATTACHMENTS TO THE DUALLINE COMMUNICATION PTOCESSOR.
000008                         *
000009                         *   THE SUBSYSTEM ITEMS SUPPORTED BY THIS PROGRAM ARE;
000010                         *
000011                         *       DLC9101 DUALLINE COMMUNICATION PROCESSOR
000012                         *       DLC9102 DUALLINE COMMUNICATION PROCESSOR
000013                         *       DLC9103 DUALLINE COMMUNICATION PROCESSOR
000014                         *
000015                         * REVISION HISTORY;
000016                         *
000017                         *       REV     DATE
000018                         *        A      FEB 78    ORIGINAL RELEASE
000019                         *
000020                         ********************************************************************************
000021                         *    THIS  DOCUMENT  AND  THE   INFORMATION   CONTAINED
000022                         *    THEREIN IS CONFIDENTIAL AND PROPRIETARY TO AND THE
000023                         *    EXCLUSIVE   PROPERTY   OF    HONEYWELL INFORMATION
000024                         *    SYSTEMS INC.  IT  IS MADE AVAILABLE ONLY TO HONEY-
000025                         *    WELL AUTHORIZED RECIPIENTS FOR THEIR USE SOLELY IN
000026                         *    THE  MAINTENANCE  AND   OPERATION   OF   HONEYWELL
000027                         *    PRODUCTS.  THIS  DOCUMENT  AND INFORMATION MUST BE
000028                         *    MAINTAINED IN STRICTEST CONFIDENCE;  IT  MUST  NOT
000029                         *    BE  REPRODUCED  IN  WHOLE OR IN PART; AND IT SHALL
000030                         *    NOT BE DISCLOSED TO ANY OTHER  PARTY  WITHOUT  THE
000031                         *    PRIOR WRITTEN CONSENT OF HONEYWELL.
000032                         *
000033                         * PROGRAM PREPARATION;
000034                         *
000035                         *
000036                         *    THE ROOT SOURCE OF THIS PROGRAM, AFTER THE ADDITION OF APPROPRIATE
000037                         *    TITLE AND END STATEMENTS, WAS PROCESSED BY THE HOST RESIDENT ASSEMBLER
000038                         *    TO CREATE EITHER SHORT OR LONG ADDRESS FORM (SAF OR LAF) OBJECT TEXT
000039                         *    AND LISTING. THE OBJECT TEXT WAS FURTHER PROCESSED BY THE HOST RESIDENT
000040                         *    LINKER USING THE APPROPRIATE CONSOLE ZV$LIB LIBRARY TO CREATE A PUNCH
000041                         *    SEGMENT CONTAINING AN EXECUTABLE MODULE. THE ASSEMBLY LISTING WAS
000042                         *    AUGMENTED WITH CROSS REFERENCE DATA PLUS THE LOAD MAP FROM THE LINKER
000043                         *    TO CREATE A LIST SEGMENT.
000044                         *
000045                         *                 ROOT           SAF            LAF
000046                         *                 ----           ---            ---
000047                         *       NAME      DCMX1          DCMS1          DCML1
000048                         *       DOCUMENT XXXXXXXX-001    XXXXXXXX-001   XXXXXXXX-001
000049                         *
000050                         *
000051                         * PROGRAM DISTRIBUTION;
000052                         *
000053                         *
000054                         *    THE ELEMENTARY ITEMS SUBMITTED TO THE T&V PROGRAM DISTRIBUTION CENTER
000055                         *    WERE THE EXECUTABLE PUNCHED CARD DECKS OF DCMS1 AND DCML1, AND MAGNETIC
000056                         *    TAPE IMAGES OF THE AUGMENTED LISTINGS.
000057                         *
000058                         *    REPRODUCTIONS OF THE EXECUTABLE CARD DECKS MAY BE AS DUPLICATE CARD DECKS
000059                         *    OR AS MEMBER "VL" OF A MULTIPLE MEMBER FILE. IN THE MOST FREQUENT CASE
000060                         *    IT WILL BE FOUND AS MEMBER VL WITHIN FILE PROGFILE OF A DISKETTE VOLUME
000061                         *    ENTITLED DIAGS.
000062                         *
000063                         *    DISTRIBUTION OF THE LISTINGS, WHICH SHOULD BE AVAILABLE IF ANY COMPLEX
000064                         *    MAINTENANCE OR REPAIR IS TO BE PERFORMED, IS NORMALLY MADE AS A
000065                         *    PRINTED COPY.
000066                         *
000067                         *
000068                         * ROUTINE DEMONSTRATION;
000069                         *
000070                         *
000071                         *    A MINIMUM SATISFACTORY TEST FOR NORMAL OPERATION MAY BE OBTAINED BY
000072                         *    ANSWERING TH QUESTION 'NEXT' WITH AN  'A' AND COMPLETING ONE PASS.
000073                         *
000074                         *
000075                         * MAIN MEMORY REQUIREMENT;
000076                         *
000077                         *
000078                         *    THIS PROGRAM REQUIRES 8K WORDS OF MAIN MEMORY AND WILL USE
000079                         *    ALL OF AVAILABLE MEMORY THROUGH 8K WORDS.
000080                         *
000081                         ********************************************************************
000082                         *
000083                         *
000084                         * TEST PROCEEDURE
000085                         *
000086                         *
000087                         *   BEFORE RUNNING THE CLA TEST, THE MOTHER BOARD TEST (DCMC1)
000088                         *   SHOULD BE RUN TO MAKE SURE THE DLCP IS OPERATING CORRECTLY.
000089                         *
000090                         *   THE PROGRAM AUTOMATICALLY TEST ALL CLA'S WHICH ARE LISTED ABOVE
000091                         *   UNDER DESCRIPTION.  IT PROCEEDS TO EXERCISE ALL THE LINE ADAPTER
000092                         *   FUNCTIONAL TESTS IN LOOP BACK MODE.
000093                         *
000094                         *   THE TESTS ARE SET UP SO THAT ALL THE CLA'S ARE TESTED ACCORDING TO A
000095                         *   PARTICULAR FUNCTION.  SOME OF THE FUNCTIONAL TESTS INCLUDE CHARACTER
000096                         *   SIZE TEST, LINE SPEED TEST, PARITY AND CRC TEST, AND UNDERRUN AND
000097                         *   OVERRRN TEST.
000098                         *
000099                         *
000100                         *
000101                         * OPERATING INSTRUCTIONS
000102                         *
000103                         *
000104                         *     LOAD AND START (OR RESTART) THE PROGRAM. THE PROGRAM IDENTIFICATION WILL
000105                         *     BE DISPLAYED ON THE CONSOLE.  THE INITIAL START WILL ALSO DISPLAY:
000106                         *
000107                         *       THE ZV$LIB REVISION NUMBER
000108                         *       THE ADDRESS FORM (SAF OR LAF)
000109                         *       I/O  EQUIPMENT DETECTED IN THE SYSTEM
000110                         *       MEMORY SIZE
000111                         *
000112                         *     THIS DISPLAY MUST BE VERIFIED BY THE OPERATOR.   THIS DISPLAY IS OMITTED
000113                         *     ON RESTARTS.
```

```
000114                          *
000115                          *     THE CONSOLE SEARCH RULES ARE:  FIND THE CONSOLE WITH THE LOWEST  CHANNEL
000116                          *     NUMBER CONNECTED THRU A DMC  CONTROLLER.  IF THERE IS NO CONSOLE ON A
000117                          *     DMC, THEN SEARCH FOR A TERMINAL WITH THE HIGHEST CHANNEL NUMBER ASSIGNED
000118                          *     TO AN ACLA ADAPTER ON A DLC CONTROLLER.  IF NO ASYNC ADAPTER  IS FOUND,
000119                          *     THEN GO TO THE FULL CONTROL PANEL.
000120                          *
000121                          *     THERE ARE THREE CONSOLE CHANNEL OPTIONS DETERMINED BY THE VALUE  OF  LO-
000122                          *     CATION "ZV$ITY".
000123                          *
000124                          *     IF ZV$TTY EQUALS (0000), SEARCH FOR A CONSOLE.
000125                          *     IF ZV$TTY EQUALS (FFFF), ASSUME THERE IS NO CONSOLE.
000126                          *     IF ZV$TTY EQUALS NEITHER (0000), NOR (FFFF), THEN IT IS THE CONSOLE CHAN-
000127                          *     NEL NUMBER.   NOTE:  DEFAULT IS TO SEARCH FOR A CONSOLE.
000128                          *
000129                          *     ALL CONSOLE I/O IS EVEN PARITY.  IF CONSOLE IS ON DLC,  IT MUST BE ASYNC
000130                          *     AND THE BAUD RATE SET AT 1200 TO MATCH THE PROGRAM SUPPLIED RATE.  IF IT
000131                          *     IS NECESSARY TO CHANGE THE PROGRAM BAUD RATE,   THEN THE  NEW  BAUD RATE
000132                          *     CODE SHOULD BE PUT INTO LOCATION "ZV$BUD" IN HEX. THE TERMINAL BAUD RATE
000133                          *     MUST BE  SET  TO MATCH THIS NEW BAUD RATE.  THE CORRECT HEX VALUE MAY BE
000134                          *     OBTAINED FROM THE FOLLOWING TABLE.
000135                          *
000136                          *-----------------------------------------*
000137                          *                                         *
000138                          *        BAUD  RATE  TABLE                *
000139                          *                                         *
000140                          *-----------------------------------------*
000141                          * ACLA I.D.E(2118)(2110) E(2108)
000142                          * BAUD-RATE
000143                          *    50    E   0 EE   1
000144                          *    75    E   1  EE   2
000145                          *   110    E   2 EE   3
000146                          *   134    E   3 EE   4
000147                          *   150    E   4 EE   5
000148                          *   200    E   5 EE  ---
000149                          *   300    E   6 EE   6
000150                          *   600    E   7 EE   7
000151                          *   900    E  --- EE   8
000152                          *  1050    E   8 EE  ---
000153                          *  1200    E   9 EE   9
000154                          *  1800    E  10 (A) EE  10 (A)
000155                          *  2000    E  11 (B) EE  ---
000156                          *  2400    E  12 (C)  E  11 (B)
000157                          *  3600    E  --- EE  12 (C)
000158                          *  4800    E  13 (D) EE  13 (D)
000159                          *  7200    E  --- EE  14 (E)
000160                          *  9600    E  14 (E) EE  15 (F)
000161                          * 19200    E  15 (F) EE  ---
000162                          *
000163                          *     TO MAKE ANY OF THE ABOVE CHANGES,  LOAD AND HALT THE PROGRAM BEFORE EX-
000164                          *     ECUTION.  INSERT CHANGE THEN EXECUTE.  MEMORY LOCATIONS OF "ZV$TTY" AND
000165                          *     "ZV$BUD" MAY BE FOUND IN MAP AT END OF LISTING.
000166                          *     CONSULT LEVEL-6 T&V MANUAL "AW94" FOR DETAILS ON HOW TO LOAD THE TESTS.
000167                          *
000168                          *     THE FOLLOWING IS A TYPICAL RESULT OF LOADING  AND  STARTING TO  RUN THE
000169                          *     PROGRAM.
000170                          *
000171                          *     CLA TEST <PGM NAME> <PGM DATE> <PGM REV>
000172                          *     ZV$LIB REV. 6.0
000173                          *     ZV$AF= 1   <2>
000174                          *     WDT
000175                          *     CHAN  DEVC  ID
000176                          *     0400  DSKT  2010
000177                          *     0480  DSKT  2010
000178                          *     0580  CDR   2008
000179                          *     1200  DISC  2330
000180                          *     1280  DISC  2330
000181                          *     1300  LPT   2000
000182                          *     1380  CONS  2019
000183                          *     MEMORY LOW  00002B2D
000184                          *     MEMORY HIGH 00003FFF  16K
000185                          *
000186                          *   THE PROGRAM WILL THEN ASK:
000187                          *
000188                          *              POWER LINE FREQ. ?:
000189                          *
000190                          *    THE OPERATOR NOW ENTERS THE POWER LINE FREQUENCY IN DECIMAL
000191                          *    FOLLOWED BY A CARRIAGE RETURN.  THE DEFAULT IS 60 HERTZ,
000192                          *    IF THIS IS DESIRED A CARRIAGE RETURN IS ENTERED.
000193                          *
000194                          *    NEXT THE PROGRAM WILL DISPLAY:
000195                          *
000196                          *             DLC CHANNEL(S) ?:
000197                          *
000198                          *    AT THIS TIME THE CHANNEL(S) OF THE LINE(S) YOU WANT
000199                          *    TO TEST SHOULD BE ENTERED,FOLLOWED BY A CARRIAGE RETURN.
000200                          *    THE CLA CHANNEL(S) WILL NORMALLY BE DISPLAYED AS AN ITEM(S) IN THE
000201                          *    CONFIGURATION PRINTOUT.
000202                          *
000203                          *    THE PROGRAM WILL THEN RESPOND WITH:
000204                          *
000205                          *                         NEXT ?:
000206                          *
000207                          *    THE OPERATOR HAS TO ENTER ONE OF THE FOLLOWING LETTERS:
000208                          *
000209                          *           "A" - RUN LINE(S) WITH INTERNAL LOOP TESTS
000210                          *
000211                          *           "C" - RUN LINE(S) WITH CONNECTOR LOOP TESTS
000212                          *
000213                          *    IN ORDER TO RUN WITH "C" SELECTED, AN EXTERNAL LOOP CONNECTOR
000214                          *    MUST BE ATTACHED TO THE LINE ADAPTER(S) EITHER AT THE
000215                          *    LINE ADAPTER CONNECTOR OF THE END OF THE CABLE.  THE REQUIRED
000216                          *    EXTERNAL LOOP CONNECTOR AND INSTRUCTIONS FOR ITS USE ARE
000217                          *    DESCRIBED IN EACH LINE ADAPTER'S PRODUCT MANUAL.
000218                          *
000219                          *    IF THERE  IS NO CONSOLE PRESENT REFER TO "SERIES 60
000220                          *    LEVEL 6 MODEL 34/36 T & V OPERATING INSTRUCTION
000221                          *    MANUAL" ORDER NO. FL39 FOR INSTRUCTIONS ON ENTERING DATA
000222                          *    AND INTERPRETING PROGRAM MESSAGES.
000223                          *
000224                          *    THE PROGRAM WILL RUN APPROXOMATELY 1 MINUTE PER ASYNCHRONOUS LINE AND
000225                          *    30 SECONDS PER SYNCHRONOUS LINE.
000226                          *
```

```
000227                          *   DURING THE LINE SPEED TEST THE PROGRAM WILL PRINT OUT THE SPEED AT WHICH
000228                          *   THE SYNCHRONOUS LINE(S) ARE TESTED AT WITHIN 3% OF THE ACTUAL SPEED.
000229                          *   ASYNCHRONOUS LINE(S) ARE TESTED AT ALL SELECTABLE BAUD RATES.
000230                          *
000231                          *   IF THERE ARE NO FAULTS. IT WILL THEN DISPLAY;
000232                          *
000233                          *        PASS
000234                          *
000235                          *   THE PROGRAM WILL CONTINUE AND TYPE "PASS" WITH NO HALTS BETWEEN PASSES.
000236                          *   TO RESTART; START THE PROGRAM AT HEX 100
000237                          *
000238                          *
000239                          * ERROR REPORTS
000240                          *
000241                          *   ERRORS WILL CAUSE THE PROGRAM TO HALT. AN ERROR MESSAGE WILL
000242                          *   BE DISPLAYED IF A CONSOLE IS PRESENT.
000243                          *
000244                          *   ERROR DISPLAYS ARE AS FOLLOWS;
000245                          *
000246                          *            ERR YP @ ZZZZ
000247                          *
000248                          *            B7  B6  B5  B4  B3  B2  B1  I
000249                          *            R7  R6  R5  R4  R3  R2  R1  M
000250                          *
000251                          *        Y    = TEST TYPE
000252                          *        P    = LINE NUMBER
000253                          *        ZZZZ = ERROR LOCATION IN LISTING. HAS COMMENT
000254                          *               GIVING FAILING FUNTION.
000255                          *
000256                          *        B1-B7, I, R1-R7, M ARE  CONTENTS OF REGISTERS.
000257                          *        INTERPERTATION OF THESE IS FOR SPECIALIST USAGE.
000258                          *
000259                          *        IN ALL CASES;
000260                          *
000261                          *             R3 = CHANNEL NUMBER
000262                          *
000263                          *        IN GENERAL
000264                          *
000265                          *             R6 = SHOULD BE DATA
000266                          *             R5 = ACTUAL DATA
000267                          *             R7 = WORD NUMBER IN BLOCK TRANSFER
000268                          *
000269                          *
000270                          ************************************************************************
```

```
000271                                  /
000272          0000                    ZERO    EQU     $
000273                                          XLOC    ZHNTSA,ZHTH15,ZV$HR
000274                                          XLOC    ZHISA2,ZHIAFB
000275                                          XLOC    ZHWDTC,ZHRTCC,ZV$TTY
000276                                          XLOC    ZHRTCI,ZHRTCL
000277                                          XLOC    ZHPFR,ZHCOMM
000278  00FF                                    ORG     ZERO+X'FF'
000279  00FF    0000                    STOP    HLT                     NO ERROR FOUND IN DLCC
000280                                  *
000281                                  *               OPERATOR MUST ENTER CLA ADDRESS
000282                                  *
000283          0100                    STRT    EQU     $
000284  0100    0F8C                            B       >STP            BRANCH TO NORMAL RUN
000285  0105                                    ORG     ZERO+X'105'
000286                                          CALL    ZV$RD,ZV$IZ     INITIALIZE FOR NON-CONSOLE USE
        0105    FBF0 0001
        0107    D380 0000       X
000287  0109    8700 0000       X               CL      <ZV$TTY         CLEAR CONSOLE CHANNEL
000288  010B    0F87                            B       >NTYR
000289                          STP             CALL    ZV$RD,MESG
        010C    FBC0 0003
        010E    D380 0000       X
        0110    0F80
        0111    1300
000290  0112    8980 1116               NTYR    CMZ     <FWFG
000291  0114    09AC                            BNE     >EDLINT
000292                                  *
000293                                  *               ASK FOR POWER LINE FREQ.
000294                                  *
000295  0115    8C51                            STS     =$R1            ASK ONLY ON 6/36 AND 3/34
000296  0116    82D1                            LB      =$R1,=Z'2000'
        0117    2000
000297  0118    050D                            BBT     >CP40
000298                                          CALL    ZV$QC,MESG9
        0119    FBC0 0003
        011B    D380 0000       X
        011D    0F80
        011E    1339
000299                                          CALL    ZV$1D,HRTZ
        011F    FBC0 0003
        0121    D380 0000       X
        0123    0F80
        0124    1119
000300                                  *
000301                                  *               FIND NUMBER OF MILSEC. IN RTC TICK
000302                                  *
000303  0125    8756                    CP40    CL      =$R6
000304  0126    F870 C350                       LDR     $R7,=Z'C350'
000305  0128    F300 1119                       DIV     $R7,<HRTZ
000306  012A    EF00 1125                       STR     $R6,<TLOC
000307  012C    8756                            CL      =$R6
000308  012D    F370 0064                       DIV     $R7,=100
000309  012F    FF00 1110                       STR     $R7,<DIV1
000310  0131    70D0                            DOR     $R7,16
000311  0132    F370 000A                       DIV     $R7,=10
000312  0134    FF00 1111                       STR     $R7,<DIV2
000313  0136    EF00 1112                       STR     $R6,<DIV3
000314  0138    8756                            CL      =$R6
000315  0139    F800 1125                       LDR     $R7,<TLOC
000316  013B    7F0A                            MLV     $R7,=10
000317  013C    F300 1119                       DIV     $R7,<HRTZ
000318  013E    FF00 1113                       STR     $R7,<DIV4
000319                                  *
000320                                  *
000321                                  *
000322                                  *
000323                                  *               INITIALIZE  THE ACTIVE LINE TABLE
000324                                  *                  INITIALIZE THE LINE ADDRESS TABLE
000325                                  *
000326                          EDLINT  CALL    ZV$F,ATLT,CPFLG+3,C16
        0140    FBC0 0003
        0142    D380 0000       X
        0144    0F80
        0145    1186
        0146    119A
        0147    125C
000327  0148    FE00 1108                       SWR     $R7,<CHANL
000328                                          CALL    ZV$QC,MESG7
        014A    FBC0 0003
        014C    D380 0000       X
        014E    0F80
        014F    132B
000329                                          CALL    ZV$IH,ADDRS,C8
        0150    FBC0 0003
        0152    D380 0000       X
        0154    0F80
        0155    118E
        0156    125B
000330  0157    8752                            CL      =$R2            CLEAR INDEX
000331  0158    AB80 118E                       LAB     $B2,<ADDRS      LOAD $B2 WITH ADDRESS OF ACTIVE LINE TABLE
000332  015A    CB80 125F                       LAB     $B4,<T5A1       SET UP FOR TRAP ON
000333  015C    CF80 0000       X               STB     $B4,<ZHNTSA     UNAVAILIABLE RESOURCE
000334  015E    CB80 0192                       LAB     $B4,<NODEVF
000335  0160    CF80 0000       X               STB     $B4,<ZHTH15
000336  0162    986E                    SALT    LDR     $R1,$B2,+$R2    LOAD ADDRESS FROM TABLE
000337  0163    1931                            BEZ     $R1,>CON3
000338  0164    9570 FFC0                       AND     $R1,=Z'FFC0'    STRIP CHANNEL WITH HEX FFC0
000339  0166    9970 03C0                       CMR     $R1,=X'3C0'     COMPARE CHANNEL WITH HEX 3C0
000340  0168    0302                            BG      >RCK1           BRANCH IF GREATER
000341  0169    0F96                            B       >DEVNF          CHANNEL NUMBER TO SMALL
000342  016A    1E26                    RCK1    ADV     $R1,=X'26'      PUT FUNCTION IN
000343  016B    8055                            IO      =$R5,=$R1       INPUT DEVICE NUMBER
        016C    0051
000344  016D    C855                            LDR     $R4,=$R5
000345  016E    D570 FF00                       AND     $R5,=Z'FF00'
000346  0170    D970 3100                       CMR     $R5,=Z'3100'    COMPARE DEVICE NUMBER RESPONSE
000347  0172    098D                            BNE     >DEVNF
000348  0173    B851                            LDR     $R3,=$R1        SET $R3 FOR INDEX OF LINE NUMBER
000349  0174    B570 03C0                       AND     $R3,=X'3C0'
000350  0176    3047                            SOR     $R3,7
000351  0177    CF30 1186                       STR     $R4,<ATLT,$R3   STORE ID INTO ACTIVE LINE TABLE
000352  0179    9570 FC00                       AND     $R1,=Z'FC00'
```

```
000353  017B  9F00 1108            STR   $R1,<CHANL
000354  017D  0F80 0162            B     <SALT
000355                        *
000356                        DEVNF CALL  ZV$TC,ERMG1          DLCC NOT FOUND ON THIS ADDRESS
        017F  FBC0 0003
        0181  D380 0000       X
        0183  0F80
        0184  1366
000357  0185  88D2                  DEC   =$R2
000358  0186  9820 118E            LDR   $R1,<ADDRS,$R2       STORE ADDRESS THAT FAILED
000359  0188  9F00 1125            PDRS  STR   $R1,<TLOC
000360                              CALL  ZV$TH,TLOC           PRINT ADDRESS THAT FAIL
        018A  FBC0 0003
        018C  D380 0000       X
        018E  0F80
        018F  1125
000361  0190  0F80 0140            B     <EDLINT
000362                        *
000363  0192  8755                NODEVF CL     =$R5           TRAP HANDLER ROUTINE
000364  0193  0003                  RTT
000365                        *
000366                        *
000367                        *
000368                        *
000369                        *     CHECK TO SEE IF THIS IS A CLA I CAN TEST
000370                        *
000371  0194  8751                CON3  CL    =$R1             CLEAR INDEX
000372  0195  AB80 1186            LAB   $B2,<ATLT
000373  0197  9B80 117F            LAB   $B1,<BIT0
000374  0199  2CF6                CAR   LDV   $R2,=-10
000375  019A  C85E                RON   LDR   $R4,$B2,+$R1
000376  019B  1D09                  CMV   $R1,=9
000377  019C  0910                  BE    >CON3A
000378  019D  497D                  BEZ   $R4,>RON
000379  019E  C921                CYN   CMR   $R4,$B1,$R2
000380  019F  097A                  BE    >CAR
000381  01A0  27FE                  BINC  $R2,>CYN
000382                              CALL  ZV$TC,ERMG2
        01A1  FBC0 0003
        01A3  D380 0000       X
        01A5  0F80
        01A6  1376
000383  01A7  88D1                  DEC   =$R1
000384  01A8  1007                  SOL   $R1,7
000385  01A9  9400 1108            OR    $R1,<CHANL
000386  01AB  0FDD                  B     >PDRS
000387                        *
000388  01AC  2CF4                CON3A LDV   $R2,=-12         NUMBER OF I/O TO BE CHANGED
000389  01AD  9800 118E            LDR   $R1,<ADDRS           LOAD $R1 WITH ADDRESS OF CLA ACTIVE
000390  01AF  DB80 1106            LAB   $B5,<FID+1           PUT ADDRESS OF CONTROL TABLE IN $B5
000391  01B1  C870 003F            ALL   LDR   $R4,=X'3F'     LOAD MASK TO CLEAR CHANNEL
000392  01B3  9570 FC00            AND   $R1,=Z'FC00'         CLEAR SUBCH. & FUNCTION
000393  01B5  C525                  AND   $R4,$B5,$R2        CLEAR CHANNEL
000394  01B6  9454                  OR    $R1,=$R4           PUT CHANNEL NUMBER IN
000395  01B7  9F25                  STR   $R1,$B5,$R2        STORE IT BACK IN CONTROL TABLE
000396  01B8  27F9                  BINC  $R2,>ALL
000397  01B9  8700 111F            CL    <PSFG              CLEAR PRINT SYNC. SPEED FLAG
000398  01BB  8700 1117            CL    <HGSPD             CLEAR HIGH SPEED FLAG
000399  01BD  8740 0F5A            CL    HISPD
000400  01BF  F900 1108            CMR   $R7,<CHANL         CHECK TO ONLY PRINT ONCE
000401  01C1  091D                  BE    >INVD
000402                        *
000403                        *           PRINT FIRMWARE REV.
000404                        *
000405  01C2  3001                  SOL   $R3,1              SHIFT $R3 TO ALIGN WITH CHANNEL NUMBER
000406  01C3  C380 0AF3            LNJ   $B4,<GENITZ        INITIALIZE CONTROLLER
000407  01C5  E380 0B5E            LNJ   $B6,<RDLCT         READ LOCATION 1 IN LCT TO PRINT REV.
000408  01C7  0000                  DC    0                  ADDRESS
000409  01C8  0002                  DC    2                  RANGE
000410                        *
000411  01C9  8780 112E            CLH   <LCTV
000412  01CB  C800 112E            LDR   $R4,<LCTV
000413  01CD  CF00 11E1            STR   $R4,<FRIMRV        STORE FIRMWARE REV NUMBER
000414  01CF  490F                  BEZ   $R4,>INVD
000415                              CALL  ZV$TC,FREV
        01D0  FBC0 0003
        01D2  D380 0000       X
        01D4  0F80
        01D5  135C
000416                              CALL  ZV$TD,LCTV
        01D6  FBC0 0003
        01D8  D380 0000       X
        01DA  0F80
        01DB  112E
000417  01DC  8A80 1116            INC   <FWFG              SET FLAG TO INHIBIT PRINT OUT
000418                        *
000419                        *           ASK WHAT MODE OF OPERATION
000420                        *
000421                        INVD  CALL  ZV$UC,MESG6
        01DE  FBC0 0003
        01E0  D380 0000       X
        01E2  0F80
        01E3  1328
000422                              CALL  ZV$IA,TP2,TLOC
        01E4  FBC0 0003
        01E6  D380 0000       X
        01E8  0F80
        01E9  1127
        01EA  1125
000423  01EB  7C08                  LDV   $R7,=8
000424  01EC  D380 0B4C            LNJ   $B5,<TDLAY         DELAY FOR DLC CONSOLE
000425  01EE  C800 1125            LDR   $R4,<TLOC
000426  01F0  4048                  SOR   $R4,8              SHIFT MODE TYPE ANSWER OVER
000427  01F1  4D53                  CMV   $R4,=X'53'         IF 'S' DO SHORT TEST ONLY
000428  01F2  0900 021E            BE    <SHTST
000429  01F4  4D43                  CMV   $R4,=X'43'         IF 'C' DO CABLE LOOP TEST ONLY
000430  01F5  0900 022F            BE    <CABLP
000431  01F7  4D4D                  CMV   $R4,=X'4D'         IF 'M' GO TO PATCH ROUTINE
000432  01F8  0900 0218            BE    <PCH
000433  01FA  4D41                  CMV   $R4,=X'41'         IF 'A' DO INTERNAL LOOP TEST ONLY
000434  01FB  09E3                  BNE   >INVD
000435                        *
000436  01FC  8700 110B            CL    <COUNT
```

```
000437  01FE  8700 112A            CL    <SHORT
000438  0200  C870 C714      RTST  LDR   $R4,=Z'C714'      STORE LINE CONTROL FOR INTERNAL LOOP
000439  0202  CF00 128B            STR   $R4,<LC1LCR
000440  0204  A380 0246            LNJ   $B2,<LPTA         TEST A:  GO TO LOOP TEST
000441  0206  A380 02AF            LNJ   $B2,<SPSYTC       TEST B:  GO TO LINE SPEED TEST
000442  0208  A380 03F1            LNJ   $B2,<CRPTB        TEST D:  GO TO CRC AND PARITY TEST
000443  020A  A380 04D5            LNJ   $B2,<CHSZTD       TEST E:  GO TO CHARACTER SIZE TEST
000444  020C  A380 052D            LNJ   $B2,<STBITE       TEST F:  GO TO STOP BIT TEST
000445  020E  A380 05B6            LNJ   $B2,<PARERF       TEST G:  GO TO PARITY ERROR TEST
000446  0210  A380 0617            LNJ   $B2,<STIOG        TEST H:  GO TO STOP I/O TEST
000447  0212  A380 06EA            LNJ   $B2,<UORNTH       TEST J:  GO TO OVERRUN AND UNDERRUN TEST
000448  0214  A380 0BE1            LNJ   $B2,<EDPCK
000449  0216  0F80 0200            B     <RTST
000450                        *
000451                        PCH   CALL  ZV$PCH
        0218  FBF0 0001
        021A  D380 0000       X
000452  021C  0F80 01DE            B     <INVD
000453                        *
000454  021E  C870 C714      SHTST LDR   $R4,=Z'C714'
000455  0220  CF00 128B            STR   $R4,<LC1LCR       STORE LINE CONTROL FOR INTERNAL LOOP
000456  0222  CF00 112A            STR   $R4,<SHORT        SET FLAG
000457  0224  A380 0246            LNJ   $B2,<LPTA         TEST A:  GO TO LOOP TEST
000458  0226  A380 02AF            LNJ   $B2,<SPSYTC       TEST B: LINE SPEED TEST TO SET HIGH SPEED FLA
000459  0228  A380 03F1            LNJ   $B2,<CRPTB        TEST D: CRC AND PARITY TEST
000460  022A  A380 05B6            LNJ   $B2,<PARERF       TEST G:  PARITY ERROR TEST
000461  022C  A380 0BE1            LNJ   $B2,<EDPCK        PRINT PASS MESSAGE
000462  022E  0FF0                 B     >SHTST
000463                        *
000464  022F  4C07            CABLP LDV   $R4,=7
000465  0230  CF00 110B            STR   $R4,<COUNT
000466  0232  8700 112A            CL    <SHORT
000467  0234  C870 C314      RTST1 LDR   $R4,=Z'C314'      STORE LINE CONTROL FOR CONNECTOR LOOP
000468  0236  CF00 128B            STR   $R4,<LC1LCR
000469                        *
000470  0238  A380 0246            LNJ   $B2,<LPTA         TEST A:  GO TO LOOP DATA TEST
000471  023A  A380 0777            LNJ   $B2,<LPCBLK       TEST K:  GO TO LOOP DSS IN DSC
000472  023C  A380 080D            LNJ   $B2,<XMMSL        TEST L:  GO TO XMIT MARK & SPACE TEST
000473  023E  8880 110B            DEC   <COUNT
000474  0240  8980 110B            CMZ   <COUNT
000475  0242  09F2                 BNE   >RTST1
000476  0243  A380 0BE1            LNJ   $B2,<EDPCK
000477  0245  0FEA                 B     >CABLP
000478                        * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
000479                        *
000480                        *          DATA LOOP TEST
000481                        *
000482  0246  C870 4120      LPTA  LDR   $R4,=A'A '
000483  0248  CF00 1364            STR   $R4,<ERMG+1
000484  024A  C870 0E18            LDR   $R4,=X'E18'       STORE FAST LINE SPEED
000485  024C  CF00 128A            STR   $R4,<LCT1S
000486  024E  B380 098B            LNJ   $B3,<CLFG         CLEAR ALL TEST FLAGS
000487  0250  8700 111A            CL    <MASK             SET MASK FOR ZV$C
000488  0252  8700 128C            CL    <LC1SYF           CLEAR SYNC. FLAG FOR ASYNC. LINES
000489  0254  7C06                 LDV   $R7,=6            NUMBER OF 30 MS DELAY
000490                        *
000491                        *          TEST ASYNCHRONOUS LINES WITH BAUD RATE
000492                        *             AT 9600 BAUD, WITH 2 STOP BITS
000493                        *
000494  0255  9B80 1176      T1    LAB   $B1,<ASYN1        LINE TYPE
000495  0257  B380 0881      T1A   LNJ   $B3,<DTALOP
000496  0259  0F85                 B     >T2AA
000497  025A  12D4                 DC    <WCP1
000498  025B  1286                 DC    <LCT1
000499  025C  003F                 DC    (DFLT-DOUT)       RANGE IN WORDS
000500  025D  119E                 DC    <DOUT             TRANSMIT BUFFER ADDRESS
000501                        *
000502                        *          TEST CURRENT LOOP ASYNCHRONOUS
000503                        *
000504  025E  8980 110B      T2AA  CMZ   <COUNT            IF CONNECTOR  LOOP
000505  0260  0900 0266            BE    <T2               CHANGE SPEED TO TEST AT 9600 BAUD
000506  0262  C870 0E18            LDR   $R4,=X'E18'
000507  0264  CF00 128A            STR   $R4,<LCT1S
000508  0266  C871           T2    LDR   $R4,+$B1          LOAD ID TYPE INTO $R4
000509  0267  C900 1177            CMR   $R4,<ASYNC        CHECK TO SEE IF ALREADY DONE
000510  0269  0902                 BE    >T2A
000511  026A  0FED                 B     >T1A
000512        026B            T2A   EQU   $
000513  026B  8980 112A            CMZ   <SHORT
000514  026D  09A1                 BNE   >NCBL
000515  026E  C870 0118            LDR   $R4,=X'118'       CHANGE BAUD RATE TO 50
000516  0270  CF00 128A            STR   $R4,<LCT1S
000517  0272  7C78                 LDV   $R7,=120          NUMBER OF 30 MS DELAY FOR DATA TRANSFER
000518                        *
000519                        *          TEST ASYNCHRONOUS LINES WITH NEW BAUD RATE
000520                        *             AT 50 BAUD, WITH  2 STOP BITS
000521                        *
000522  0273  C870 0018      T3    LDR   $R4,=X'18'
000523  0275  CF00 128A            STR   $R4,<LCT1S
000524  0277  9B80 1176            LAB   $B1,<ASYN1        LOAD LINE TYPE
000525  0279  B380 0881      T3A   LNJ   $B3,<DTALOP       LOOP SUB-ROUTINE
000526  027B  0F85                 B     >T4
000527  027C  12D4                 DC    <WCP1
000528  027D  1286                 DC    <LCT1
000529  027E  0007                 DC    7
000530  027F  11AE                 DC    <DOUT1
000531                        *
000532                        *          TEST CURRENT LOOP ASYNCHRONOUS
000533                        *
000534  0280  C871           T4    LDR   $R4,+$B1          LOAD ID TYPE INTO $R4
000535  0281  C900 1177            CMR   $R4,<ASYNC        CHECK TO SEE IF ALREADY DONE
000536  0283  0902                 BE    >T4A
000537  0284  0FF5                 B     >T3A
000538        0285            T4A   EQU   $
000539  0285  C800 128B            LDR   $R4,<LC1LCR       CHECK LINE CONTROL FOR CONNECTOR LOOP
000540  0287  82D4                 LB    =$R4,=Z'0400'     IF CONNECTOR LOOP SET DIRECT CONNECT BIT
        0288  0400
000541  0289  0505                 BBT   >NCBL
000542  028A  C870 CB14            LDR   $R4,=Z'CB14'
000543  028C  CF00 128B            STR   $R4,<LC1LCR
000544  028E  C870 3A18      NCBL  LDR   $R4,=X'3A18'      PUT IN SYNC. CHARACTER TO TEST
000545  0290  CF00 128A            STR   $R4,<LCT1S        SYNC. LINE TYPES
000546  0292  C870 021C            LDR   $R4,=X'21C'       SET LC1 SYNC. FLAG
```

```
000547  0294  CF00 128C                STR   $R4,=<LC1SYF
000548  0296  7C2D                     LDV   $R7,=45             NUMBER OF 30 MS DELAY FOR 800 BAUD
000549                          *
000550                          *          TEST SYNCHRONOUS (3A AS SYNC. CHAR.)
000551                          *
000552  0297  9B80 1178          QA1   LAB   $B1,=<SYNC           LOAD LINE TYPE
000553  0299  B380 0881                LNJ   $B3,=<DTALOP        LOOP SUB-ROUTINE
000554  029B  0F85                     B     >T5
000555  029C  12D4                     DC    <WCP1
000556  029D  1286                     DC    <LCT1
000557  029E  002F                     DC    (DFLT-DOUT1)
000558  029F  11AE                     DC    <DOUT1
000559                          *
000560                          *          TEST BI-SYNCHRONOUS LINES (3A AS SYNC. CHAR.)
000561                          *
000562  02A0  C871                T5    LDR   $R4,=+$B1           LOAD ID TYPE INTO $R4
000563  02A1  C900 1179                CMR   $R4,=<BISYNC        CHECK TO SEE IF ALREADY DONE
000564  02A3  0902                     BE    >T6
000565  02A4  0FF5                     B     >QA1
000566                          *
000567                          *          TEST MIL 188 SYNCHRONOUS (3A AS SYNC. CHAR.)
000568                          *
000569  02A5  9B80 117A          T6    LAB   $B1,=<SYN188        LOAD LINE TYPE
000570  02A7  B380 0881                LNJ   $B3,=<DTALOP        LOOP SUB-ROUTINE
000571  02A9  0F85                     B     >T30
000572  02AA  12D4                     DC    <WCP1
000573  02AB  1286                     DC    <LCT1
000574  02AC  002F                     DC    (DFLT-DOUT1)
000575  02AD  11AE                     DC    <DOUT1
000576                          *
000577  02AE  8382                T30   JMP   $B2                 JUMP OUT OF LOOP TEST
000578                          *
000579                          * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
000580                          *
000581                          *          LINE SPEED TEST
000582                          *
000583  02AF  C870 4220          SPSYTC LDR   $R4,=A'B '
000584  02B1  CF00 1364                STR   $R4,=<ERMG+1
000585  02B3  B380 098B                LNJ   $B3,=<CLFG          CLEAR ALL TEST FLAGS
000586  02B5  C800 1119                LDR   $R4,=<HRTZ          CALCULATE THE LOW AND HIGH RANGE OF RTC TICKS
000587  02B7  4002                     SOL   $R4,=2
000588  02B8  4EF8                     ADV   $R4,=-8
000589  02B9  CF00 11DF                STR   $R4,=<SECL
000590  02BB  4E10                     ADV   $R4,=16
000591  02BC  CF00 11E0                STR   $R4,=<SECH
000592  02BE  C800 1119                LDR   $R4,=<HRTZ          CALCULATE NUMBER OF RTC TICKS IN 5 SEC.
000593  02C0  4F0A                     MLV   $R4,=10
000594  02C1  CF00 11DE                STR   $R4,=<FIVSEC
000595  02C3  CB80 02CD                LAB   $B4,=<SPSYT1        SET UP INTERRUPT VECTORS AND
000596  02C5  CF80 1281                STB   $B4,=<ISA4P         POINTERS TO CHANGE RUNNING LEVEL TO 4
000597  02C7  CB80 127E                LAB   $B4,=<ISA4+$AF
000598  02C9  CF80 0004       X        STB   $B4,=<ZHISAZ+4*$AF
000599  02CB  8E70 8004                LEV   =Z'8004'            CHANGE RUNNING LEVEL TO 4
000600  02CD  4C01               SPSYT1 LDV   $R4,=1              SET RTC LEVEL TO 1
000601  02CE  CF00 0000       X        STR   $R4,=<ZHRTCL
000602                          *
000603  02D0  CB80 03D6                LAB   $B4,=<NOSPD         SET RUPT POINTER FOR RTC INTERRUPT
000604  02D2  CF80 127B                STB   $B4,=<ISA1P
000605                          *
000606  02D4  CB80 1278                LAB   $B4,=<ISA1+$AF      SET RUPT VECTOR FOR LEVEL 1 RUPT
000607  02D6  CF80 0001       X        STB   $B4,=<ZHISAZ+$AF
000608                          *
000609  02D8  8980 112A                CMZ   <SHORT
000610  02DA  0996                     BNE   >STSP1
000611  02DB  4C0A                     LDV   $R4,=10             STORE CHAR. SIZE
000612  02DC  CF00 110D                STR   $R4,=<CSB
000613  02DE  8700 129C                CL    <LC3SYF             CLEAR SYNC. FLAG FOR ASYNC. LINES
000614  02E0  8700 1107                CL    <BYTP               SET BYTE POSITION TO ZERO
000615                          *
000616                          *        TEST LINE SPEEDS FOR ASYCHRONOUS NEW BAUD RATE
000617                          *
000618  02E2  4C18                     LDV   $R4,=X'18'          SET $R4 WITH FIRST SPEED TO TEST (ASYNC.)
000619  02E3  9B80 1176                LAB   $B1,=<ASYN1         LOAD LINE TYPE
000620  02E5  EB80 11F2                LAB   $B6,=<NSPTBL        LOAD ADDRESS OF CHAR. NUMBER TABLE
000621  02E7  B380 0318                LNJ   $B3,=<SPDTLP
000622                          *
000623                          *        TEST LINE SPEEDS FOR ASYCHRONOUS CURRENT LOOP
000624                          *
000625  02E9  4C18                     LDV   $R4,=X'18'
000626  02EA  9B80 1177                LAB   $B1,=<ASYNC
000627  02EC  EB80 11F2                LAB   $B6,=<NSPTBL
000628  02EE  B380 0318                LNJ   $B3,=<SPDTLP
000629                          *
000630  02F0  C870 0F00          STSP1 LDR   $R4,=3840           SET RANGE FOR SYNC. LINES
000631  02F2  CF00 1120                STR   $R4,=<RANGE
000632  02F4  EB80 1120                LAB   $B6,=<RANGE         LOAD ADDRESS OF RANGE FOR SPEED TEST SUB-ROUT
000633                          *
000634  02F6  CB80 03E0                LAB   $B4,=<IRRG          SET RUPT POINTER FOR RTC INTERRUPT (SYNC.)
000635  02F8  CF80 127B                STB   $B4,=<ISA1P
000636  02FA  8A80 1107                INC   <BYTP               SET BYTE POSITION TO ONE
000637                          *
000638  02FC  C870 021C                LDR   $R4,=X'21C'         SET LCT SYNC. LINE FLAG
000639  02FE  CF00 129C                STR   $R4,=<LC3SYF
000640  0300  4C08                     LDV   $R4,=8              SET CHAR. SIZE FOR SPEED CALCULATION
000641  0301  CF00 110D                STR   $R4,=<CSB
000642  0303  C870 C118                LDR   $R4,=Z'C118'        LOAD $R4 WITH SYNC CHARACTER TO BE USED
000643                          *
000644                          *        MEASURE AND PRINT LINE SPEED FOR BI-SYCHRONOUS
000645                          *
000646  0305  9B80 1179                LAB   $B1,=<BISYNC        LOAD ADDRESS OF LINE TYPE
000647  0307  B380 0318                LNJ   $B3,=<SPDTLP        GO TO LINE SPEED TEST
000648                          *
000649                          *        MEASURE AND PRINT LINE SPEED FOR SYCHRONOUS
000650                          *
000651  0309  9B80 1178                LAB   $B1,=<SYNC          LOAD ADDRESS OF LINE TYPE
000652  030B  B380 0318                LNJ   $B3,=<SPDTLP        GO TO LINE SPEED TEST
000653                          *
000654                          *        MEASURE AND PRINT LINE SPEED FOR MIL 188
000655                          *
000656  030D  9B80 117A                LAB   $B1,=<SYN188
000657  030F  B380 0318                LNJ   $B3,=<SPDTLP        GO TO LINE SPEED TEST
000658                          *
000659  0311  8A80 111F                INC   <PSFG               SET FLAG TO INHIBIT PRINTOUT
```

```
                                  CALL    ZV$RD               RE-INITIALIZE INTERRUPT VECTOR AND POINTERS &
 0313  FBF0 0001
 0315  D380 0000      X                                         .
                          *                                   SET LEVEL TO ZERO
 0317  8382                       JMP     $B2                 JUMP OUT OF TEST
                          *
                          * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                          *
                          *           LINE SPEED LOOP
                          *
 0318  8F00 1155          SPDTLP SAVE    <SAV3,=Z'0810'      SAVE REGS.
 031A  0810
 031B  8753                       CL      =$R3                "   $R3
 031C  9F80 0320                  STB     $B1,<ST1            STORE LINE TYPE FOR FIND LINE ON
                          *
 031E  D380 0B38          JPT3    LNJ     $B5,<FLN            FIND LINE ON OF TYPE BELOW
 0320  0000               ST1     RESV    $AF,0               LINE TYPE FOUND AND STORED HERE
 0321  8383                       JMP     $B3                 NO MORE LINE THIS TYPE END TEST
                          *
 0322  8751                       CL      =$R1                CLEAR $R1
 0323  8752                       CL      =$R2                CLEAR INDEX ON NEW LINE
 0324  CF00 129A          NXSPD   STR     $R4,<LCT3S          STORE LINE SPEED IN LCT TABLE
                          *
 0326  C808 0320          NXSPD1 LDR     $R4,*<ST1
 0328  C900 1177                  CMR     $R4,<ASYNC
 032A  0983                       BNE     >NXSPD2             B IF NOT CURRENT LOOP ADAPT.
 032B  C380 09FD                  LNJ     $B4,<SCLS           SET CURRENT LOOP ADAPTER  SPEED
 032D  C380 0AF3          NXSPD2 LNJ     $B4,<GENITZ         GENERAL INITIALIZE
 032F  C570 FF80                  AND     $R4,=Z'FF80'        PREPARE  TO WRITE CHANNEL PROG
 0331  C470 0040                  OR      $R4,=X'40'
 0333  CF00 110A                  STR     $R4,<CHNZ
                                  CALL    ZV$MLW,CHLP1,RAG1,HX200,CHNZ
 0335  FBC0 0003
 0337  D380 0000      X
 0339  0F80
 033A  0BF8
 033B  12D6
 033C  125D
 033D  110A
                                  CALL    ZV$MLW,CHLP1A,RAG2,HX400,CHNZ
 033E  FBC0 0003
 0340  D380 0000      X
 0342  0F80
 0343  0C34
 0344  12D7
 0345  125E
 0346  110A
                          *
 0347  D380 0B24                  LNJ     $B5,<SETLCT         SEND OUT LCT TABLE
 0349  1296                       DC      <LCT3
                          *
 034A  C826                       LDR     $R4,$B6,$R2         LOAD $R4 WITH RANGE IN BYTES
 034B  8AD3                       INC     =$R3
 034C  D380 0B82                  LNJ     $B5,<MCCB           FORM CCB FOR XMIT
 034E  119E                       DC      <DOUT
 034F  0060                       DC      X'60'
 0350  88D3                       DEC     =$R3                CHANGE TO RECV. SIDE
 0351  4900 03C4                  BEZ     $R4,<TSNL
 0353  CF55                       STR     $R4,=$R5
 0354  C970 03C0          DLCCB   CMR     $R4,=X'3C0'         MAKE MULTIPLE CCB IF LARGE RANGE
 0356  0A80 0362                  BALE    <SGCCB
 0358  C870 03C0                  LDR     $R4,=X'3C0'
 035A  D380 0B82                  LNJ     $B5,<MCCB
 035C  1C00                       DC      <DIN
 035D  0040                       DC      X'40'
 035E  88D1                       DEC     =$R1
 035F  D254                       SUB     $R5,=$R4
 0360  DF54                       STR     $R5,=$R4
 0361  0FF3                       B       >DLCCB
                          *
 0362  D380 0B82          SGCCB   LNJ     $B5,<MCCB           FORM CCB FO RECV.
 0364  1C00                       DC      <DIN
 0365  0060                       DC      X'60'
                          *
                          *
                          *       START I/O XMIT FIRST THEN RECV.
                          *
 0366  7C01                       LDV     $R7,=1
 0367  8AD3                       INC     =$R3                CHANGE TO XMIT SIDE
 0368  C800 10FC                  LDR     $R4,<FOCC
 036A  C380 0B7E                  LNJ     $B4,<CGSCH
 036C  8000 1180                  IO      <BIT1,=$R4
 036E  0054
 036F  0703                       BIOT    >MELCO1
 0370  F380 0BA3                  LNJ     $B7,<ERROR
 0372  D380 0B4C          MELCO1 LNJ     $B5,<TDLAY
 0374  88D3                       DEC     =$R3
                          *
 0375  88D1                       DEC     =$R1
 0376  C800 10FC                  LDR     $R4,<FOCC
 0378  C380 0B7E                  LNJ     $B4,<CGSCH
 037A  F800 11DE                  LDR     $R7,<FIVSEC         SYNC. UP CLOCK
 037C  FF00 0000      X           STR     $R7,<ZHRTCC         STORE RTCC WITH 5 SEC OF TICKS
 037E  0004                       RTCN                        TURN  ON REAL TIME CLOCK
 037F  F900 0000      X   SYCT    CMR     $R7,<ZHRTCC
 0381  027E                       BL      >SYCT
 0382  8040 0DFD                  IO      BIT1,=$R4           START I/O
 0384  0054
 0385  0703                       BIOT    >JPT1
 0386  F380 0BA3                  LNJ     $B7,<ERROR          ERROR;  I/O WAS NAK'ED
 0388  C380 0B0E          JPT1    LNJ     $B4,<INXT           INPUT NEXT STATUS
 038A  C800 1101                  LDR     $R4,<FIS            LOOP ON INPUT STATUS UNTIL STATUS COMPLETE
 038C  C380 0B7E                  LNJ     $B4,<CGSCH
 038E  8055               ST3     IO      =$R5,=$R4           INPUT STATUS
 038F  0054
 0390  0703                       BIOT    >JPT2
 0391  F380 0BA3                  LNJ     $B7,<ERROR          ERROR;  I/O WAS NAK'ED
 0393  82D5               JPT2    LB      =$R5,=Z'1000'       SEE IF STATUS COMPLETE
 0394  1000
 0395  0506                       BBT     >JPT15
 0396  D870 0100                  LDR     $R5,=Z'0100'
 0398  5701 FFFF                  BDEC    $R5,$               TIME DELAY FOR STATUS CHECK
 039A  0FF4                       B       >ST3
```

```
039B  17ED               JPT5   BINC   $R1,>JPT1           LOOP TO FINAL CCB
                         *
039C  0005                      RTCF                       TURN OFF REAL TIME CLOCK
039D  C800 11DE                 LDR    $R4,<FIVSEC         CALCULATE ELAPSED TIME
039F  C200 0000    X            SUB    $R4,<ZHRTCC
03A1  8755                      CL     =$R5
03A2  8980 1129                 CMZ    <SBTFG              CHECK FLAG FOR STOP BIT TIME TEST
03A4  0980 0577                 BNE    <SBTE
                         *
03A6  8980 129C          JPT4   CMZ    <LC3SYF             SEE IF SYNC. OR ASYNC. LINE
03A8  09A2                      BNE    >SYNSP
03A9  C900 11DF                 CMR    $R4,<SECL           SEE IF NUMBER OF TICKS WITHIN
03AB  0A84                      BALE   >TRYH               ACCEPTABLE RANGE
03AC  C900 11E0                 CMR    $R4,<SECH
03AE  080E                      BAL    >TYXL
03AF  F800 1114          TRYH   LDR    $R7,<ERF            CLEAR ERROR FLAG TO
03B1  8700 1114                 CL     <ERF                SUPPRESS REGISTER DUMP
03B3  F380 0BA3                 LNJ    $B7,<ERROR
03B5  C380 0A8D                 LNJ    $B4,<PRTLSP         ERROR:  INCORRECT LINE SPEED
03B7  FF00 1114                 STR    $R7,<ERF
03B9  4C0A                      LDV    $R4,=10
03BA  CF00 110D                 STR    $R4,<CSB
03BC  8AD2               TYXL   INC    =$R2
03BD  C800 129A                 LDR    $R4,<LCT3S          SET LCT TO NEXT SPEED
03BF  CA70 0100                 ADD    $R4,=X'100'         CHANGE TO NEXT SPEED
03C1  2D0F                      CMV    $R2,=15             SEE IF ALL SPEED TESTED
03C2  0800 0324                 BAL    <NXSPD              BRANCH TO TEST NEXT SPEED
                         *
03C4  3E02               TSNL   ADV    $R3,=2              TEST NEXT LINE
03C5  8F80 1155                 RSTR   <SAV3,=Z'0810'      RESTORE REGS.
03C7  0810
03C8  0F80 031E                 B      <JPT3
                         *
03CA  8980 111F          SYNSP  CMZ    <PSFG
03CC  0986                      BNE    >SYNSP1
03CD  8980 112A                 CMZ    <SHORT
03CF  0983                      BNE    >SYNSP1
03D0  C380 0A8D                 LNJ    $B4,<PRTLSP         PRINT SYNCHRONOUS LINE SPEED
03D2  4C08               SYNSP1 LDV    $R4,=8
03D3  CF00 110D                 STR    $R4,<CSB
03D5  0FEF                      B      >TSNL
                         *
                         *
                         *      RTC INTERRUPT ROUTINE
                         *
03D6  0005               NOSPD  RTCF                       TURN OFF RTC
03D7  8700 110D                 CL     <CSB
03D9  DB80 03A6                 LAB    $B5,<JPT4           SET RUPT POINTER TO RETURN TO LEVEL 4
03DB  DF80 1281          .      STB    $B5,<ISA4P
03DD  8E70 8004                 LEV    =Z'8004'
03DF  0FF7                      B      >NOSPD
                         *
                         *
                         * + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + + +
                         *        SYNC. RTC INTERRUPT ROUTINE
                         *
03E0  0005               IRRG   RTCF                       TURN OFF RTC
03E1  C380 0B14                 LNJ    $B4,<INRG           INPUT RANGE INTO $R5
03E3  8AD1                      INC    =$R1                FORM TOTAL RESIDUAL RANGE
03E4  1904                      SEZ    $R1,>IRRG2          FROM OTHER CCB'S
03E5  DA70 03C0          IRRG1  ADD    $R5,=X'3C0'
03E7  17FE                      BINC   $R1,>IRRG1
03E8  C800 11DE          IRRG2  LDR    $R4,<FIVSEC         PUT ELAPSED TIME IN $R4
03EA  DB80 03CA                 LAB    $B5,<SYNSP          SET RUPT POINTER FOR RETURN TO LEVEL 4
03EC  DF80 1281                 STB    $B5,<ISA4P
03EE  8E70 8004                 LEV    =Z'8004'
03F0  0FF0                      B      >IRRG
                         *
                         * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                         *
                         *      CRC AND PARITY TEST
                         *
03F1  C870 4420          CRPTB  LDR    $R4,=A'D '
03F3  CF00 1364                 STR    $R4,<ERMG+1
03F5  B380 098B                 LNJ    $B3,<CLFG           CLEAR ALL TEST FLAGS
03F7  8A80 110F                 INC    <CRCFG              SET FLAG FOR DTALOP SUB-ROUTINE
03F9  8700 111A                 CL     <MASK               SET MASK FOR ZVSC
03FB  C870 C802                 LDR    $R4,=Z'C802'        SET LCT CONFIG. FOR
03FD  CF00 1290                 STR    $R4,<LC2CFR         RECV., ODD PARITY, CRC16
03FF  4E20                      ADV    $R4,=X'20'
0400  CF00 1291                 STR    $R4,<LC2CFX         XMIT LCT CONFIG. ODD PARITY
0402  EB80 1197                 LAB    $B6,<CPFLG          LOAD ADDRESS OF LCT FLAG TABLE
0404  C876                      LDR    $R4,+$B6            LOAD $R4 WITH LCT FLAG
0405  CF00 1294          CRPTB1 STR    $R4,<LCT2FG         STORE INTO LCT TABLE
0407  F870 0E18                 LDR    $R7,=X'E18'         STORE INITIAL SPEED FOR ASYC. LINES
0409  FF00 1292                 STR    $R7,<LCT2S
040B  7C06                      LDV    $R7,=6              NUMBER OF 30 MS TIME DELAY FOR TRANSFER
                         *
                         *         TEST ASYCHRONOUS LINES WITH BAUD RATE
                         *           9600 BAUD, WITH 2 STOP BITS
040C  9B80 1176          T8     LAB    $B1,<ASYN1          LOAD  LINE TYPE
040E  B380 0881          T8A    LNJ    $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
0410  0F85                      B      >T9
0411  12D9                      DC     <WCP2               ADDRESS TABLE FOR BLOCK WRITE
0412  128E                      DC     <LCT2               LCT TABLE
0413  003F                      DC     (DFLT-DOUT)         RANGE IN WORDS
0414  119E                      DC     <DOUT
                         *
                         *         TEST CURRENT LOOP ASYNCHRONOUS
                         *
0415  C871               T9     LDR    $R4,+$B1            LOAD ID TYPE INTO $R4
0416  C900 1177                 CMR    $R4,<ASYNC          CHECK TO SEE IF ALREADY DONE
0418  0902                      BE     >T9A
0419  0FF5                      B      >T8A
      041A               T9A    EQU    $
041A  F870 3218                 LDR    $R7,=X'3218'        STORE SYNC. CHAR. INTO LCT TABLE
041C  FF00 1292                 STR    $R7,<LCT2S
041E  7C28                      LDV    $R7,=40             NUMBER OF 30 MS TIME DELAYS
                         *
                         *         TEST SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
                         *
041F  9B80 1178                 LAB    $B1,<SYNC           LOAD LINE TYPE
0421  B380 0881          QA2    LNJ    $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
```

```
000864  0423  0F85                        B     >T10
000865  0424  12D9                        DC    <WCP2              ADDRESS TABLE FOR BLOCK WRITE
000866  0425  128E                        DC    <LCT2              LCT TABLE
000867  0426  002F                        DC    (DFLT-DOUT1)       RANGE IN WORDS
000868  0427  11AE                        DC    <DOUT1
000869                              *
000870                              *          TEST BI-SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
000871                              *
000872  0428  C871                  T10   LDR   $R4,+$B1             LOAD ID TYPE INTO $R4
000873  0429  C900 1179                    CMR   $R4,<BISYNC          CHECK IO SEE IF ALREADY DONE
000874  042B  0902                        BE    >T12
000875  042C  0FF5                        B     >UA2
000876                              *
000877                              *          TEST MIL 188 SYNCHRONOUS (32 AS SYNC. CHAR.)
000878                              *
000879  042D  9B80 117A             T12   LAB   $B1,<SYN188          LOAD LINE TYPE
000880  042F  B380 0881                    LNJ   $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
000881  0431  0F85                        B     >T38
000882  0432  12D9                        DC    <WCP2              ADDRESS TABLE FOR BLOCK WRITE
000883  0433  128E                        DC    <LCT2              LCT TABLE
000884  0434  002F                        DC    (DFLT-DOUT1)       RANGE IN WORDS
000885  0435  11AE                        DC    <DOUT1
000886                              *
000887  0436  C876                  T38   LDR   $R4,+$B6             CHANGE LCT FLAG
000888  0437  4980 0405                    BNEZ  $R4,<CRPTB1          SEE IF ALL FLAGS CHECKED
000889  0439  C800 1290                    LDR   $R4,<LC2CFR          SEE IF ALL COMBINATIONS HAVE BEEN RUN
000890  043B  C970 D802                    CMR   $R4,=Z'D802'
000891  043D  090D                        BE    >CRPTB2              BRANCH IF ALL TESTED
000892  043E  C870 D802                    LDR   $R4,=Z'D802'         CHANGE LCT CONFIG. TO EVEN PARITY
000893  0440  CF00 1290                    STR   $R4,<LC2CFR          STORE RECV CONFIG.
000894  0442  4E20                        ADV   $R4,=X'20'           CHANGE ADDRESS FOR XMIT  LCT
000895  0443  CF00 1291                    STR   $R4,<LC2CFX          STORE XMIT CONFIG.
000896  0445  EB80 1198                    LAB   $B6,<CPFLG+1         SET TO TEST WITH PARITY
000897  0447  C876                        LDR   $R4,+$B6             LOAD $R4 WITH LCT FLAG
000898  0448  0F80 0405                    B     <CRPTB1              BRANCH TO TEST ALL LINES AGAIN WITH EVEN PARI
000899  044A  8700 110F             CRPTB2 CL    <CRCFG             CLEAR CRC AND PARITY TEST FLAG
000900  044C  8382                        JMP   $B2                  JUMP OUT OF TEST
000901                              *
000902                              * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
000903                              *
000904                              *          CRC AND PARITY CHECK ROUTINE
000905                              *
000906                              *
000907  044D  8F00 1145             CRPRT SAVE  <SAV2,=Z'91E0'       SAVE REGS. $B1,B2,$R7
        044F  91E0
000908  0450  ACB0 08CC                    LDB   $B2,<TG2             LOAD ADDRESS OF LCT TABLE INTO $B2
000909  0452  C842 0002                    LDR   $R4,$B2,2            LOAD $R4 WITH CONFIG.
000910  0454  404E                        SOR   $R4,14               SHIFT TO CHAR. SIZE
000911  0455  4E05                        ADV   $R4,=5
000912  0456  CF00 110C                    STR   $R4,<CS              STORE CHARACTER SIZE FROM CONFIG.
000913                              *
000914                              *          SET ZV$C MASK ACCORDING TO CHAR. SIZE
000915                              *
000916  0458  4D08                        CMV   $R4,=8
000917  0459  090A                        BE    >CRPR2
000918  045A  4D07                        CMV   $R4,=7
000919  045B  0984                        BNE   >CHRF1
000920  045C  C870 7F7F                    LDR   $R4,=Z'7F7F'         SET MASK FOR 7 BIT CHAR.
000921  045E  0F83                        B     >CRPR1
000922  045F  C870 3F3F             CHRF1 LDR   $R4,=Z'3F3F'         SET MASK FOR 6 BIT CHAR.
000923  0461  CF00 111A             CRPR1 STR   $R4,<MASK
000924                              *
000925  0463  C842 0006             CRPR2 LDR   $R4,$B2,6            CHECK LCT FLAG TO SEE IF CRC AND/OR PARITY
000926  0465  4D17                        CMV   $R4,=X'17'
000927  0466  0A00 0499                    BAG   <CKCRC               BRANCH TO CHECK CRC ONLY
000928                              *
000929                              *
000930  0468  BC80 08EF                    LDB   $B3,<TG3             LOAD ADDRESS OF DATA BUFFER
000931  046A  9B80 1D00                    LAB   $B1,<DIN1            LOAD ADDRESS OF SHOULD BE WITH PARITY
000932  046C  F800 1120                    LDR   $R7,<RANGE           LOAD RANGE OF TRANSFER IN WORDS
000933  046E  7001                        SOL   $R7,1                CHANGE RANGE TO BYTES
000934  046F  C842 0002                    LDR   $R4,$B2,2            CHECK CONFIGURATION TO SEE IF ODD
000935  0471  C570 1000                    AND   $R4,=X'1000'         OR EVEN PARITY
000936  0473  404C                        SOR   $R4,12               SHIFT TO LSB
000937  0474  8B54                        LBC   =$R4,=X'1'           COMPLEMENT THE  BIT TO WORK IN SUB-ROUTINE
        0475  0001
000938  0476  CF00 111E                    STR   $R4,<PTTY            STORE IN PARITY TYPE LOCATION
000939                              *
000940  0478  C380 09BE                    LNJ   $B4,<CPAR            GO TO CALCULATE PARITY
000941                              *
000942                                    CALL  ZV$C,DIN1,DIN,MASK,RANGE,ERAR
        047A  FBC0 0003
        047C  D380 0000             X
        047E  0F80
        047F  1D00
        0480  1C00
        0481  111A
        0482  1120
        0483  1131
000943  0484  8980 1131                    CMZ   <ERAR                SEE IF ANY ERRORS
000944  0486  090D                        BE    >CKPTY2
000945  0487  D800 1133                    LDR   $R5,<ERAR+2          LOAD '1S' INTO $R5
000946  0489  E800 1132                    LDR   $R6,<ERAR+1          LOAD 'SHOULD BE' INTO $R6
000947  048B  F800 1131                    LDR   $R7,<ERAR            LOAD $R7 WITH WORD COUNT
000948  048D  F380 0BA3                    LNJ   $B7,<ERROR           ERROR;  INCORRECT PARITY
000949                                    CALL  ZV$CO                CONTINUE CHECKING ENTIRE BUFFER
        048F  FBF0 0001
        0491  D380 0000             X
000950  0493  DB80 0959             CKPTY2 LAB  $B5,<JP5             CHANGE $B5  TO POINT TO AFTER COMPARE
000951  0495  89C2 0006                    CMZ   $B2,6                CHECK FLAG IF PARITY ON IO
000952  0497  0A80 04D1                    BALE  <EDCK                BRANCH IF PARITY NOT ON
000953  0499  E380 099C             CKCRC  LNJ   $B6,<BASE           LOAD $R5 WITH LCT BASE ADDRESS
000954  049B  5E03                        ADV   $R5,=3               FOR  BLOCK READ OF CRC RESIDUE
000955  049C  DF00 04A3                    STR   $R5,<RCRRL           STORE RECV. RESIDUE ADDRESS
000956  049E  5E20                        ADV   $R5,=32
000957  049F  DF00 04AC                    STR   $R5,<RCRXL           STORE XMIT RESIDUE ADDRESS
000958                              *
000959  04A1  E380 0B5E                    LNJ   $B6,<RDLCT           READ RECV. CRC RESIDUE
000960  04A3  0000                  RCRRL  DC    0                  RAM ADDRESS
000961  04A4  0002                        DC    2                  RANGE
000962                              *
000963  04A5  C800 112E                    LDR   $R4,<LCTV
000964  04A7  4018                        SCL   $R4,8                ROTATE THE ORDER
```

```
04A8  CF00 1121               STR    $R4,<RCRC
                       *
04AA  E380 0B5E               LNJ    $B6,<RDLCT           READ XMIT CRC RESIDUE
04AC  0000             RCRXL  DC     0                    RAM ADDRESS
04AD  0002                    DC     2                    RANGE
                       *
04AE  C800 112E               LDR    $R4,<LCTV
04B0  4018                    SCL    $R4,8
04B1  CF00 112D               STR    $R4,<XCRC
                       *
04B3  EB80 1185               LAB    $B6,<CRC16           LOAD CRC TYPE
04B5  E800 110C               LDR    $R6,<CS              LOAD CHAR. SIZE
04B7  BC80 08EF               LDB    $B3,<TG3             LOAD ADDRESS OF OUTPUT BUFFER
04B9  C842 0006               LDR    $R4,$B2.6            CHECK FLAG TO SEE IF OUTPUT BUFFER CORRECT
04BB  4D17                    CMV    $R4,=X'17'
04BC  0983                    BNE    >CKPCR
04BD  BB80 1D00               LAB    $B3,<DIN1
04BF  F800 1120        CKPCR  LDR    $R7,<RANGE           LOAD RANGE OF TRANSFER IN WORDS
                       *
04C1  C380 09A0               LNJ    $B4,<CRC             CALCULATE CRC FOR THE TRANSFERS
                       *
04C3  E800 110E               LDR    $R6,<CRCAC           LOAD $R6 WITH SHOULD BE
04C5  D800 1121               LDR    $R5,<RCRC            LOAD CRC RECV. RESIDUE
04C7  E955                    CMR    $R6,=$R5             COMPARE RECEIVE CRC
04C8  0903                    BE     >RCRCG
04C9  F380 0BA3               LNJ    $B7,<ERROR           ERROR; RECEIVE CRC RESIDUE WRONG
04CB  D800 112D        RCRCG  LDR    $R5,<XCRC            LOAD XMIT CRC RESIDUE
04CD  E955                    CMR    $R6,=$R5             COMPARE XMIT CRC
04CE  0903                    BE     >EDCK
04CF  F380 0BA3               LNJ    $B7,<ERROR           ERROR;  XMIT RESIDUE WRONG
04D1  8F80 1145        EDCK   RSTR   <SAV2,=Z'91E0'       RESTORE REGS.
04D3  91E0
04D4  8385                    JMP    $B5                  GO BACK TO DATLOP SUB-ROUTINE
                       *
                       *
                       * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                       *
                       *             CHARACTER SIZE TEST
                       *
04D5  C870 4520        CHSZTD LDR    $R4,=A'E '
04D7  CF00 1364               STR    $R4,<ERMG+1
04D9  B380 098B               LNJ    $B3,<CLFG            CLEAR ALL TEST FLAGS
04DB  8A80 1109               INC    <CHSZFG              SET CHARACTER SIZE TEST FLAG
04DD  4C02                    LDV    $R4,=2               LOAD CONFIGURATION FOR 1 STOP BIT ODD
04DE  CF00 12A0               STR    $R4,<LC4CFR          PARITY, 5 BIT CHAR. RECV.
04E0  4E20                    ADV    $R4,=X'20'
04E1  CF00 12A1               STR    $R4,<LC4CFX          STORE XMIT CONF.
04E3  C870 1F1F               LDR    $R4,=Z'1F1F'         SET MASK FOR FIVE BIT CHAR.
04E5  CF00 111A               STR    $R4,<MASK
04E7  89C0 0C30               CMZ    HISPD                IF SYNC 9600 BAUD, NO 5 BIT MODE
04E9  0980 051A               BNE    <T36
04EB  7C0A             CHSZT1 LDV    $R7,=10              NUMBER OF 30 MS TIME DELAY FOR TRANSFER
04EC  C870 0E18               LDR    $R4,=X'E18'          SET LINE SPEED FOR ASYNC. LINE
04EE  CF00 12A2               STR    $R4,<LCT4S
                       *
                       *             TEST ASYNCHRONOUS LINES WITH BAUD RATE
                       *                  9600 BAUD, WITH 1 STOP BIT
                       *
04F0  9B80 1176        T14    LAB    $B1,<ASYN1           LOAD  LINE TYPE
04F2  B380 0881        T14A   LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
04F4  0F85                    B      >T15
04F5  12DE                    DC     <WCP3                ADDRESS TABLE FOR BLOCK WRITE
04F6  129E                    DC     <LCT4                LCT TABLE
04F7  003F                    DC     (DFLT-DOUT)          RANGE IN WORDS
04F8  119E                    DC     <DOUT
                       *
                       *             TEST CURRENT LOOP ASYNCHRONOUS
                       *
04F9  C871             T15    LDR    $R4,+$B1             LOAD ID TYPE INTO $R4
04FA  C900 1177               CMR    $R4,<ASYNC           CHECK TO SEE IF ALREADY DONE
04FC  0902                    BE     >T15A
04FD  0FF5                    B      >T14A
      04FE             T15A   EQU    $
04FE  F870 E618               LDR    $R7,=Z'E618'         STORE SYNC. CHAR. INTO LCT TABLE
0500  FF00 12A2               STR    $R7,<LCT4S
0502  7C28                    LDV    $R7,=40              NUMBER OF 30 MS TIME DELAYS
                       *
                       *             TEST SYNCHRONOUS LINES (E6 AS SYNC. CHAR.)
                       *
0503  9B80 1178               LAB    $B1,<SYNC            LOAD LINE TYPE
0505  B380 0881        QA3    LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
0507  0F85                    B      >T16
0508  12DE                    DC     <WCP3                ADDRESS TABLE FOR BLOCK WRITE
0509  129E                    DC     <LCT4                LCT TABLE
050A  002F                    DC     (DFLT-DOUT1)         RANGE IN WORDS
050B  11AE                    DC     <DOUT1
                       *
                       *             TEST BI-SYNCHRONOUS LINES (E6 AS SYNC. CHAR.)
                       *
050C  C871             T16    LDR    $R4,+$B1             LOAD ID TYPE INTO $R4
050D  C900 1179               CMR    $R4,<BISYNC          CHECK TO SEE IF ALREADY DONE
050F  0902                    BE     >T17
0510  0FF5                    B      >QA3
                       *
                       *             TEST MIL 188 SYNCHRONOUS (E6 AS SYNC. CHAR.)
                       *
0511  9B80 117A        T17    LAB    $B1,<SYN188          LOAD LINE TYPE
0513  B380 0881               LNJ    $B3,<DTALOP          GO TO LOOP SUB-ROUTINE
0515  0F85                    B      >T36
0516  12DE                    DC     <WCP3                ADDRESS TABLE FOR BLOCK WRITE
0517  129E                    DC     <LCT4                LCT TABLE
0518  002F                    DC     (DFLT-DOUT1)         RANGE IN WORDS
0519  11AE                    DC     <DOUT1
                       *
                       *
051A  8A80 110F        T36    INC    <CRCFG
051C  C800 12A0               LDR    $R4,<LC4CFR          CHANGE CONFIGURATION TO NEXT CHAR. SIZE
051E  CA70 4000               ADD    $R4,=X'4000'
0520  0608                    BCT    >CHSZDE              IF CARRY ALL CHAR. SIZES HAVE BEEN TESTED
0521  CF00 12A0               STR    $R4,<LC4CFR          STORE RECV CONFIG.
0523  4E20                    ADV    $R4,=X'20'
0524  CF00 12A1               STR    $R4,<LC4CFX          STORE XMIT CONFIG.
0526  0F80 04EB               B      <CHSZT1              DO NEXT CHAR.
```

```
                                *
0528  8700 110F                 CHSZDE CL    <CRCFG                CLEAR CRC AND PARITY FLAG
052A  8700 1109                        CL    <CHSZFG               CLEAR CHARACTER SIZE TEST FLAG
052C  8382                             JMP   $B2                   JUMP OUT OF TEST
                                *
                                * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                                *
                                *               STOP BIT TEST
                                *
052D  C870 4620                 STBITE LDR   $R4,=A'F '
052F  CF00 1364                        STR   $R4,<ERMG+1
0531  B380 098B                        LNJ   $B3,<CLFG             CLEAR ALL TEST FLAGS
0533  CB80 0546                        LAB   $B4,<SBIT1            SET UP INTERRUPT VECTORS AND
0535  CF80 1281                        STB   $B4,<ISA4P            POINTERS TO CHANGE RUNNING LEVEL TO 4
0537  CB80 127E                        LAB   $B4,<ISA4+$AF
0539  CF80 0004          X             STB   $B4,<ZHISAZ+4
                                *
053B  C800 1119                        LDR   $R4,<HRTZ             CALCULATE THE LOW AND THE
053D  4F06                             MLV   $R4,=6                HIGH  RANGE OF ACCEPTABLE CLOCK TICKS
053E  4EF7                             ADV   $R4,=-9
053F  CF00 11DF                        STR   $R4,<SECL
0541  4E12                             ADV   $R4,=18
0542  CF00 11E0                        STR   $R4,<SECH
0544  8E70 8004                        LEV   =Z'8004'              CHANGE RUNNING LEVEL TO 4
0546  4C01                      SBIT1  LDV   $R4,=1                SET RTC LEVEL TO 1
0547  CF00 0000          X             STR   $R4,<ZHRTCL
                                *
0549  CB80 05AC                        LAB   $B4,<NBITT            SET RUPT POINTER FOR RTC INTERRUPT
054B  CF80 127B                        STB   $B4,<ISA1P
                                *
054D  CB80 1278                        LAB   $B4,<ISA1+$AF         SET RUPT VECTOR FOR LEVEL 1 RUPT
054F  CF80 0001          X             STB   $B4,<ZHISAZ+1
                                *
0551  8700 129C                        CL    <LC3SYF               CLEAR SYNC. FLAG FOR ASYNC. LINE
0553  8A80 1129                        INC   <SBTFG                SET STOP BIT TIME TEST FLAG
0555  C870 C002                        LDR   $R4,=Z'C002'          LOAD AND STORE INITIAL LINE CONF.
0557  CF00 1298                        STR   $R4,<LC3CFR           FOR RECV.
0559  4E20                             ADV   $R4,=X'20'
055A  CF00 1299                        STR   $R4,<LC3CFX           FOR XMIT
055C  C870 0318                        LDR   $R4,=X'318'           LOAD $R4 WITH SPEED FOR ASYNC. LINE
055E  9B80 1175                        LAB   $B1,<ASYN             LOAD LINE TYPE
0560  EB80 1231                        LAB   $B6,<SBITBL           LOAD ADDRESS OF CHAR. NUMBER TABLE
0562  B380 0318                        LNJ   $B3,<SPDTLP           GO TO SPEED TIME LOOP
                                *
0564  C870 0218                        LDR   $R4,=X'218'           LOAD $R4 WITH SPEED SELECT FOR NEW  ASYC.
0566  9B80 1176                        LAB   $B1,<ASYN1            LOAD ADDRESS OF LINE TYPE
0568  B380 0318                        LNJ   $B3,<SPDTLP           GO TO SPEED TIME LOOP
                                *
056A  C870 0218                        LDR   $R4,=X'218'
056C  9B80 1177                        LAB   $B1,<ASYNC
056E  B380 0318                        LNJ   $B3,<SPDTLP
                                *
                                       CALL  ZV$RD                 RE-INITIALIZE INTERRUPT POINTERS AND VECTORS
0570  FBF0 0001
0572  D380 0000          X
0574  8700 1129                        CL    <SBIFG                CLEAR STOP BIT TEST FLAG
0576  8382                             JMP   $B2                   JUMP OUT OF TEST
                                *
                                * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                                *
                                *          STOP BIT TIME CHECKED HERE
                                *
0577  C940 0C68                 SBTE   CMR   $R4,SECH              SEE IF CLOCK TICK WITHIN
0579  0803                             BAL   >SBTE1                ACCEPTED RANGE
057A  F380 0BA3                        LNJ   $B7,<ERROR            ERROR;  STOP BITS ARE TOO LONG
057C  C900 11DF                 SBTE1  CMR   $R4,<SECL
057E  0A03                             BAG   >SBTE2
057F  F380 0BA3                        LNJ   $B7,<ERROR            ERROR;  STOP BITS ARE TOO SHORT
0581  D800 1298                 SBTE2  LDR   $R5,<LC3CFR           LOAD LCT CONFIG.
0583  8955                             LBT   =$R5,=Z'0800'         SEE IF BIT SET AND SET IT
0584  0800
0585  0509                             BBT   >SBTE3                IF TRUE TRY SLOW SPEED FOR 1.5 STOP BIT
0586  DF00 1298                        STR   $R5,<LC3CFR           STORE LCT CONFIG. WITH 2 STOP BITS
0588  5E20                             ADV   $R5,=X'20'
0589  DF00 1299                        STR   $R5,<LC3CFX           STORE CONFIG. XMIT
058B  8AD2                             INC   =$R2                  INCREMENT INDEX
058C  0F80 0326                        B     <NXSPD1
                                *
058E  2D03                      SBTE3  CMV   $R2,=3                SEE IF ALL STOP BIT S CHECKED
058F  0900 05A3                        BE    <SBTE4
0591  4C02                             LDV   $R4,=2                CHANGE CONFIG. FOR 5 BIT CHAR.
0592  CF00 1298                        STR   $R4,<LC3CFR           STORE RECV  CONFIG.
0594  4E20                             ADV   $R4,=X'20'
0595  CF00 1299                        STR   $R4,<LC3CFX           STORE XMIT CONFIG.
0597  8AD2                             INC   =$R2                  INCREMENT INDEX
0598  C870 0218                        LDR   $R4,=X'218'           LDR $R4 TO SET LINE SPEED TO 75 BAUD
059A  D801                             LDR   $R5,$B1               IF NEW ASYC. RESET $R4 FOR 75 BAUD
059B  D900 1175                        CMR   $R5,<ASYN
059D  0900 0324                        BE    <NXSPD
059F  C870 0118                        LDR   $R4,=X'118'
05A1  0F80 0324                        B     <NXSPD
                                *
05A3  C870 C002                 SBTE4  LDR   $R4,=Z'C002'          LOAD AND STORE BEGINNING CONF.
05A5  CF00 1298                        STR   $R4,<LC3CFR           FOR RECV.
05A7  4E20                             ADV   $R4,=X'20'
05A8  CF00 1299                        STR   $R4,<LC3CFX           FOR XMIT
05AA  0F80 03C4                        B     <TSNL
                                *
                                *
                                * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                                *
                                *               CLOOCK INTERRUPT ROUTINE
                                *
05AC  0005                      NBITT  RTCF                        TURN OFF CLOCK
05AD  F380 0BA3                        LNJ   $B7,<ERROR            ERROR;  STATUS NOT COMPLETE AFTER 5 SEC.
05AF  DB80 0326                        LAB   $B5,<NXSPD1           RETRY THE LINE
05B1  DF80 1281                        STB   $B5,<ISA4P            WITH SAME CONFIG.
05B3  8E70 8004                        LEV   =Z'8004'
05B5  0FF7                             B     >NBITT
                                *
                                *
                                *
                                * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

```
001187                          *
001188                          *          PARITY ERROR TEST
001189                          *
001190  05B6  C870 4720         PARERF LDR   $R4,=A'G '
001191  05B8  CF00 1364                STR   $R4,<ERMG+1
001192  05BA  B380 098B                LNJ   $B3,<CLFG          CLEAR TEST FLAGS
001193  05BC  8A80 111D                INC   <PARFG
001194  05BE  C870 C802                LDR   $R4,=Z'C802'       LOAD CONFIG. ON REC. FOR ODD PARITY
001195  05C0  CF00 1290                STR   $R4,<LC2CFR
001196  05C2  CA70 1020                ADD   $R4,=X'1020'       SET CONFIG. ON XMIT TO EVEN PARITY
001197  05C4  CF00 1291                STR   $R4,<LC2CFX
001198  05C6  C870 F017                LDR   $R4,=Z'F017'       SET CCP FLAG FOR PARITY ONLY
001199  05C8  CF00 1294                STR   $R4,<LCT2FG
001200  05CA  F870 0E18                LDR   $R7,=X'E18'        STORE INITIAL SPEED FOR ASYC. LINES
001201  05CC  FF00 1292                STR   $R7,<LCT2S
001202  05CE  7C06                     LDV   $R7,=6             NUMBER OF 30 MS TIME DELAY FOR TRANSFER
001203                          *
001204                          *          TEST ASYCHRONOUS LINES WITH BAUD RATE
001205                          *            9600 BAUD, WITH 2 STOP BITS
001206  05CF  9B80 1176.        T39    LAB   $B1,<ASYN1          LOAD  LINE TYPE
001207  05D1  B380 0881         T39A   LNJ   $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
001208  05D3  0F85                     B     >T40
001209  05D4  12D9                     DC    <WCP2              ADDRESS TABLE FOR BLOCK WRITE
001210  05D5  128E                     DC    <LCI2              LCT TABLE
001211  05D6  003F                     DC    (DFLT-DOUT)        RANGE IN WORDS
001212  05D7  119E                     DC    <DOUT
001213                          *
001214                          *          TEST CURRENT LOOP ASYNCHRONOUS
001215                          *
001216  05D8  C871              T40    LDR   $R4,+$B1           LOAD ID TYPE INTO $R4
001217  05D9  C900 1177                CMR   $R4,<ASYNC         CHECK TO SEE IF ALREADY DONE
001218  05DB  0902                     BE    >T40A
001219  05DC  0FF5                     B     >T39A
001220  05DD                    T40A   EQU   $
001221  05DD  F870 3218                LDR   $R7,=X'3218'       STORE SYNC. CHAR. INTO LCT TABLE
001222  05DF  FF00 1292                STR   $R7,<LCT2S
001223  05E1  7C28                     LDV   $R7,=40            NUMBER OF 30 MS TIME DELAYS
001224                          *
001225                          *          TEST SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
001226                          *
001227  05E2  9B80 1178                LAB   $B1,<SYNC          LOAD LINE TYPE
001228  05E4  B380 0881         QA4    LNJ   $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
001229  05E6  0F85                     B     >T41
001230  05E7  12D9                     DC    <WCP2              ADDRESS TABLE FOR BLOCK WRITE
001231  05E8  128E                     DC    <LCI2              LCT TABLE
001232  05E9  002F                     DC    (DFLT-DOUT1)       RANGE IN WORDS
001233  05EA  11AE                     DC    <DOUT1
001234                          *
001235                          *          TEST BI-SYNCHRONOUS LINES (32 AS SYNC. CHAR.)
001236                          *
001237  05EB  C871              T41    LDR   $R4,+$B1           LOAD ID TYPE INTO $R4
001238  05EC  C900 1179                CMR   $R4,<BISYNC        CHECK TO SEE IF ALREADY DONE
001239  05EE  0902                     BE    >T42
001240  05EF  0FF5                     B     >QA4
001241                          *
001242                          *          TEST MIL 188 SYNCHRONOUS (32 AS SYNC. CHAR.)
001243                          *
001244  05F0  9B80 117A         T42    LAB   $B1,<SYN188        LOAD LINE TYPE
001245  05F2  B380 0881                LNJ   $B3,<DTALOP         GO TO LOOP SUB-ROUTINE
001246  05F4  0F85                     B     >T45
001247  05F5  12D9                     DC    <WCP2              IORB ADDRESS FOR BLOCK WRITE
001248  05F6  128E                     DC    <LCI2              LCT TABLE
001249  05F7  002F                     DC    (DFLT-DOUT1)                RANGE IN WORDS
001250  05F8  11AE                     DC    <DOUT1
001251                          *
001252  05F9  8700 111D         T45    CL    <PARFG
001253  05FB  8382                     JMP   $B2
001254                          *
001255                          * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
001256                          *
001257                          *           CHECK PARITY ERROR BIT SET
001258                          *
001259  05FC  C380 0B0E         PAR1   LNJ   $B4,<INXT          INPUT  NEXT STATUS
001260  05FE  82D5                     LB    =$R5,=X'1000'
        05FF  1000
001261  0600  0503                     BBT   >PAR2
001262  0601  F380 0BA3                LNJ   $B7,<ERROR         ERROR; STATUS COMPLETE BIT NOT SET
001263  0603  82D5              PAR2   LB    =$R5,=X'40'        SEE IF DATA CHECK ERROR BIT SET
        0604  0040
001264  0605  0503                     BBT   >PAR3
001265  0606  F380 0BA3                LNJ   $B7,<ERROR         ERROR;  DATA CHECK ERROR BIT NOT SET
001266  0608  8AD3              PAR3   INC   =$R3
001267  0609  C380 0B0E                LNJ   $B4,<INXT          INPUT NEXT STATUS
001268  060B  82D5                     LB    =$R5,=X'1000'
        060C  1000
001269  060D  0503                     BBT   >PAR4
001270  060E  F380 0BA3                LNJ   $B7,<ERROR         ERROR;  STATUS NOT COMPLETE ON XMIT SIDE
001271  0610  82D5              PAR4   LB    =$R5,=X'40'
        0611  0040
001272  0612  0583                     BBF   >PAR5
001273  0613  F380 0BA3                LNJ   $B7,<ERROR         ERROR;  DATA CHECK ERROR BIT SET ON XMIT
001274  0615  0F80 0985         PAR5   B     <JP5A
001275                          *
001276                          *
001277                          * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
001278                          *
001279                          *          STOP I/O  TEST
001280                          *
001281  0617  C870 4820         STIOG  LDR   $R4,=A'H '
001282  0619  CF00 1364                STR   $R4,<ERMG+1
001283  061B  B380 098B                LNJ   $B3,<CLFG          CLEAR ALL TEST FLAGS
001284  061D  A800 1181                LDR   $R2,<BIT2          LOAD $R2 WITH FUNCTION COMMAND
001285  061F  8A80 112B                INC   <STPFG
001286  0621  8700 12AD                CL    <LC5SYF            CLEAR SYNC. FLAG FOR ASYNC. LINES
001287  0623  C870 0E18                LDR   $R4,=X'E18'
001288  0625  CF00 12A9                STR   $R4,<LCT5S
001289  0627  7C02                     LDV   $R7,=2             NUMBER OF 30 MS DELAY
001290                          *
001291                          *          TEST ASYNCHRONOUS LINES WITH BAUD RATE
001292                          *            AT 9600 BAUD, WITH 2 STOP BITS
001293                          *
001294  0628  9B80 1176         T19    LAB   $B1,<ASYN1          LINE TYPE
001295  062A  B380 0881         T19A   LNJ   $B3,<DTALOP
```

```
062C  0F85                      B      >T20
062D  12D4                      DC     <WCP1
062E  12A5                      DC     <LCT5
062F  01B6                      DC     438                    RANGE IN WORDS
0630  119E                      DC     <DOUT                  TRANSMIT BUFFER ADDRESS
                         *
                         *            TEST CURRENT LOOP ASYNCHRONOUS
                         *
0631  C871               T20    LDR    $R4,+$B1               LOAD ID TYPE INTO $R4
0632  C900 1177                 CMR    $R4,<ASYNC             CHECK TO SEE IF ALREADY DONE
0634  0902                      BE     >T20A
0635  0FF5                      B      >T19A
      0636               T20A   EQU    $
0636  C870 3A18                 LDR    $R4,=X'3A18'           PUT IN SYNC. CHARACTER TO TEST
0638  CF00 12A9                 STR    $R4,<LCT5S             SYNC. LINE TYPES
063A  C870 021C                 LDR    $R4,=X'21C'            SET LCI SYNC. FLAG
063C  CF00 12AD                 STR    $R4,<LC5SYF
063E  7C05                      LDV    $R7,=5                 NUMBER OF 30 MS DELAY FOR 800 BAUD
                         *
                         *            TEST SYNCHRONOUS (3A AS SYNC. CHAR.)
                         *
063F  9B80 1178                 LAB    $B1,<SYNC              LOAD LINE TYPE
0641  B380 0881          QA5    LNJ    $B3,<DTALOP            LOOP SUB-ROUTINE
0643  0F85                      B      >T21
0644  12D4                      DC     <WCP1
0645  12A5                      DC     <LCT5
0646  01B6                      DC     438
0647  11AE                      DC     <DOUT1
                         *
                         *            TEST BI-SYNCHRONOUS LINES (3A AS SYNC. CHAR.)
                         *
0648  C871               T21    LDR    $R4,+$B1               LOAD ID TYPE INTO $R4
0649  C900 1179                 CMR    $R4,<BISYNC            CHECK TO SEE IF ALREADY DONE
064B  0902                      BE     >T22
064C  0FF5                      B      >QA5
                         *
                         *            TEST MIL 188 SYNCHRONOUS (3A AS SYNC. CHAR.)
                         *
064D  9B80 117A          T22    LAB    $B1,<SYN188            LOAD LINE TYPE
064F  B380 0881                 LNJ    $B3,<DTALOP            LOOP SUB-ROUTINE
0651  0F85                      B      >T34
0652  12D4                      DC     <WCP1
0653  12A5                      DC     <LCT5
0654  01B6                      DC     438
0655  11AE                      DC     <DOUT1
                         *
                         *
0656  8700 112B          T34    CL     <STPFG                 CLEAR STOP I/O FLAG
0658  8382                      JMP    $B2
                         *
                         *
                         * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                         *
                         *           STOP I/O  CHECK ROUTINE
                         *
0659  8AD3               STIQ   INC    =$R3                   CHANGE TO TRASMIT SIDE OF LINE
065A  C380 0B07          STIQ1  LNJ    $B4,<STCR              STOP I/O
                         *
065C  C380 0B0E                 LNJ    $B4,<INXT              INPUT NEXT STATUS
065E  82D5               STIQ5  LB     =$R5,=Z'1000'          CHECK STATUS COMPLETE
065F  1000
0660  0503                      BBT    >TSTF3
0661  F380 0BA3                 LNJ    $B7,<ERROR             ERROR;  STATUS NOT COMPLETE
0663  C380 0B14          TSTF3  LNJ    $B4,<INRG              INPUT RANGE
0665  5983                      BNEZ   $R5,>TSTF4
0666  F380 0BA3                 LNJ    $B7,<ERROR             ERROR;  RANGE IS EQUAL TO ZERO
0668  3B00 0687          TSTF4  BEVN   $R3,<TSTF4A            IF EVEN BOTH SIDE OF THE LINE SHUT-OFF
066A  88D3                      DEC    =$R3                   CHANGE $R3 TO RECEIVE SIDE OF THE LINE
066B  DF00 1126                 STR    $R5,<TP1               STORE XMIT RANGE RESIDUE
066D  8980 112C                 CMZ    <WDBFG                 SEE IF BROADBAND LINE
066F  0914                      BE     >STIQ2                 SHUT RECV. LINE OFF AFTER 70 TRANSFERS
0670  C380 0B0E                 LNJ    $B4,<INXT              INPUT NEXT STATUS
0672  C380 0B14          STIQ3  LNJ    $B4,<INRG              INPUT RANGE
0674  D970 0220                 CMR    $R5,=X'220'            SEE IF 166 TRANSFERS HAVE BEEN DONE
0676  0A7C                      BAG    >STIQ3                 IF DONE
0677  C380 0B07                 LNJ    $B4,<STCR              STOP I/O ON RECV.
                         *
0679  C800 1101                 LDR    $R4,<FIS               INPUT STATUS
067B  C380 0B7E                 LNJ    $B4,<CGSCH
067D  8055                      IO     =$R5,=$R4
067E  0054
067F  0703                      BIOT   >STIQ4
0680  F380 0BA3                 LNJ    $B7,<ERROR
0682  0FDC               STIQ4  B      >STIQ5
                         *
0683  7C03               STIQ2  LDV    $R7,=3
0684  D380 0B4C                 LNJ    $B5,<TDLAY             TIME DELAY BEFORE SHUTTING OFF RECEIVE
0686  0FD4                      B      >STIQ1
                         *
                         *          CHECK RECEIVED DATA
                         *
0687  C800 10FC          TSTF4A LDR    $R4,<FOCC              CCB RESET FOR REC
0689  C380 0B7E                 LNJ    $B4,<CGSCH
068B  8000 1184                 IO     <BIT7,=$R4
068D  0054
068E  0703                      BIOT   >QT2
                         *
068F  F380 0BA3                 LNJ    $B7,<ERROR             ERROR;  I/O WAS NAK'ED
                         *
0691  8AD3               QT2    INC    =$R3
0692  C800 10FC                 LDR    $R4,<FOCC              CCB RESET FOR TRANSMIT
0694  C380 0B7E                 LNJ    $B4,<CGSCH
0696  8000 1184                 IO     <BIT7,=$R4
0698  0054
0699  0703                      BIOT   >QT1
                         *
069A  F380 0BA3                 LNJ    $B7,<ERROR             ERROR;  I/O WAS NAK'ED
                         *
069C  DF00 1127          QT1    STR    $R5,<TP2               STORE RECEIVE RANGE RESIDUE
069E  E380 099C                 LNJ    $B6,<BASE              LOAD $R5 WITH LCT BASE
06A0  5E10                      ADV    $R5,=16
06A1  DF00 06A8                 STR    $R5,<TSTF2B
06A3  5E20                      ADV    $R5,=X'20'
```

```
001405  06A4  DF00 06B1              STR   $R5,<TSTF8B
001406                         *
001407  06A6  E380 0B5E              LNJ   $B6,<RDLCT          READ LCT
001408  06A8  0000              TSTF2B DC    0                 ADDRESS
001409  06A9  0002                     DC    2
001410                         *
001411  06AA  8980 112E              CMZ   <LCTV
001412  06AC  0903                   BE    >TSTF7K
001413  06AD  F380 0BA3              LNJ   $B7,<ERROR          ERROR:  RECV. LCT STATUS NOT CLEAR
001414                         *
001415  06AF  E380 0B5E              TSTF7K LNJ  $B6,<RDLCT    READ LCT
001416  06B1  0000                   TSTF8B DC   0
001417  06B2  0002                          DC   2
001418                         *
001419  06B3  8980 112E              CMZ   <LCTV               CHECK LCT STATUS IS CLEARED
001420  06B5  0903                   BE    >TSTF2
001421  06B6  F380 0BA3              LNJ   $B7,<ERROR          ERROR:  LCT STATUS IS NOT CLEAR
001422  06B8  9800 1120              TSTF2 LDR   $R1,<RANGE    CALCULATE RANGE OF GOOD DATA TO BE CHECK
001423  06BA  9F00 1115              STR   $R1,<FBRG
001424  06BC  1001                   SOL   $R1,1               CHANGE TO BYTE RANGE
001425  06BD  9200 1126              SUB   $R1,<TP1
001426  06BF  1041                   SOR   $R1,1               CHANGE TO WORDS
001427  06C0  88D1                   DEC   =$R1
001428  06C1  9F00 1120              STR   $R1,<RANGE
001429                         *
001430  06C3  8980 12AD              CMZ   <LC5SYF             CHECK FOR SYNC LINE
001431  06C5  0917                   BE    >TSTF4C
001432  06C6  1E04                   ADV   $R1,=4
001433  06C7  C800 1126              LDR   $R4,<TP1            SYNC LINE CHECK FOR LINE MARKING
001434  06C9  C200 1127              SUB   $R4,<TP2
001435  06CB  4041                   SOR   $R4,1
001436  06CC  CA51                   ADD   $R4,=$R1
001437  06CD  4EF9                   ADV   $R4,-7
001438  06CE  CF00 1127              STR   $R4,<TP2
001439  06D0  C870 FFFF              LDR   $R4,=Z'FFFF'        LINE MARKING DATA
001440  06D2  C910 1C00              TSTF5A CMR  $R4,<DIN.$R1
001441  06D4  0903                   BE    >TSTF5
001442  06D5  F380 0BA3              LNJ   $B7,<ERROR          ERROR:  SYNC. LINE NOT MARKING WITH XMIT OFF
001443  06D7  8AD1                   TSTF5 INC   =$R1
001444  06D8  9900 1127              CMR   $R1,<TP2            SEE IF ALL MARKING DATA CHECK
001445  06DA  0882                   BAGE  >TSTF4C
001446  06DB  0FF7                   B     >TSTF5A
001447                         *
001448  06DC  C800 11DD              TSTF4C LDR  $R4,<DFLT     CHECK REMAINING DATA IS DEFAULT (5555)
001449  06DE  1E07                   ADV   $R1,=7
001450  06DF  C910 1C00              TSTF4D CMR  $R4,<DIN.$R1
001451  06E1  0903                   BE    >TSTF4E
001452  06E2  F380 0BA3              LNJ   $B7,<ERROR          ERROR;RECEIVE DATA IN NO DATA AREA
001453  06E4  8AD1                   TSTF4E INC  =$R1
001454  06E5  9900 1115              CMR   $R1,<FBRG
001455  06E7  0878                   BAL   >TSTF4D
001456  06E8  0F80 0940              B     <JP4                CHECK FINAL GOOD DATA IN DTALOP
001457                         *
001458                         *
001459                         * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
001460                         *
001461                         *
001462                         *          UNDERRUN AND OVERRUN TEST
001463                         *
001464  06EA  C870 4A20              UORNTH LDR  $R4,=A'J  '
001465  06EC  CF00 1364                     STR  $R4,<ERMG+1
001466  06EE  B380 098B              LNJ   $B3,<CLFG           CLEAR ALL TEST FLAGS
001467  06F0  8700 12CA              CL    <LCT8FG
001468  06F2  8A80 112C              INC   <WDBFG
001469  06F4  8A80 1124              INC   <RUNFG              SET UNDERRUN AND OVERRUN TEST FLAG
001470  06F6  C870 021D              LDR   $R4,=X'21D'
001471  06F8  CF00 12B6              STR   $R4,<LCT6TF
001472  06FA  7C06                   LDV   $R7,=6              NUMBER OF 30 MS TIME DELAY FOR TRANSFER
001473  06FB  C870 0E18              LDR   $R4,=X'E18'         SET LINE SPEED FOR ASYC. LINES
001474  06FD  CF00 12B4              STR   $R4,<LCT6S
001475                         *
001476                         *          TEST ASYCHRONOUS LINES WITH BAUD RATE
001477                         *              9600 BAUD, WITH 1 STOP BIT
001478                         *
001479  06FF  9B80 1176              T24   LAB   $B1,<ASYN1    LOAD  LINE TYPE
001480  0701  B380 0881              T24Z  LNJ   $B3,<DTALOP   GO TO LOOP SUB-ROUTINE
001481  0703  0F85                   B     >T25
001482  0704  12E3                   DC    <WCP4               ADDRESS TABLE FOR BLOCK WRITE
001483  0705  12AF                   DC    <LCT6               LCT TABLE
001484  0706  003F                   DC    (DFLT-DOUT)         RANGE IN WORDS
001485  0707  119E                   DC    <DOUT
001486                         *
001487                         *          TEST CURRENT LOOP ASYNCHRONOUS
001488                         *
001489  0708  C871                   T25   LDR   $R4,+$B1      LOAD ID TYPE INTO $R4
001490  0709  C900 1177              CMR   $R4,<ASYNC          CHECK TO SEE IF ALREADY DONE
001491  070B  0902                   BE    >T25Z
001492  070C  0FF5                   B     >T24Z
001493  070D                         T25Z  EQU   $
001494  070D  8700 112C              CL    <WDBFG
001495  070F  F870 401D              LDR   $R7,=Z'401D'
001496  0711  FF00 12B6              STR   $R7,<LCT6TF
001497  0713  F870 1618              LDR   $R7,=Z'1618'        STORE SYNC. CHAR. INTO LCT TABLE
001498  0715  FF00 12B4              STR   $R7,<LCT6S
001499  0717  7C40                   LDV   $R7,=X'40'          NUMBER OF 30 MS TIME DELAYS
001500                         *
001501                         *          TEST SYNCHRONOUS LINES (16 AS SYNC. CHAR.)
001502                         *
001503  0718  9B80 1178              LAB   $B1,<SYNC           LOAD LINE TYPE
001504  071A  B380 0881              QA6   LNJ   $B3,<DTALOP   GO TO LOOP SUB-ROUTINE
001505  071C  0F85                   B     >T26
001506  071D  12E3                   DC    <WCP4               ADDRESS TABLE FOR BLOCK WRITE
001507  071E  12AF                   DC    <LCT6               LCT TABLE
001508  071F  002F                   DC    (DFLT-DOUT1)        RANGE IN WORDS
001509  0720  11AE                   DC    <DOUT1
001510                         *
001511                         *          TEST BI-SYNCHRONOUS LINES (16 AS SYNC. CHAR.)
001512                         *
001513  0721  C871                   T26   LDR   $R4,+$B1      LOAD ID TYPE INTO $R4
001514  0722  C900 1179              CMR   $R4,<BISYNC         CHECK TO SEE IF ALREADY DONE
001515  0724  0902                   BE    >T27
001516  0725  0FF5                   B     >QA6
001517                         *
```

```
                          *          TEST MIL 188 SYNCHRONOUS (16 AS SYNC. CHAR.)
                          *
0726  9B80 117A           T27     LAB    $B1,<SYN188           LOAD LINE TYPE
0728  B380 0881                   LNJ    $B3,<DTALOP           GO TO LOOP SUB-ROUTINE
072A  0F85                        B      >T32
072B  12E3                        DC     <WCP4                 ADDRESS TABLE FOR BLOCK WRITE
072C  12AF                        DC     <LCT6                 LCT TABLE
072D  002F                        DC     (DFLT-DOUT1)          RANGE IN WORDS
072E  11AE                        DC     <DOUT1
                          *
                          *
072F  8700 112C           T32     CL     <WDBFG
0731  8700 1124                   CL     <RUNFG
0733  8700 12CA                   CL     <LCT8FG
0735  8382                        JMP    $B2
                          *
                          *
                          * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
                          *
                          *       UNDERRUN AND OVERRUN CHECK ROUTINE
                          *
0736  88D3                RUNCK   DEC    =$R3
0737  E380 099C                   LNJ    $B6,<BASE             LOAD $R5 WITH LCT BASE ADDRESS
0739  5E3C                        ADV    $R5,=X'3C'
073A  DF00 0761                   STR    $R5,<RNCK1
                          *
073C  8980 112C                   CMZ    <WDBFG
073E  080F                        BAL    >RNCK4
073F  C800 1101                   LDR    $R4,<FIS
0741  C380 0B7E                   LNJ    $B4,<CGSCH            INPUT STATUS TO CHECK DATA
0743  8055                        IO     =$R5,=$R4             SERVICE ERROR (BIT2) SET
0744  0054
0745  0703                        BIOT   >RNCK4A
0746  F380 0BA3                   LNJ    $B7,<ERROR            ERROR;  I/O NAK'ED
0748  82D5                RNCK4A  LB     =$R5,=Z'2000'
0749  2000
074A  0503                        BBT    >RNCK4
074B  F380 0BA3                   LNJ    $B7,<ERROR            ERROR;  DATA SERVICE ERROR BIT NOT SET ON OVR
074D  8AD3                RNCK4   INC    =$R3                  CHECK XMIT DATA SERVICE ERROR
074E  8980 112C                   CMZ    <WDBFG
0750  0A0F                        BAG    >RNCK5
0751  C800 1101                   LDR    $R4,<FIS
0753  C380 0B7E                   LNJ    $B4,<CGSCH
0755  8055                        IO     =$R5,=$R4
0756  0054
0757  0703                        BIOT   >RNCK4B
0758  F380 0BA3                   LNJ    $B7,<ERROR            ERROR;  I/O NAK'ED
                          *
075A  82D5                RNCK4B  LB     =$R5,=Z'2000'
075B  2000
075C  0503                        BBT    >RNCK5
075D  F380 0BA3                   LNJ    $B7,<ERROR            ERROR;  DATA SERVICE ERROR BIT NOT SET ON UDR
075F  E380 0B5E           RNCK5   LNJ    $B6,<RDLCT            READ LCT
0761  0000                RNCK1   DC     0
0762  0002                        DC     2
                          *
0763  8980 112C                   CMZ    <WDBFG                CHECK FLAG
0765  080A                        BAL    >RNCK9
0766  8280 112E                   LB     <LCTV,=Z'0200'
0768  0200
0769  050C                        BBT    >RNCK6
076A  F380 0BA3                   LNJ    $B7,<ERROR            ERROR;  OVERRUN FAIL TO SET BIT IN LR5
076C  8980 112C                   CMZ    <WDBFG                CHECK FLAG TO TEST UNDERRUN BIT
076E  0987                        BNE    >RNCK6
                          *
076F  8280 112E           RNCK9   LB     <LCTV,=Z'0100'
0771  0100
0772  0503                        BBT    >RNCK6
0773  F380 0BA3                   LNJ    $B7,<ERROR            ERROR;  UNDERRUN FAIL TO SET BIT IN LR5
0775  0F80 0985           RNCK6   B      <JP5A
                          *
                          *
                          * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
                          *
                          *         LOOP DSC INTO DSS WITH CABLE LOOP
                          *
                          *
                          *          THE SIGNALS ARE TIED IN ACCORDING TO THE FOLLOWING CHART;
                          *
                          * ----------------------------------------------------------------------
                          *  $             $             $              $             $             $
                          *  $     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    $    ----     $    ----     $
                          *  $-------------$-------------$--------------$-------------$-------------$
                          *
                          *
0777  C870 4B20           LPCBLK  LDR    $R4,=A'K '
0779  CF00 1364                   STR    $R4,<ERMG+1
077B  B380 098B                   LNJ    $B3,<CLFG             CLEAR ALL TEST FLAGS
077D  8700 111A                   CL     <MASK
077F  9B80 1235                   LAB    $B1,<TLTBLA           PUT ADDRESS OF SHOULD BE TABLE IN ZV$C
0781  9F80 07F6                   STB    $B1,<LOPD7+4+$AF
0783  9B80 125A                   LAB    $B1,<C4               PUT ADDRESS OF RANGE IN ZV$C
0785  9F80 07F9                   STB    $B1,<LOPD7+4+4*$AF
0787  C870 FC1D                   LDR    $R4,=Z'FC1D'
0789  CF00 12BD                   STR    $R4,<LC75M            SET MASK FO ASYC. LINE
078B  4C08                        LDV    $R4,=8                SET RANGE FOR CHECK ROUTINE
```

```
001625  078C  CF00 1127              STR   $R4,<IP2
001626  078E  7C02                   LDV   $R7,=2              LOAD $R7 WITH NUMBER OF 30MS DELAY
001627                          *
001628  078F  9B80 1175              LAB   $B1,<ASYN           LOAD $B1 WITH ADDRESS OF LINE TYPE
001629  0791  B380 07B4              LNJ   $B3,<LOPDS          BRANCH TO LOOP DSS INTO DSC AND CHECK
001630                          *
001631  0793  9B80 1176              LAB   $B1,<ASYN1          LOAD ADDRESS OF LINE TYPE
001632  0795  B380 07B4              LNJ   $B3,<LOPDS
001633                          *
001634  0797  9B80 1239              LAB   $B1,<TLTBLC         STORE SHOULD BE ADDRESS FOR ZV$C
001635  0799  9F80 07F6              STB   $B1,<LOPD7+4+$AF
001636  079B  9B80 1177              LAB   $B1,<ASYNC          LOAD ADDRESS OF LINE TYPE
001637  079D  B380 07B4              LNJ   $B3,<LOPDS
001638                          *
001639  079F  9B80 123D              LAB   $B1,<TLTBLS
001640  07A1  9F80 07F6              STB   $B1,<LOPD7+4+$AF
001641  07A3  C870 F81D              LDR   $R4,=Z'F81D'        CHANGE MASK FOR SYNC. LINES
001642  07A5  CF00 12BD              STR   $R4,<LC7SM
001643  07A7  9B80 1179              LAB   $B1,<BISYNC
001644  07A9  B380 07B4              LNJ   $B3,<LOPDS
001645                          *
001646  07AB  9B80 1178              LAB   $B1,<SYNC
001647  07AD  B380 07B4              LNJ   $B3,<LOPDS
001648                          *
001649  07AF  9B80 117A              LAB   $B1,<SYN188
001650  07B1  B380 07B4              LNJ   $B3,<LOPDS
001651  07B3  8382                   JMP   $B2
001652                          *
001653                          * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
001654                          *
001655                          *          LOOP AND TEST DSC INTO DSS LOOP
001656                          *
001657                          *
001658  07B4  9F80 07B9          LOPDS STB   $B1,<LOPD1          STORE ADDRESS OF LINE TYPE
001659  07B6  8753                   CL    =$R3
001660  07B7  D380 0B38          LOPD  LNJ   $B5,<FLN            FIND LINE ON
001661  07B9  0000               LOPD1 RESV  $AF,0               LINE TYPE ADDRESS
001662  07BA  8383                   JMP   $B3
001663  07BB  C380 0AF3              LNJ   $B4,<GENITZ         GENERAL INITIALIZE
001664                          *
001665                          *    WRITE  CHANNEL PROGRAM AND TLU TABLE INTO RAM
001666                          *
001667  07BD  C570 FF80              AND   $R4,=Z'FF80'
001668  07BF  C470 0040              OR    $R4,=X'40'
001669  07C1  CF00 110A              STR   $R4,<CHNZ
001670                               CALL  ZV$MLW,CHLP5,WCP5,HX200,CHNZ
        07C3  FBC0 0003
        07C5  D380 0000      X
        07C7  0F80
        07C8  0E49
        07C9  12E8
        07CA  125D
        07CB  110A
001671                               CALL  ZV$MLW,TLUTB2,WCP5B,HX400,CHNZ
        07CC  FBC0 0003
        07CE  D380 0000      X
        07D0  0F80
        07D1  1246
        07D2  12E9
        07D3  125E
        07D4  110A
001672                          *
001673  07D5  D380 0B24              LNJ   $B5,<SETLCT         SEND OUT LCT LOCATIONS
001674  07D7  12B9                   DC    <LCT7
001675                          *
001676  07D8  C800 1127              LDR   $R4,<IP2            LOAD $R4 WITH RANGE
001677  07DA  D380 0B82              LNJ   $B5,<MCCB           FORM CCB TO TRANSFER DSS INTO MEMORY
001678  07DC  1C00                   DC    <DIN
001679  07DD  0060                   DC    X'60'
001680                          *
001681  07DE  C800 10FC              LDR   $R4,<FOCC           FORM I/O CONTROL WORD
001682  07E0  C380 0B7E              LNJ   $B4,<CGSCH          FOR START I/O
001683  07E2  8000 1180              IO    <BITI,=$R4
        07E4  0054
001684  07E5  0703                   BIOT  >LOPD2
001685  07E6  F380 0BA3              LNJ   $B7,<ERROR          ERROR:  I/O WAS NAK'ED
001686  07E8  D380 0B4C          LOPD2 LNJ   $B5,<TDLAY          TIME DELAY
001687                          *
001688  07EA  C380 0B0E              LNJ   $B4,<INXT           INPUT NEXT STATUS
001689  07EC  82D5                   LB    =$R5,=Z'1000'       CHECK STATUS COMPLETE
        07ED  1000
001690  07EE  0503                   BBT   >LOPD7
001691  07EF  F380 0BA3              LNJ   $B7,<ERROR          ERROR:  STATUS NOT COMPLETE AFTER GNB
001692                           LOPD7 CALL  ZV$C,$,DIN,MASK,$,ERAR
        07F1  FBC0 0003
        07F3  D380 0000      X
        07F5  0F80
        07F6  07F1
        07F7  1C00
        07F8  111A
        07F9  07F1
        07FA  1131
001693  07FB  8980 1131              CMZ   <ERAR               CHECK FOR ERROR
001694  07FD  090D                   BE    >LOPD3
001695  07FE  D800 1133              LDR   $R5,<ERAR+2         PUT '15' INTO $R5
001696  0800  E800 1132              LDR   $R6,<ERAR+1         PUT 'SHOULD BE' INTO $R6
001697  0802  F800 1131              LDR   $R7,<ERAR           PUT WORD COUNT INTO $R7
001698  0804  F380 0BA3              LNJ   $B7,<ERROR          ERROR:  DSC DID NOT LOOP INTO DSS CORRECTLY
001699                               CALL  ZV$CO               CONTINUE CHECKING FOR ERROR
        0806  FBF0 0001
        0808  D380 0000      X
001700  080A  3E02               LOPD3 ADV   $R3,=2              CHANGE TO NEXT LINE
001701  080B  0F80 07B7              B     <LOPD
001702                          *
001703                          *
001704                          * - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
001705                          *
001706                          *          TRANSMIT MARK AND SPACE TEST
001707                          *
001708  080D  C870 4C20          XMMSL LDR   $R4,=A'L '
001709  080F  CF00 1364              STR   $R4,<ERMG+1
001710  0811  B380 098B              LNJ   $B3,<CLFG
001711  0813  C870 FF07              LDR   $R4,=Z'FF07'        ALTER DATA
```

```
001712  0815  CF00 11A1                STR   $R4,<DOUT+3
001713  0817  8700 111A                CL    <MASK
001714  0819  C870 081A                LDR   $R4,=X'81A'          LOAD LCT WITH MARK BIT
001715  081B  8A80 111D                INC   <PARFG
001716  081D  8A80 1128                INC   <TRMRK
001717  081F  EB80 1177          U2    LAB   $B6,<ASYNC           LOAD $B6 WITH ADDRESS OF TABLE END
001718  0821  CF00 12D0                STR   $R4,<LCT9MS
001719  0823  7C06                     LDV   $R7,=6               NUMBER OF 60 MS DELAY
001720                             *
001721                             *          TEST ALL ASYNCHRONOUS LINES
001722                             *
001723  0824  2C01                     LDV   $R2,=1               LOAD INDEX
001724  0825  9B80 1175                LAB   $B1,<ASYN
001725  0827  B380 0881          U3    LNJ   $B3,<DTALOP          LOOP SUB-ROUTINE
001726  0829  0F85                     B     >U1
001727  082A  12FB                     DC    <WCP9
001728  082B  12CC                     DC    <LCT9
001729  082C  0005                     DC    5
001730  082D  119E                     DC    <DOUT
001731                             *
001732        082E               U1    EQU   $
001733  082E  9DD6                     CMB   $B1,=$B6             SEE IF ALL ASYNCHRONOUS  LINES TESTED
001734  082F  0903                     BE    >U5
001735  0830  9BA1                     LAB   $B1,$B1,$R2
001736  0831  0FF6                     B     >U3
001737                             *
001738  0832  C870 101A          U5    LDR   $R4,=X'101A'         LOAD LCT FOR SPACE CONDITION
001739  0834  C900 12D0                CMR   $R4,<LCT9MS          CHECK IF ALREADY DONE
001740  0836  09E9                     BNE   >U2
001741  0837  8700 111D                CL    <PAKFG
001742  0839  8700 1128                CL    <TRMRK
001743  083B  C870 0607                LDR   $R4,=X'607'          PUT DATA BACK TO NORMAL
001744  083D  CF00 11A1                STR   $R4,<DOUT+3
001745  083F  8382                     JMP   $B2
001746                             *
001747                             ****************
001748                             *
001749                             *          DATA CHECK FOR XMIT MARK AND SPACE
001750                             *
001751        0840               XMSCK EQU   $
001752  0840  8F00 1145                SAVE  <SAV2,=Z'0002'       SAVE $B6
      0842  0002
001753  0843  DB80 1251                LAB   $B5,<XMSB            LOAD ADDRESS FOR ZV$C
001754  0845  C800 12D0                LDR   $R4,<LCT9MS          CHECK LCT TO SEE IF MARK
001755  0847  C970 081A                CMR   $R4,=X'81A'          B IF MARK TEST
001756  0849  0903                     BE    >XMSCK1
001757  084A  DB80 1255                LAB   $B5,<XSSB            LOAD ADDRESS OF SPACE SHOULD BE
001758  084C  DF80 0853          XMSCK1 STB   $B5,<XMSCK2+4+$AF    STORE SHOULD BE ADDRESS FOR ZV$C
001759                             XMSCK2 CALL  ZV$C,$,DIN,MASK,C4,ERAR
      084E  FBC0 0003
      0850  D380 0000      X
      0852  0F80
      0853  084E
      0854  1C00
      0855  111A
      0856  125A
      0857  1131
001760  0858  8980 1131                CMZ   <ERAR                CHECK FOR ERROR
001761  085A  090D                     BE    >XMSCK3
001762  085B  D800 1133                LDR   $R5,<ERAR+2          PUT 'IS' INTO $R5
001763  085D  E800 1132                LDR   $R6,<ERAR+1          PUT 'SHOULD BE' INTO $R6
001764  085F  F800 1131                LDR   $R7,<ERAR            PUT WORD COUNT INTO $R7
001765  0861  F380 0BA3                LNJ   $B7,<ERROR           ERROR;  TRANSMIT MARK OR SPACE ERROR
001766                                 CALL  ZV$CO                CONTINUE CHECKING FOR ERROR
      0863  FBF0 0001
      0865  D380 0000      X
001767  0867  C870 101A          XMSCK3 LDR   $R4,=X'101A'
001768  0869  C900 12D0                CMR   $R4,<LCT9MS
001769  086B  0991                     BNE   >XMSCK5
001770  086C  E380 099C                LNJ   $B6,<BASE            LOAD $R5 WITH LCT BASE
001771  086E  5E3C                     ADV   $R5,=60
001772  086F  DF00 0873                STR   $R5,<XMSCK4          STORE ADDRESS TO READ LCT
001773                             *
001774  0871  E380 0B5E                LNJ   $B6,<RDLCT
001775  0873  0000               XMSCK4 DC    0                  ADDRESS
001776  0874  0002                     DC    2
001777                             *
001778  0875  D870 FE00                LDR   $R5,=Z'FE00'         LOAD $R5 WITH SHOULD BE
001779  0877  D900 112E                CMR   $R5,<LCTV
001780  0879  0903                     BE    >XMSCK5
001781  087A  F380 0BA3                LNJ   $B7,<ERROR           ERROR;  FRAMING ERROR FAILURE
001782  087C  8F80 1145          XMSCK5 RSTR  <SAV2,=Z'0002'       RESTORE $B6
      087E  0002
001783  087F  0F80 0985                B     <JP5A
001784                             *
001785                             *
001786                             *=================================================================
001787  0881  8F00 1155          DTALOP SAVE  <SAV3,=Z'0112'       SAVE $B3,$B6
      0883  0112
001788  0884  9F80 0889                STB   $B1,<TG1             FOR FIND LINE ON
001789  0886  8753                     CL    =$R3                 CLEAR $R3
001790  0887  D380 0B38          NXLT  LNJ   $B5,<FLN             FIND LINE ON
001791  0889  0000               TG1   RESV  $AF,0
001792  088A  8383                     JMP   $B3                  NO LINE ON RETURN
001793                             *
001794                                 CALL  ZV$F,DIN,DFLT,HX200  FILL INPUT BUFFER
      088B  FBC0 0003
      088D  D380 0000      X
      088F  0F80
      0890  1C00
      0891  11DD
      0892  125D
001795  0893  C808 0889                LDR   $R4,*<TG1
001796  0895  C900 1177                CMR   $R4,<ASYNC
001797  0897  0983                     BNE   >DTAL                B IF NOT CURRENT LOOP ADAPTER
001798  0898  C380 09FD                LNJ   $B4,<SCLS            SET  CURRENT LOOP ADAPTER SPEED
001799  089A  BBC3 0001          DTAL  LAB   $B3,$B3,1            DUMMY INCREMENT OF $B3
001800  089C  C380 0AF3                LNJ   $B4,<GENITZ          GENERAL INITIALIZE
001801  089E  DCF3                     LDB   $B5,+$B3             GET ADDRESS TO STORE FOR ZV$MLW CHANNEL
001802  089F  C570 FF80                AND   $R4,=Z'FF80'         SET CHANNEL FOR ZV$MLW
001803  08A1  C470 0040                OR    $R4,=X'40'           OR IN DIRECTION BIT
001804  08A3  CF00 110A                STR   $R4,<CHNZ            STORE CHANNEL FOR ZV$LIB CALL
001805  08A5  CCF5                     LDB   $B4,+$B5             LOAD AND STORE ADDRESS FOR
```

```
001806  08A6  CF80 08BA                    STB   $B4,<LPWRT+4+$AF     ZV$LIB CALL
001807  08A8  CCF5                         LDB   $B4,+$B5
001808  08A9  CF80 08C3                    STB   $B4,<LPWRT1+4+$AF
001809  08AB  DF80 08BB                    STB   $B5,<LPWRT+4+2*$AF
001810  08AD  DBC5 0001                    LAB   $B5,$B5,1            INCREMENT $B5
001811  08AF  DF80 08C4                    STB   $B5,<LPWRT1+4+2*$AF
001812  08B1  DBC5 0001                    LAB   $B5,$B5,1
001813  08B3  DF80 08C5                    STB   $B5,<LPWRT1+4+3*$AF
001814                              *
001815                        LPWRT CALL    ZV$MLW,$,$,HX200,CHNZ
        08B5  FBC0 0003
        08B7  D380 0000       X
        08B9  0F80
        08BA  08B5
        08BB  08B5
        08BC  125D
        08BD  110A
001816                        LPWRT1 CALL   ZV$MLW,$,$,$,CHNZ
        08BE  FBC0 0003
        08C0  D380 0000       X
        08C2  0F80
        08C3  08BE
        08C4  08BE
        08C5  08BE
        08C6  110A
001817                              *
001818  08C7  DCF3                         LDB   $B5,+$B3             STORE ADDRESS FOR LCT TABLE
001819  08C8  DF80 08CC                    STB   $B5,<TG2
001820                              *
001821  08CA  D380 0B24                    LNJ   $B5,<SETLCT          SEND OUT LCT
001822  08CC  0000                   TG2   RESV  $AF,0
001823                              *
001824  08CD  C800 10FC                    LDR   $R4,<FOCC            CCB RESET FOR REC
001825  08CF  C380 0B7E                    LNJ   $B4,<CGSCH
001826  08D1  8000 1184                    IO    <BIT7,=$R4
        08D3  0054
001827  08D4  0703                         BIOT  >DT2
001828                              *
001829  08D5  F380 0BA3                    LNJ   $B7,<ERROR           ERROR:  I/O WAS NAK'ED
001830                              *
001831  08D7  8AD3                   DT2   INC   =$R3
001832  08D8  C800 10FC                    LDR   $R4,<FOCC            CCB RESET FOR TRANSMIT
001833  08DA  C380 0B7E                    LNJ   $B4,<CGSCH
001834  08DC  8000 1184                    IO    <BIT7,=$R4
        08DE  0054
001835  08DF  0703                         BIOT  >DT3
001836                              *
001837  08E0  F380 0BA3                    LNJ   $B7,<ERROR           ERROR:  I/O WAS NAK'ED
001838                              *
001839  08E2  C803                   DT3   LDR   $R4,$B3              LOAD RANGE
001840  08E3  CF00 1120                    STR   $R4,<RANGE           STORE RANGE FOR ZV$C
001841  08E5  4001                         SOL   $R4,1                CONVERT TO BYTES FOR CCB RANGE
001842                              *
001843  08E6  BBC3 0001                    LAB   $B3,$B3,1
001844  08E8  DCF3                         LDB   $B5,+$B3             STORE ADDRESS FOR TRANSMIT BUFFER
001845  08E9  DF80 08EF                    STB   $B5,<TG3             FOR CCB FORMATION
001846  08EB  DF80 0945                    STB   $B5,<JP4+4+$AF       FOR DATA CHECK CALL(ZV$C)
001847                              *
001848  08ED  D380 0B82                    LNJ   $B5,<MCCB            FORM CCB FOR TRANSMIT
001849  08EF  0000                   TG3   RESV  $AF,0
001850  08F0  0060                         DC    X'60'
001851                              *
001852  08F1  88D3                         DEC   =$R3                 GET RECV. CHANNEL
001853  08F2  D380 0B82                    LNJ   $B5,<MCCB            FORM CCB FOR RECIEVE SIDE
001854  08F4  1C00                         DC    <DIN
001855  08F5  0060                         DC    X'60'
001856                              *
001857  08F6  5C01                         LDV   $R5,=1               START I/O ON XMIT SIDE TO SET
001858  08F7  8AD3                         INC   =$R3                 CONFIG. AND XMIT FILL CHAR. (SYNC. ONLY)
001859  08F8  FE55                         SWR   $R7,=$R5
001860  08F9  C800 10FC                    LDR   $R4,<FOCC
001861  08FB  C380 0B7E                    LNJ   $B4,<CGSCH
001862  08FD  8980 1128                    CMZ   <TRMRK
001863  08FF  0980 0907                    BNE   <MELCO
001864  0901  8000 1180                    IO    <BIT1,=$R4
        0903  0054
001865  0904  0703                         BIOT  >MELCO
001866  0905  F380 0BA3                    LNJ   $B7,<ERROR
001867  0907  D380 0B4C              MELCO LNJ   $B5,<TDLAY
001868  0909  FE55                         SWR   $R7,=$R5
001869  090A  88D3                         DEC   =$R3
001870                              *
001871  090B  C800 10FC                    LDR   $R4,<FOCC
001872  090D  C380 0B7E                    LNJ   $B4,<CGSCH           FORM I/O FOR START I/O
001873  090F  8000 1180                    IO    <BIT1,=$R4
        0911  0054
001874  0912  0703                         BIOT  >JP2
001875  0913  F380 0BA3                    LNJ   $B7,<ERROR
001876  0915  D380 0B4C              JP2   LNJ   $B5,<TDLAY           TIME DELAY
001877  0917  8980 112B                    CMZ   <STPFG               CHECK STOP I/O TEST FLAG
001878  0919  0903                         BE    >JP2A
001879  091A  0F80 0659                    B     <STIO                BRANCH TO STOP I/O AND CHANNEL RESET TEST
001880  091C  8980 111D              JP2A  CMZ   <PARFG
001881  091E  0908                         BE    >JP2B
001882  091F  8980 110B                    CMZ   <COUNT
001883  0921  0900 05FC                    BE    <PAR1                BRANCH TO CHECK THAT PARITY ERROR SET
001884  0923  8AD3                         INC   =$R3
001885  0924  0F80 0840                    B     <XMSCK               B TO MARK AND SPACE CHECK ROUTINE
001886  0926  C380 0B0E              JP2B  LNJ   $B4,<INXT            INPUT NEXT STATUS ON REC. SIDE
001887  0928  E870 1000                    LDR   $R6,=Z'1000'
001888  092A  82D5                         LB    =$R5,=Z'1000'        SEE IF STATUS COMPLETE
        092B  1000
001889  092C  0503                         BBT   >JP3
001890  092D  F380 0BA3                    LNJ   $B7,<ERROR           STATUS NOT COMPLETE ON REC.
001891                              *
001892  092F  8AD3                   JP3   INC   =$R3
001893  0930  C380 0B0E                    LNJ   $B4,<INXT            INPUT NEXT STATUS ON XMIT SIDE
001894  0932  82D5                         LB    =$R5,=Z'1000'        SEE IF STATUS COMPLETE
        0933  1000
001895  0934  0503                         BBT   >JP4A
001896  0935  F380 0BA3                    LNJ   $B7,<ERROR           STATUS INCOMPLETE ON XMIT SIDE
001897  0937  8980 1124              JP4A  CMZ   <RUNFG
001898  0939  0980 0736                    BNE   <RUNCK               BRANCH TO OVERRUN AND UNDERRUN TEST
```

```
                                  *
 093B  8980 110F                          CMZ     <CRCFG                   TEST CRC AND PARITY FLAG
 093D  0903                               BE      >JP4                     FLAG NOT SET JUST CHECK DATA
 093E  D380 044D                          LNJ     $B5,<CRPRT               GO CHECK CRC RESIDUE AND PARITY
                                  *
                                  *
                                  *          CHECK IF DATA LOOP CORRECTLY
                                  *
                                  JP4     CALL    ZV$C,$,DIN,MASK,RANGE,ERAR
 0940  FBC0 0003
 0942  D380 0000          X
 0944  0F80
 0945  0940
 0946  1C00
 0947  111A
 0948  1120
 0949  1131
 094A  8980 1131                          CMZ     <ERAR                    CHECK FOR ERROR
 094C  090D                               BE      >JP5
 094D  D800 1133                          LDR     $R5,<ERAR+2              PUT IS IN $R5
 094F  E800 1132                          LDR     $R6,<ERAR+1              PUT SHOULD BE INTO $R6
 0951  F800 1131                          LDR     $R7,<ERAR                PUT WORD COUNT INTO $R7
 0953  F380 0BA3                          LNJ     $B7,<ERROR
                                          CALL    ZV$CO                    CONTINUE LOOKING FOR ERRORS
 0955  FBF0 0001
 0957  D380 0000          X
 0959  8980 1109                  JP5     CMZ     <CHSZFG                  CHECK CHARACTER SIZE TEST FLAG
 095B  092A                               BE      >JP5A                    IF SET CHECK HIGH ORDER DIGITS FOR ZERO'S
 095C  5C08                               LDV     $R5,=8
 095D  D900 110C                          CMR     $R5,<CS
 095F  0926                               BE      >JP5A
 0960  8600 111A                          CPL     <MASK                    COMPLEMENT MASK TO CHECK OTHER BITS
                                          CALL    ZV$F,DIN1,CPFLG+3,RANGE CLEAR SHOULD BE TO ZERO'S
 0962  FBC0 0003
 0964  D380 0000          X
 0966  0F80
 0967  1D00
 0968  119A
 0969  1120
                                          CALL    ZV$C,DIN1,DIN,MASK,RANGE,ERAR
 096A  FBC0 0003
 096C  D380 0000          X
 096E  0F80
 096F  1D00
 0970  1C00
 0971  111A
 0972  1120
 0973  1131
 0974  8980 1131                          CMZ     <ERAR                    CHECK FOR ERRORS
 0976  090D                               BE      >JP5B
 0977  D800 1133                          LDR     $R5,<ERAR+2              PUT IS INTO $R5
 0979  E800 1132                          LDR     $R6,<ERAR+1              PUT SHOULD BE INTO $R6
 097B  F800 1131                          LDR     $R7,<ERAR                PUT WORD COUNT INTO $R7
 097D  F380 0BA3                          LNJ     $B7,<ERROR               ERROR;  SOME HIGH ORDER BITS SET IN CHARACTER
                                          CALL    ZV$CO                    CONTINUE LOOKING FOR ERRORS
 097F  FBF0 0001
 0981  D380 0000          X
 0983  8600 111A                  JP5B    CPL     <MASK                    RESET MASK
                                  *
 0985  8AD3                       JP5A    INC     =$R3                     FIND NEXT LINE ON OF THIS TYPE
 0986  8F80 1155                          RSTR    <SAV3,=Z'0112'           RESTORE $B3,$B6
 0988  0112
 0989  0F80 0887                          B       <NXLT
                                  *
                                  *
                                  *=================================================================
                                  *
                                  *         CLEAR TEST FLAGS
                                  *
 098B  8700 110F                  CLFG    CL      <CRCFG
 098D  8700 111D                          CL      <PARFG
 098F  8700 1129                          CL      <SBTFG
 0991  8700 112B                          CL      <STPFG
 0993  8700 112C                          CL      <WDBFG
 0995  8700 1106                          CL      <ARFG
 0997  8700 1109                          CL      <CHSZFG
 0999  8700 1124                          CL      <RUNFG
 099B  8383                               JMP     $B3
                                  *
                                  *=================================================================
                                  *
                                  *         LOAD $R5 WITH LCT BASE ADDRESS
                                  *
 099C  D853                       BASE    LDR     $R5,=$R3
 099D  5041                               SOR     $R5,1
 099E  5F40                               MLV     $R5,=X'40'
 099F  8386                               JMP     $B6
                                  *
                                  *=================================================================
                                  *
                                  *                 CRC GENERATOR
                                  *
                                  *            OUTPUT - TAKES TABLE OF DATA AND FORMS ACCUMULATED CRC.
                                  *                            ONLY FOR CRC16 AND CCITT.
                                  *
                                  *       TO USE, MUST SET;
                                  *                R6 - NUMBER OF BITS IN CHAR.
                                  *                R7 - RANGE IN BYTES
                                  *                B3 - TABLE ADDRESS
                                  *                B6 - ADDRESS OF CRC TYPE.
                                  *
                                  *
                                  *          LNJ     $B4,<CRC
                                  *
                                  *
                                  *       $B3 - ADDRESS WHERE CRC ACCUMULATOR IS STORED
                                  *       $B6 - ADDRESS OF CRC TYPE
                                  *
                                  *       $R6 - CONTAINS NUMBER OF BITS IN CHARACTER
                                  *       $R7 - THE NUMBER OF CRC TO BE
                                  *              GENERATED FROM THE TABLE AT $B3
                                  *
                                  *          THE ACCUMULATED CRC IS STORED IN LOCATION CRCAC.
```

```
001985                              *
001986  09A0  8F00 1135            CRC     SAVE    <SAV1,=Z'F994'
        09A2  F994
001987  09A3  C876                        LDR     $R4,+$B6            PUT CRC TYPE INTO $R4
001988  09A4  8751                        CL      =$R1
001989  09A5  8753                        CL      =$R3                CLEAR $R3 FOR LONG SHIFT
001990  09A6  8752                        CL      =$R2
001991  09A7  8256                        NEG     =$R6
001992  09A8  EF00 1125                   STR     $R6,<TLOC           STORE COUNTER
001993  09AA  D2FF                        LLH     $R5,$B3,+$R3
001994  09AB  7001                        SOL     $R7,1
001995  09AC  8257                        NEG     =$R7
001996  09AD  DF51                  CRCA  STR     $R5,=$R1
001997  09AE  5041                        SOR     $R5,1
001998  09AF  9652                        XOR     $R1,=$R2
001999  09B0  2041                        SOR     $R2,1
002000  09B1  1B02                        BEVN    $R1,>CRCB
002001  09B2  A654                        XOR     $R2,=$R4
002002  09B3  67FA                  CRCB  BINC    $R6,>CRCA
002003  09B4  D0FF                        LDH     $R5,$B3,+$R3
002004  09B5  E800 1125                   LDR     $R6,<TLOC           RESET CHAR. COUNT
002005  09B7  77F6                        BINC    $R7,>CRCA           DO NEW CHARACTER
002006  09B8  AF00 110E                   STR     $R2,<CRCAC          STORE LAST CRC ACCUMULATED
002007  09BA  8F80 1135                   RSTR    <SAV1,=Z'F994'      RESTORE REGS.
        09BC  F994
002008  09BD  8384                        JMP     $B4
002009                              *
002010                              *=====================================================================
002011                              *
002012                              *           PARITY CALCULATOR
002013                              *
002014                              *                     TO USE, MUST SET;
002015                              *                        $R7 - RANGE  OF DATA IN WORDS
002016                              *                        $B1 - ADDRESS OF DATA AFTER PARITY HAS BEEN ADDED
002017                              *                        $B3 - ADDRESS OF DATA TO ADD PARITY ON
002018                              *                        PTTY - 1MEANS ODD PARITY
002019                              *                               0 MEANS EVEN PARITY
002020                              *
002021  09BE  8F00 1135            CPAR  SAVE    <SAV1,=Z'EFD0'      SAVE REGS.
        09C0  EFD0
002022  09C1  9800 110C                   LDR     $R1,<CS             LDR $R1 WITH NUMBER TO SHIFT
002023  09C3  88D1                        DEC     =$R1
002024  09C4  8755                        CL      =$R5
002025  09C5  8752                        CL      =$R2                CLEAR INDEX
002026  09C6  8257                        NEG     =$R7                NEGATE RANGE FOR BRANCH AND INCREMENT
002027  09C7  C2A3                  CPAR2 LLH     $R4,$B3,$R2         LOAD $R4 WITH HALF WORD OF DATA BUFFER
002028  09C8  8980 1109                   CMZ     <CHSZFG             SEE IF CHAR. SIZE TEST
002029  09CA  090F                        BE      >CPAR3
002030  09CB  6C08                        LDV     $R6,=8              IF 8 BIT CHAR, DO NOT STRIP BITS
002031  09CC  E900 110C                   CMR     $R6,<CS
002032  09CE  090B                        BE      >CPAR3
002033  09CF  88D6                        DEC     =$R6                IF 7 OR 6 BIT CHAR, STRIP FOR PARITY TO BE SE
002034  09D0  E900 110C                   CMR     $R6,<CS
002035  09D2  0904                        BE      >CPAR4
002036  09D3  E870 001F                   LDR     $R6,=X'1F'          MASK TO STRIP WITH 6 BIT CHAR.
002037  09D5  0F83                        B       >CPAR5
002038  09D6  E870 003F            CPAR4 LDR     $R6,=X'3F'          MASK TO STRIP WITH 7 BIT CHAR.
002039  09D8  C556                  CPAR5 AND     $R4,=$R6            STRIP BIT TO CHAR SIZE MINUS PARITY BIT
002040  09D9  CF56                  CPAR3 STR     $R4,=$R6            SAVE CHAR IN $R6
002041                              *
002042                              *           CALCULATE PARITY FOR THE CHAR.
002043                              *
002044  09DA  50C4                        DOR     $R5,4
002045  09DB  CF00 1125                   STR     $R4,<TLOC
002046  09DD  8754                        CL      =$R4
002047  09DE  5084                        DOL     $R5,4
002048  09DF  C600 1125                   XOR     $R4,<TLOC
002049  09E1  50C2                        DOR     $R5,2
002050  09E2  CF00 1125                   STR     $R4,<TLOC
002051  09E4  8754                        CL      =$R4
002052  09E5  5082                        DOL     $R5,2
002053  09E6  C600 1125                   XOR     $R4,<TLOC
002054  09E8  4041                        SOR     $R4,1
002055  09E9  0682                        BCF     >CPAR1
002056  09EA  8AD4                        INC     =$R4
002057  09EB  C600 111E            CPAR1 XOR     $R4,<PTTY
002058                              *
002059  09ED  4000                        SOL     $R4,0               SHIFT PARITY BIT TO POSITION
002060  09EE  E454                        OR      $R6,=$R4            PUT PARITY BIT INTO DATA
002061  09EF  E7ED                        STH     $R6,$B1,+$R2        STORE INTO DATA SHOULD BE
002062  09F0  7780 09C7                   BINC    $R7,<CPAR2          DO NEXT CHARACTER
002063  09F2  8F80 1135                   RSTR    <SAV1,=Z'EFD0'      RESTORE REG.
        09F4  EFD0
002064  09F5  8384                        JMP     $B4                 JUMP OUT OF PARITY FORMATION SUB-ROUTINE
002065                              *
002066                              *=====================================================================
002067                              *
002068                              *           SET TEST MODE ON FIRST LINE OF CLA
002069                              *
002070                              *                     LNJ $B4,<STMD
002071                              *
002072  09F6  8F00 1145            STMD  SAVE    <SAV2,=Z'DF8C'      SAVE REGS.
        09F8  DF8C
002073  09F9  3EFE                        ADV     $R3,=-2             SET $R3 TO RECV. CHAN. ON FIRST LINE
002074  09FA  CB80 0BEE                   LAB     $B4,<CHLP           LOAD $B4 WITH ADDRESS OF CHANNEL PROG.
002075  09FC  0F8E                        B       >STMD2
002076                              *=====================================================================
002077                              *
002078                              *           SET CURRENT LOOP ADAPTER FOR FASTEST SPEED
002079                              *
002080  09FD  8F00 1145            SCLS  SAVE    <SAV2,=Z'DF8C'      SAVE REGS.
        09FF  DF8C
002081  0A00  C800 10FC                   LDR     $R4,<FOCC
002082  0A02  C380 0B7E                   LNJ     $B4,<CGSCH
002083  0A04  8000 117F            SCLS1 IO      <BIIO,=$R4          CHANNEL INITIALIZE
        0A06  0054
002084  0A07  07FD                        BIOF    >SCLS1
002085  0A08  CB80 0BF3                   LAB     $B4,<CHLPS          LOAD $B4 WITH ADDRESS OF CHANNEL PROG.
002086  0A0A  CF80 0A19            STMD2 STB     $B4,<STMD3+4+$AF    STORE ADDRESS FOR ZV$MLW
002087  0A0C  C800 1102                   LDR     $R4,<FINS           LOAD AN I/O TO GET CHANNEL FOR ZV$MLW
002088  0A0E  C380 0B7E                   LNJ     $B4,<CGSCH
002089  0A10  C570 FFC0                   AND     $R4,=Z'FFC0'        STRIP OFF DIRECTION BIT AND FUN CODE
002090  0A12  CF00 110A                   STR     $R4,<CHNZ           STORE CHANNEL FOR CALL
```

```
002091                              *
002092                              STMD3  CALL   ZV$MLW,$,WCP,MX200,CHNZ
       0A14  FBC0 0003
       0A16  D380 0000      X
       0A18  0F80
       0A19  0A14
       0A1A  12D3
       0A1B  125D
       0A1C  110A
002093 0A1D  D380 0B24              LNJ    $B5,<SETLCT          SEND OUT LCT
002094 0A1F  1283                   DC     <LCT
002095                              *
002096 0A20  8757                   CL     =$R7                 CLEAR FOR SHORT DELAY OF 30 MS.
002097 0A21  C800 10FC              LDR    $R4,<FOCC            FORM CONTROL WORD
002098 0A23  C380 0B7E              LNJ    $B4,<CGSCH           FOR
002099 0A25  8000 1180              IO     <BIT1,=$R4           START I/O
       0A27  0054
002100 0A28  0700 0A2C              BIOT   <STMD1
002101 0A2A  F380 0BA3              LNJ    $B7,<ERROR           ERROR;  I/O WAS NAK'ED
002102                              *
002103 0A2C  D380 0B4C              STMD1  LNJ    $B5,<TDLAY           TIME DELAY
002104 0A2E  8F80 1145              RSTR   <SAV2,=Z'DF8C'       RESTORE REGS.
       0A30  DF8C
002105 0A31  8384                   JMP    $B4
002106                              *
002107                              *=====================================================================
002108                              *
002109                              *         CONVERT RTC TICKS TO MILSEC.
002110                              *
002111 0A32  8F00 1135              TKSEC  SAVE   <SAV1,=Z'D380'       SAVE REGS.
       0A34  D380
002112                              *
002113 0A35  CF51                   STR    $R4,=$R1             DUPLICATE VALUE IN $R1
002114 0A36  CB00 1110              MUL    $R4,<DIV1
002115 0A38  CF00 1125              STR    $R4,<TLOC
002116 0A3A  C851                   LDR    $R4,=$R1
002117 0A3B  CB00 1111              MUL    $R4,<DIV2
002118 0A3D  C370 000A              DIV    $R4,=10
002119 0A3F  CA00 1125              ADD    $R4,<TLOC
002120 0A41  CF00 1125              STR    $R4,<TLOC
002121 0A43  C851                   LDR    $R4,=$R1
002122 0A44  CB00 1112              MUL    $R4,<DIV3
002123 0A46  C370 0064              DIV    $R4,=100
002124 0A48  CA00 1125              ADD    $R4,<TLOC
002125 0A4A  CF00 1125              STR    $R4,<TLOC
002126 0A4C  C851                   LDR    $R4,=$R1
002127 0A4D  CB00 1113              MUL    $R4,<DIV4
002128 0A4F  C370 03E8              DIV    $R4,=1000
002129 0A51  CA00 1125              ADD    $R4,<TLOC
002130                              *
002131 0A53  8F80 1135              RSTR   <SAV1,=Z'D380'
       0A55  D380
002132 0A56  8385                   JMP    $B5
002133                              *
002134                              *=====================================================================
002135                              *
002136                              *         CONVERT MS TO BITS/SEC (GIVEN CHAR. SIZE & NUMBER)
002137                              *
002138 0A57  8F00 1135              BPS    SAVE   <SAV1,=Z'DF80'       SAVE REGS.
       0A59  DF80
002139 0A5A  8756                   CL     =$R6                 CLEAR $R6 FOR DIVIDE
002140 0A5B  9826                   LDR    $R1,$B6,$R2          LOAD $R1 WITH NUMBER OF CHAR.
002141 0A5C  9970 0F00              CMR    $R1,=3840
002142 0A5E  0983                   BNE    >BPS1
002143 0A5F  1E07                   ADV    $R1,=7
002144 0A60  9255                   SUB    $R1,=$R5
002145 0A61  F870 03E8              BPS1   LDR    $R7,=1000            CONVERSION FACTOR FOR MS TOC.
002146 0A63  FB00 110D              MUL    $R7,<CSB             MULTIPLY BY CHARACTER SIZE
002147 0A65  F354                   DIV    $R7,=$R4             DIVIDE BY ACTUAL TIME IN MS
002148 0A66  8D00 1126              SDI    <TP1                 STORE DOUBLE INTEGER
002149 0A68  8756                   CL     =$R6
002150 0A69  F800 1127              LDR    $R7,<TP2
002151 0A6B  FB51                   MUL    $R7,=$R1
002152 0A6C  82D7                   LB     =$R7,=Z'8000'
       0A6D  8000
002153 0A6E  0502                   BBT    >BPS2A
002154 0A6F  6903                   BEZ    $R6,>BPS2
002155 0A70  8A80 1117              BPS2A  INC    <HGSPD
002156 0A72  8D00 111B              BPS2   SDI    <MSBS
002157 0A74  8D30 11E2              SDI    <SYLSPO,$R3
002158 0A76  F800 1126              LDR    $R7,<TP1
002159 0A78  7F0A                   MLV    $R7,=10
002160 0A79  F354                   DIV    $R7,=$R4
002161 0A7A  FF55                   STR    $R7,=$R5
002162 0A7B  DB51                   MUL    $R5,=$R1
002163 0A7C  D370 000A              DIV    $R5,=10
002164 0A7E  DA00 111C              ADD    $R5,<MSBS+1
002165 0A80  70D0                   DOR    $R7,16
002166 0A81  7F0A                   MLV    $R7,=10
002167 0A82  F354                   DIV    $R7,=$R4
002168 0A83  FB51                   MUL    $R7,=$R1
002169 0A84  F370 0064              DIV    $R7,=100
002170 0A86  FA55                   ADD    $R7,=$R5
002171 0A87  FF00 111C              STR    $R7,<MSBS+1
002172 0A89  8F80 1135              RSTR   <SAV1,=Z'DF80'
       0A8B  DF80
002173 0A8C  8385                   JMP    $B5
002174                              *
002175                              *=====================================================================
002176                              *
002177                              *           PRINT LINE SPEED
002178                              *
002179 0A8D  8F00 1145              PRTLSP SAVE   <SAV2,=Z'B88F'       SAVE REGS.
       0A8F  B88F
002180 0A90  3041                   SOR    $R3,1                SHIFT $R3 TO REPRESENT LINE NUMBER
002181 0A91  BF00 1125              STR    $R3,<TLOC            STORE LINE NUMBER FOR ZV$TD
002182                              CALL   ZV$TC,MESG2,BYTP     PRINT LINE TYPE
       0A93  FBC0 0003
       0A95  D380 0000      X
       0A97  0F80
       0A98  1313
       0A99  1107
002183                              CALL   ZV$TD,TLOC           PRINT LINE NUMBER
```

```
        0A9A  FBC0 0003
        0A9C  D380 0000      X
        0A9E  0F80
        0A9F  1125
002184                              CALL   ZV$T,MESG4          PRINT 'SPEED'
        0AA0  FBC0 0003
        0AA2  D380 0000      X
        0AA4  0F80
        0AA5  131C
002185  0AA6  D380 0A32             LNJ    $B5,<TKSEC          CONVERT NUMBER OF TICK IN $R4 TO MS
002186                         *
002187  0AA8  D380 0A57             LNJ    $B5,<BPS            USE $R4 TO CALCULATE BITS/SEC
002188  0AAA  8980 1117             CMZ    <HGSPD              CHECK FOR HIGH SPEED LINE
002189  0AAC  0919                  BE     >PRLT3
002190  0AAD  8C80 111B             LDI    <MSBS               LOAD DOUBLE INTERGER INTO $R6,$R7
002191  0AAF  F370 03E8             DIV    $R7,=1000           FIND NUMBER OF KB
002192  0AB1  FF00 111B             STR    $R7,<MSBS
002193  0AB3  6D64                  CMV    $R6,=100
002194  0AB4  0A02                  BAG    >PRLT5
002195  0AB5  6C64                  LDV    $R6,=100
002196  0AB6  EF00 111C      PRLT5  STR    $R6,<MSBS+1
002197                              CALL   ZV$TD,MSBS,BKLS     PRINT NUMBER OF KB
        0AB8  FBC0 0003
        0ABA  D380 0000      X
        0ABC  0F80
        0ABD  111B
        0ABE  119D
002198                              CALL   ZV$T,COMMA
        0ABF  FBC0 0003
        0AC1  D380 0000      X
        0AC3  0F80
        0AC4  119C
002199                       PRLT3  CALL   ZV$TD,MSBS+1,BKLS   PRINT LINE SPEED
        0AC5  FBC0 0003
        0AC7  D380 0000      X
        0AC9  0F80
        0ACA  111C
        0ACB  119D
002200  0ACC  8980 129C             CMZ    <LC3SYF             IF SYNC. LINE ALL DONE
002201  0ACE  0999                  BNE    >PRLT1              IF ASYNC. PRINT SHOULD BE
002202  0ACF  EB80 1212             LAB    $B6,<ALSPBL         LOAD ADDRESS OF 'SHOULD BE' SPEED TABLE
002203  0AD1  C870 2108             LDR    $R4,=X'2108'        IF NEW LINE SPEED CHANGE $B6 TO POINT TO TABL
002204  0AD3  C908 0320             CMR    $R4,*<ST1
002205  0AD5  0903                  BE     >PRLT7
002206  0AD6  EB80 1221             LAB    $B6,<CMSPBL
002207                       PRLT7  CALL   ZV$T,MESG5          PRINT 'SHOULD BE'
        0AD8  FBC0 0003
        0ADA  D380 0000      X
        0ADC  0F80
        0ADD  1322
002208  0ADE  B826                  LDR    $R3,$B6,$R2         LOAD $R3 WITH LINE SPEED
002209  0ADF  BF00 1125             STR    $R3,<TLOC
002210                              CALL   ZV$TD,TLOC
        0AE1  FBC0 0003
        0AE3  D380 0000      X
        0AE5  0F80
        0AE6  1125
002211  0AE7  9800 111C      PRLT1  LDR    $R1,<MSBS+1         STORE AWAY LINE SPEED (SYNC)
002212  0AE9  9F00 1118             STR    $R1,<MISPD
002213  0AEB  8700 111B             CL     <MSBS
002214  0AED  8700 111C             CL     <MSBS+1
002215  0AEF  8F80 1145             RSTR   <SAV2,=Z'B88F'      RESTORE R
        0AF1  B88F
002216  0AF2  8384                  JMP    $B4
002217                         *
002218                         *=============================================================
002219                         *
002220                         * GENERAL INITIALIZE FOR DLCC
002221                         *
002222  0AF3  8F00 1145      GENITZ SAVE   <SAV2,=Z'1118'      SAVE B7,$R7
        0AF5  1118
002223  0AF6  7C1A                  LDV    $R7,=26             ALLOW 800 MS DELAY
002224  0AF7  C800 10FA             LDR    $R4,<FOMC           GET FUNCTION CODE FOR INITIALIZE
002225  0AF9  C380 0B7E             LNJ    $B4,<CGSCH          MODIFY FOR CHANNEL
002226  0AFB  8000 117F             IO     <BITO,=$R4          INITIALIZE
        0AFD  0054
002227  0AFE  0703                  BIOT   >N2
002228  0AFF  F380 0BA3             LNJ    $B7,<ERROR          INITIALIZE WAS NAK'ED
002229  0B01  D380 0B4C      N2     LNJ    $B5,<TDLAY          WAIT 30 MS
002230  0B03  8F80 1145             RSTR   <SAV2,=Z'1118'
        0B05  1118
002231  0B06  8384                  JMP    $B4                 RETURN
002232                         *
002233                         *=============================================================
002234                         *
002235                         *           OUTPUT CHANNEL CONTROL
002236                         *
002237                         *              $R2 - CONTAINS CHANNEL CONTROL TYPE
002238                         *
002239  0B07  8F00 1135      STCR   SAVE   <SAV1,=Z'0008'      SAVE REG.
        0B09  0008
002240  0B0A  D852                  LDR    $R5,=$R2
002241  0B0B  C800 10FC             LDR    $R4,<FOCC           LOAD FUNCTION CODE FOR CHANNEL CONTROL
002242  0B0D  0F8C                  B      >N1A
002243                         *=============================================================
002244                         *
002245                         * INPUT NEXT STATUS TO R5
002246                         *
002247  0B0E  8F00 1135      INXT   SAVE   <SAV1,=Z'0008'      B4
        0B10  0008
002248  0B11  C800 1102             LDR    $R4,<FINS           GET CONTROL WORD FOR INPUT NEXT STATUS
002249  0B13  0F86                  B      >N1A
002250                         *=============================================================
002251                         *
002252                         *           INPUT RANGE
002253                         *
002254                         *            $R5 - RESULT OF RANGE INPUTED
002255                         *
002256  0B14  8F00 1135      INRG   SAVE   <SAV1,=Z'0008'      SAVE REG.
        0B16  0008
002257  0B17  C800 1100             LDR    $R4,<FIR            LOAD $R4 WITH FUNCTION CODE FOR INPUT RANGE
002258  0B19  C380 0B7E      N1A    LNJ    $B4,<CGSCH          MODIFY FOR CHANNEL
002259  0B1B  8055                  IO     =$R5,=$R4
```

```
      0B1C  0054
002260 0B1D  0703                    BIOT   >N1
002261 0B1E  F380 0BA3               LNJ    $B7,<ERROR          ERROE:  I/O WAS NAK'ED
002262 0B20  8F80 1135        N1     RSTR   <SAV1,=Z'0008'
      0B22  0008
002263 0B23  8384                    JMP    $B4
002264                               *
002265                               *=============================================================
002266                               *
002267                               * SET UP LCT FOR CHANNEL SPECIFIED IN R3
002268                               *
002269                               *            LNJ $B5,<SETLCT
002270                               *            DC  TABLE
002271                               *
002272 0B24  8F00 1135        SETLCT SAVE   <SAV1,=Z'E8E0'       R1,R2,R4,B2,B1
      0B26  E8E0
002273 0B27  9CF5                    LDB    $B1,+$B5            GET ADDRESS OF TABLE
002274 0B28  8751                    CL     =$R1
002275                               *
002276 0B29  A811             LCT4A  LDR    $R2,$B1,$R1         GET BYTE TO OUTPUT
002277 0B2A  8AD1                    INC    =$R1
002278 0B2B  2985                    BNEZ   $R2,>LCT5A          BRANCH IF NOT AT END OF TABLE
002279 0B2C  8F80 1135               RSTR   <SAV1,=Z'E8E0'
      0B2E  E8E0
002280 0B2F  8385                    JMP    $B5                 RETURN
002281                               *
002282 0B30  C800 10FE        LCT5A  LDR    $R4,<FOBL           FUNCTION CODE FOROUT LCT BYTE
002283 0B32  C380 0B7E               LNJ    $B4,<CGSCH          FORM IO CONTROL WORD
002284 0B34  8052             LCT3A  IO     =$R2,=$R4           OUTPUT BYTE
      0B35  0054
002285 0B36  07FE                    BIOF   >LCT3A              CHECK IF TAKEN
002286 0B37  0FF2                    B      >LCT4A              GET NEXT BYTE
002287                               *
002288                               *=============================================================
002289                               *
002290                               *          FIND FIRST ACTIVE LINE
002291                               *
002292                               *            $R3=START CHECK ON THIS LINE AND
002293                               *                CYCLE THROUGH ALL LINES IF
002294                               *                ONE FOUND ON LINE NUMBER IS HERE
002295                               *
002296                               *            LNJ  $B5,<FLN
002297                               *              DC  OF LINE TYPE  TO SEARCH FOR
002298                               *            <RETURN> IF NO LINES ON
002299                               *            <RETURN>
002300                               *
002301 0B38  8F00 1135        FLN    SAVE   <SAV1,=Z'A0E0'      SAVE REG. $R2,$B2,$B1
      0B3A  A0E0
002302 0B3B  3041                    SOR    $R3,1
002303 0B3C  9CF5                    LDB    $B1,+$B5
002304 0B3D  A801                    LDR    $R2,$B1             PUT LINE TYPE INTO $R2
002305 0B3E  AB80 1186               LAB    $B2,<ATLT           ADDRESS OF ACTIVE LINE TABLE
002306 0B40  3D08             FLN2   CMV    $R3,=8              SEE IF AT BOTTOM OF TABLE
002307 0B41  0287                    BGE    >FLN3
002308 0B42  A97E                    CMR    $R2,$B2,+$R3        SEE IF INDEX LINE ON
002309 0B43  09FD                    BNE    >FLN2
002310 0B44  88D3                    DEC    =$R3                SET $R3 BACK TO VALUE FOUND
002311 0B45  3001                    SOL    $R3,1
002312 0B46  DBC5 0001               LAB    $B5,$B5,1           INCREMENT TO VALID RETURN
002313 0B48  8F80 1135        FLN3   RSTR   <SAV1,=Z'A0E0'      RESTORE REG.
      0B4A  A0E0
002314 0B4B  8385                    JMP    $B5                 JUMP OUT OF ROUTINE
002315                               *
002316                               *=============================================================
002317                               *
002318                               *                    TIME DELAY
002319                               *
002320                               *          LNJ  $B5,<TDLAY
002321                               *
002322 0B4C  8F00 1135        TDLAY  SAVE   <SAV1,=Z'C180'      SAVE REG. $R1,$R7
      0B4E  C180
002323 0B4F  1C09             TDLA   LDV    $R1,=9              LOAD CLOCK INITIAL COUNT
002324 0B50  9F00 0000        X      STR    $R1,<ZHRTCC
002325 0B52  1C05                    LDV    $R1,=5
002326 0B53  0004                    RTCN                       TURN ON REAL TIME CLOCK
002327 0B54  9900 0000        X      TDLA1  CMR    $R1,<ZHRTCC
002328 0B56  0882                    BAGE   >TDLA3
002329 0B57  0FFD                    B      >TDLA1
002330 0B58  0005             TDLA3  RTCF                       TURN OFF REAL TIME CLOCK
002331 0B59  7776                    BDEC   $R7,>TDLA
002332 0B5A  8F80 1135               RSTR   <SAV1,=Z'C180'      RESTORE $R1
      0B5C  C180
002333 0B5D  8385                    JMP    $B5
002334                               *
002335                               *=============================================================
002336                               *
002337                               *          READ LCT
002338                               *
002339                               *           LNJ  $B6,<RDLCT
002340                               *           DC   <ADDRESS>
002341                               *           DC   <RANGE IN BYTES>
002342                               *
002343 0B5E  8F00 1135        RDLCT  SAVE   <SAV1,=Z'C88C'      SAVE REGS.
      0B60  C88C
002344 0B61  C800 1100               LDR    $R4,<FIR            LOAD DUMMY I/O
002345 0B63  C380 0B7E               LNJ    $B4,<CGSCH          PUT PROPER CHANNEL IN
002346 0B65  C570 FF80               AND    $R4,=Z'FF80'
002347 0B67  CF00 110A               STR    $R4,<CHNZ           STORE CHANNEL NUMBER  FOR CALL
002348 0B69  C806                    LDR    $R4,$B6             STORE ADDRESS IN ZV$MLR CALL
002349 0B6A  CF00 1122               STR    $R4,<RLCTAD
002350 0B6C  C846 0001               LDR    $R4,$B6,1           STORE RANGE FOR ZV$MLR
002351 0B6E  CF00 1123               STR    $R4,<RLCTRG
002352                               CALL   ZV$MLR,LCTV,RLCTRG,RLCTAD,CHNZ
      0B70  FBC0 0003
      0B72  D380 0000        X
      0B74  0F80
      0B75  112E
      0B76  1123
      0B77  1122
      0B78  110A
002353 0B79  8F80 1135               RSTR   <SAV1,=Z'C88C'
      0B7B  C88C
002354 0B7C  83C6 0002               JMP    $B6,2
```

```
002355                                  *
002356                                  *=============================================================
002357                                  *
002358                                  *              CHANGE SUBCHANNEL
002359                                  *
002360                                  *     $R3 -CONTAINS SUBCHANNEL WANTED
002361                                  *     $R4 -CONTAINS I/O CONTROL WORD TO BE CHANGED
002362                                  *
002363                                  *      LNJ     $B4,<CGSCH
002364                                  *
002365                                  *
002366  0B7E  3006                     CGSCH  SOL     $R3,6                SHIFT TO SUBCHANNEL POSITION
002367  0B7F  C453                            OR      $R4,=$R3             OR SUBCHANNEL NUMBER INTO CONTROL WORD
002368  0B80  3046                            SOR     $R3,6                SHIFT TO NORMAL POSITION
002369  0B81  8384                            JMP     $B4                  GO BACK
002370                                  *
002371                                  *=============================================================
002372                                  *
002373                                  *       CCB FORMATION
002374                                  *
002375                                  *        $R3 - CONTAINS SUBCHANNEL WANTED
002376                                  *        $R4 - CONTAINS RANGE (NUMBER OF BYTES)
002377                                  *
002378                                  *        LNJ     $B5,<MCCB
002379                                  *        DC      CPU ADDRESS
002380                                  *        DC      RAM ADDRESS NUMBER OR CHANNEL CONTROL WORD.
002381                                  *
002382  0B82  8F00 1135                MCCB   SAVE    <SAV1,=Z'FDF8'       SAVES $B1,$B3,$B2,$B4,$R7,$R5,$R4,R2,6 $R1
      0B84  FDF8
002383  0B85  7C7E                            LDV     $R7,=126
002384  0B86  ACF5                            LDB     $B2,+$B5             LOAD $B2 WITH CPU ADDRESS
002385  0B87  A805                            LDR     $R2,$B5              PUT RAM ADDRESS IN $R2
002386  0B88  D854                            LDR     $R5,=$R4
002387  0B89  C800 10FD                       LDR     $R4,<FOLD            LOAD $R4 WITH I/O CONTROL WORD
002388  0B8B  C380 0B7E                       LNJ     $B4,<CGSCH
002389  0B8D  7900 0B9C                MCB5   BEZ     $R7,<MCER
002390  0B8F  8182                            IOLD    $B2,=$R4,=$R5        OUTPUT ADDRESS AND RANGE
      0B90  0054
      0B91  0055
002391  0B92  88D7                            DEC     =$R7
002392  0B93  07FA                            BIOF    >MCB5
002393  0B94  C800 10FF                       LDR     $R4,<FOCB            LOAD $R4 WITH I/O CONTROL WORD
002394  0B96  C380 0B7E                       LNJ     $B4,<CGSCH           PUT I/O CONTROL WORD IN $R4
002395  0B98  8052                MCB2   IO      =$R2,=$R4            OUTPUT DLCC RAM ADDRESS
      0B99  0054
002396  0B9A  07FE                            BIOF    >MCB2
002397  0B9B  0F83                            B       >MCB6
002398  0B9C  F380 0BA3                MCER   LNJ     $B7,<ERROR           IOLD HAS BEEN NAK'ED
002399  0B9E  8F80 1135                MCB6   RSTR    <SAV1,=Z'FDF8'       RESTORE REGS.
      0BA0  FDF8
002400  0BA1  83C5 0001                       JMP     $B5.1
002401                                  *
002402                                  *=============================================================
002403                                  *
002404                                  *              ERROR PRINT ROUTINE
002405                                  *
002406                                  *
002407  0BA3  8F00 1165                ERROR  SAVE    <SAV4,=Z'FFFF'       SAVE REGISTERS
      0BA5  FFFF
002408  0BA6  8800 1364                       LBF     <ERMG+1,=X'FE'
      0BA8  00FF
002409  0BA9  3041                            SOR     $R3,1                SHIFT TO GET LINE NUMBER
002410  0BAA  3E30                            ADV     $R3,=X'30'           CHANGE NUMBER INTO ASCII
002411  0BAB  B400 1364                       OR      $R3,<ERMG+1
002412  0BAD  BF00 1364                       STR     $R3,<ERMG+1          STORE LINE NUMBER
002413  0BAF  1CF0                            LDV     $R1,=-16             INTIALIZE LOOP COUNTER
002414  0BB0  8753                            CL      =$R3                 CLEAR INDEX
002415  0BB1  EBC7 FFFE                       LAB     $B6,$B7.-1-$AF       DECREMENT $B7
002416  0BB3  EF80 0BBA                       STB     $B6,<ERR1+4+$AF      STORE ERROR ADDRESS FOR ERROR CALL
002417                                 ERR1   CALL    ZV$ER,$,ERMG
      0BB5  FBC0 0003
      0BB7  D380 0000      X
      0BB9  0F80
      0BBA  0BB5
      0BBB  1363
002418  0BBC  EB80 119B                       LAB     $B6,<CRLF            PUT CR-L INTO CALL
002419  0BBE  CB80 1245                       LAB     $B4,<SPACE
002420  0BC0  AB80 0BD2                       LAB     $B2,<ERR3+4+2*$AF
002427  0BC2  EF82                    ER11   STB     $B6,$B2
002428  0BC3  8980 1114                       CMZ     <ERF                 CHECK FLAG FOR SHORT REPORT
002429  0BC5  0900 0BDD                       BE      <ERR5
002430  0BC7  BBB0 1165                ERR2   LAB     $B3,<SAV4,$R3        FECTH AND STORE REGISTER
002431  0BC9  9B80 0BD1                       LAB     $B1,<ERR3+4+$AF
002437  0BCB  BF81                    ERR21  STB     $B3,$B1
002445                                 ERR3   CALL    ZV$THZ,$,$
      0BCC  FBC0 0003
      0BCE  D380 0000      X
      0BD0  0F80
      0BD1  0BCC
      0BD2  0BCC
002450  0BD3  8AD3                    ERR34  INC     =$R3                 BUMP INDEX
002451  0BD4  CF82                           STB     $B4,$B2
002452  0BD5  3D08                           CMV     $R3,=7*$AF+1         SEE IF CR-LF NEEDED
002453  0BD6  0984                           BNE     >ERR4
002454  0BD7  EB80 119B                       LAB     $B6,<CRLF
002455  0BD9  EF82                           STB     $B6,$B2              PUT CR-LF INTO CALL
002456  0BDA  1780 0BC7                ERR4   BINC    $R1,<ERR2            DO NEXT REGISTER
002457  0BDC  0000                           HLT                          HALT AFTER ERROR
002458  0BDD  8F80 1165                ERR5   RSTR    <SAV4,=Z'FFFF'       RESTORE REGISTERS
      0BDF  FFFF
002459  0BE0  8387                           JMP     $B7                  RETURN
002469                                 SAFF   NULL
002470                                 *
002471                                 *=============================================================
002472                                 *
002473                                 *              END OF PASS CHECK
002474                                 *
002475  0BE1  8700 00FF                EDPCK  CL      <STOP
002476  0BE3  8980 0000      X                CMZ     <ZV$TTY
002477  0BE5  0900 00FF                        BE      <STOP
002478                                         CALL    ZV$TC,MESG8
      0BE7  FBC0 0003
      0BE9  D380 0000      X
```

```
       0BEB  0F80
       0BEC  1333
002479 0BED  8382                    JMP    $B2
002480                               *
002481                               *          -VC-VH-VA-VN-VN-VE-VL -VP-VR-VO-VG-VK-VA-VM-VS
002482                               *
002483                               *
002484                               *
002485                               *
002486                               *        SET TEST MODE ON CLA FOR SECOND LINE OF CLA
002487                               *
002488                               *       ORG    X'200'
002489       0BEE                    CHLP   EQU    $
002490                               *       LD     20
002493 0BEE  D614
002494                               *       OUT    2
002495 0BEF  BD30
002498                               *       WAIT
002499 0BF0  28BD
002500 0BF1  3100
002501                               *       NOP
002502                               *
002503                               *
002504                               *        SET CURRENT LOOP ADAPTER TO FASTEST SPEED
002505                               *
002506                               *       ORG    X'200'
002507 0BF2  0100
002508       0BF3                    CHLPS  EQU    $
002509                               *       LD     =X'F'
002512 0BF3  C60F
002513                               *       OUT    4
002514 0BF4  BD30
002517                               *       WAIT
002518 0BF5  30BD
002519 0BF6  3100
002520                               *       NOP
002521                               *
002522                               *
002523                               *                  CHANNEL PROGRAM FOR LOOP AT LINE ADAPT (REC)
002524                               *
002525                               *
002526                               *       ORG    X'200'
002527 0BF7  0100
002528       0BF8                    CHLP1  EQU    $
002529                               *       LOC    HLP1
002530       0200                    HLP1 EQU X'0200'
002531                               *       LD     24              GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
002534 0BF8  D618
002535                               *       OUT    4               OUTPUT IT
002536 0BF9  BD30
002539                               *       RECV   0               DUMMY REC. FOR CURRENT LOOP ADAPT
002540 0BFA  30BD
002543 0BFB  3060
002544                               *       LD     20              GET LINE CONTROL
002547 0BFC  D614
002548                               *       OUT    2               OUTPUT IT
002549 0BFD  BD30
002552                               *       WAIT
002553 0BFE  28BD
002554 0BFF  3100
002555                               *       LD     28              LOAD SYNC. FLAG IF ZERO LINE IS ASYNC.
002558 0C00  D61C
002559                               *       BZT    GETC
002560 0C01  5D27
002563                               *       LOC    TRNX
002564       0215                    TRNX EQU X'0215'
002565                               *       RECV   0               INPUT 1ST SYNC
002566 0C02  3ABD
002569 0C03  3060
002570                               *       WAIT
002571 0C04  BD31
002572                               *       RECV   0               INPUT SECOND SYNC
002573 0C05  00BD
002576 0C06  3060
002577                               *       C      24
002580 0C07  9618
002581 0C08  1097
002582                               *       BET    INSY
002583 0C09  4196
002584 0C0A  4127
002587                               *       LD     20
002588 0C0B  0ED6
002591                               *       AND    =X'FD'
002592 0C0C  14C4
002595                               *       OUT    2
002596 0C0D  FDBD
002599 0C0E  3028
002600                               *       LD     20
002603 0C0F  D614
002604                               *       OUT    2
002605 0C10  BD30
002608                               *       B      TRNX
002609 0C11  2820
002612                               *       LOC    INSY
002613       0235                    INSY EQU X'0235'
002614                               *       WAIT
002615 0C12  E0BD
002616 0C13  3100
002617                               *       RECV   0               INPUT FINAL SYNC.
002618 0C14  BD30
002621                               *       WAIT
002622 0C15  60BD
002623 0C16  3100
002624                               *       LOC    CKST
002625       023E                    CKST EQU X'023E'
002626                               *       RECV   0               INPUT FIRST CHAR.
002627 0C17  BD30
002630                               *       C      =X'FF'          SEE IF START OF DATA OR SYNC.
002631 0C18  6086
002634 0C19  FF10
002635 0C1A  9741
002636                               *       BET    GETC1
002637 0C1B  9641
```

```
002640  0C1C  270F
002641                        *       WAIT
002642  0C1D  BD31
002643                        *       B       CKST
002644  0C1E  0020
002647                        *       LOC     GETC
002648        024F            GETC EQU X'024F'
002649                        *       RECV    0                       INPUT CHARACTER
002650  0C1F  EFBD
002653  0C20  3060
002654                        *       ST,     STORE                   INTO DATA BUFFER
002655  0C21  BD30
002656                        *       BLCT    HLP1A
002657  0C22  C0DE
002658  0C23  6327
002661                        *       LOC     GETC1
002662        0259            GETC1 EQU X'0259'
002663                        *       WAIT
002664  0C24  05BD
002665  0C25  3100
002666                        *       B       GETC
002669  0C26  20F1
002670                        *       LOC     HLP1A
002671        025E            HLP1A EQU X'025E'
002672                        *       BLBT    GETX
002673  0C27  9665
002674  0C28  8520
002677  0C29  2605
002678                        *       GNB
002679  0C2A  BD31
002680                        *       B       GETC
002681  0C2B  2020
002684                        *       LOC     GETX
002685        0269            GETX EQU X'0269'
002686                        *       LD      =0                      TURN OFF REC.
002687  0C2C  E6C6
002690                        *       OUT     2
002691  0C2D  00BD
002694  0C2E  3028
002695                        *       GNB
002696  0C2F  BD31
002697                        *       WAIT
002698  0C30  20BD
002699  0C31  3100
002700                        *       B       HLP1
002703  0C32  208A
002704                        *       NOP
002705                        *
002706                        *
002707                        *         CHANNEL PROGRAM FOR LOOP AT LINE ADAPT (XMIT)
002708                        *
002709                        *
002710                        *       ORG     X'400'
002711  0C33  0100
002712        0C34            CHLP1A EQU    $
002713                        *       LOC     LP1A
002714        0400            LP1A EQU X'0400'
002715                        *       LD      2                       GET LINE CONFIGURATION
002718  0C34  D602
002719                        *       OUT     6                       OUTPUT IT
002720  0C35  BD30
002723                        *       LD      =X'AA'                  LOAD FILL CHAR.
002724  0C36  38C6
002727                        *       OUT     4                       OUTPUT SYNC. FILL CHAR.
002728  0C37  AABD
002731  0C38  3030
002732                        *       WAIT
002733  0C39  BD31
002734                        *       LD      28                      LOAD SYNC. FLAG
002735  0C3A  00D6
002738                        *       BZT     LP1ABD                  IF ZERO BRANCH TO OUTPUT CHAR.
002739  0C3B  1C5D
002742  0C3C  271F
002743                        *       LD      =6
002746  0C3D  C606
002747                        *       LOC     XMSY
002748        0414            XMSY EQU X'0414'
002749                        *       ST      63
002752  0C3E  D73F
002753                        *       LD      24                      LOAD SYNC. CHAR
002756  0C3F  D618
002757                        *       SEND    0                       SEND IT
002758  0C40  BD30
002761                        *       LD      63                      DO SIX TIMES
002762  0C41  40D6
002765                        *       DEC
002766  0C42  3F5A
002767                        *       BZT     SDSY
002768  0C43  5D27
002771                        *       WAIT
002772  0C44  05BD
002773  0C45  3100
002774                        *       B       XMSY
002777  0C46  20EE
002778                        *       LOC     SDSY
002779        0426            SDSY EQU X'0426'
002780                        *       WAIT
002781  0C47  BD31
002782                        *       LD      =X'FF'
002783  0C48  00C6
002786                        *       SEND    0
002787  0C49  FFBD
002790  0C4A  3040
002791                        *       WAIT
002792  0C4B  BD31
002793                        *       LOC     LP1ABD
002794        0431            LP1ABD EQU X'0431'
002795                        *       LD,     GET                     CHARACTER FROM DATA BUFFER
002796  0C4C  00BD
002797  0C4D  30A0
002798                        *       SEND    0                       OUTPUT IT
002799  0C4E  BD30
002802                        *       BLCT    LP1AB
```

```
002803  0C4F  40DE
002804  0C50  6327
002807                        *       LOC     LP1AA
002808          043B          LP1AA EQU X'043B'
002809                        *       WAIT
002810  0C51  05BD
002811  0C52  3100
002812                        *       B       LP1ABD
002815  0C53  20F1
002816                        *       LOC     LP1AB
002817          0440          LP1AB EQU X'0440'
002818                        *       WAIT
002819  0C54  BD31
002820                        *       LD      =X'FF'
002821  0C55  00C6
002824                        *       SEND    0
002825  0C56  FFBD
002828  0C57  3040
002829                        *       BLBT    LP1AC1
002830  0C58  9665
002831  0C59  8520
002834  0C5A  2605
002835                        *       GNB
002836  0C5B  BD31
002837                        *       B       LP1AA
002838  0C5C  2020
002841                        *       LOC     LP1AC1
002842          0453          LP1AC1 EQU X'0453'
002843                        *       GNB
002844  0C5D  E8BD
002845  0C5E  3120
002846                        *       WAIT
002847  0C5F  BD31
002848                        *       B       LP1A
002849  0C60  0020
002852                        *       NOP
002853  0C61  A501
002854                        *
002855                        *
002856                        *          RECEIVE PROGRAM FOR CRC AND PARITY
002857                        *
002858                        *
002859                        *       ORG     X'200'
002860          0C62          CHLP2   EQU     $
002861                        *       LD      24              GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
002864  0C62  D618
002865                        *       OUT     4               OUTPUT IT
002866  0C63  BD30
002869                        *       LD      20              LOAD LINE CONTROL
002870  0C64  30D6
002873                        *       OUT     2               OUTPUT IT
002874  0C65  14BD
002877  0C66  3028
002878                        *       RECV    0
002879  0C67  BD30
002882                        *       WAIT
002883  0C68  60BD
002884  0C69  3100
002885                        *
002886                        *       LD      24              SEE IF SYNC. LINE
002888  0C6A  D618
002890                        *       C       =X'E'           IF SYNC. SET UP TO  RECV.
002893  0C6B  860E
002894  0C6C  1097
002895                        *       BET     RCVA            4 OR 5 SYNC. CHAR.
002896  0C6D  4196
002897  0C6E  4127
002900                        *       LOC     JHQQ
002901          021B          JHQQ EQU X'021B'
002902                        *       RECV    0               RECEIVE FIRST SYNC CHAR.
002903  0C6F  38BD
002906  0C70  3060
002907                        *       WAIT
002908  0C71  BD31
002909                        *       RECV    0               RECEIVE SECOND SYNC.
002910  0C72  00BD
002913  0C73  3060
002914                        *       C       24              CHECK TO SEE IF TRUE SYNC.
002917  0C74  9618
002918  0C75  1097
002919                        *       BET     JFQQ
002920  0C76  4196
002921  0C77  4127
002924                        *       LD      20              SHUT OFF LINE
002925  0C78  0ED6
002928                        *       AND     =X'FD'
002929  0C79  14C4
002932                        *       OUT     2
002933  0C7A  FDBD
002936  0C7B  3028
002937                        *       LD      20              SHUT LINE BACK ON
002940  0C7C  D614
002941                        *       OUT     2
002942  0C7D  BD30
002945                        *       B       JHQQ
002946  0C7E  2820
002949                        *       LOC     JFQQ
002950          023B          JFQQ EQU X'023B'
002951                        *       WAIT
002952  0C7F  E0BD
002953  0C80  3100
002954                        *       RECV    0               RECEIVE THIRD SYNC.
002955  0C81  BD30
002958                        *       LOC     NSTR
002959          0241          NSTR EQU X'0241'
002960                        *       WAIT
002961  0C82  60BD
002962  0C83  3100
002963                        *       RECV    0               RECEIVE SYNC. OR START OF DATA CHAR. (FF)
002964  0C84  BD30
002967                        *       C       =X'FF'
002968  0C85  6086
002971  0C86  FF10
```

```
002972 0C87 9741
002973                          *       BEF     NSTR
002974 0C88 9641
002977 0C89 26F1
002978                          *       WAIT
002979 0C8A BD31
002980                          *
002981                          *       LOC     RCVA
002982      0253                RCVA EQU X'0253'
002983                          *       LD      23              LOAD FLAG FOR TYPE OF RECV.
002984 0C8B 00D6
002987                          *       C       =1              IF TRUE TEST JUST CRC
002988 0C8C 1786
002991 0C8D 0110
002992 0C8E 9741
002993                          *       BET     RECV1
002994 0C8F 9641
002997 0C90 2718
002998                          *       C       =X'F0'          IF TRUE TEST JUST
003001 0C91 86F0
003002 0C92 1097
003003                          *       BET     RECV2
003004 0C93 4196
003005 0C94 4127
003008                          *
003009                          *       LOC     RECV3
003010      0267                RECV3 EQU X'0267'
003011                          *       RECV    3               TEST CRC AND PARITY
003012 0C95 1EBD
003015 0C96 3078
003016                          *       ST,     STORE           INTO BUFFER
003017 0C97 BD30
003018                          *       BLCT    RCV             BRANCH IF LAST CHARACTER TRUE
003019 0C98 CODE
003020 0C99 6327
003023                          *       WAIT
003024 0C9A 23BD
003025 0C9B 3100
003026                          *       B       RECV3
003029 0C9C 20F1
003030                          *
003031                          *
003032                          *       LOC     RECV1
003033      0276                RECV1 EQU X'0276'
003034                          *       RECV    1               TEST CRC
003035 0C9D BD30
003038                          *       ST,     STORE           IN BUFFER
003039 0C9E 68BD
003040 0C9F 30C0
003041                          *       BLCT    RCV
003042 0CA0 DE63
003045 0CA1 2714
003046                          *       WAIT
003047 0CA2 BD31
003048                          *       B       RECV1
003049 0CA3 0020
003052                          *
003053                          *
003054                          *       LOC     RECV2
003055      0285                RECV2 EQU X'0285'
003056                          *       RECV    2               TEST PARITY
003057 0CA4 F1BD
003060 0CA5 3070
003061                          *       ST,
003062 0CA6 BD30
003063                          *       BLCT    RCV
003064 0CA7 CODE
003065 0CA8 6327
003068                          *       WAIT
003069 0CA9 05BD
003070 0CAA 3100
003071                          *       B       RECV2
003074 0CAB 20F1
003075                          *
003076                          *
003077                          *       LOC     RCV
003078      0294                RCV EQU X'0294'
003079                          *       LD      =0              LOAD ZERO TO TURN OFF
003082 0CAC C600
003083                          *       OUT     2               RECEIVE
003084 0CAD BD30
003087                          *       GNB
003088 0CAE 28BD
003089 0CAF 3120
003090                          *       WAIT
003091 0CB0 BD31
003092                          *       B       RCVA
003093 0CB1 0020
003096                          *       NOP
003097 0CB2 B201
003098                          *
003099                          *
003100                          *            XMIT PROGRAM FOR CRC AND PARITY TEST
003101                          *
003102                          *
003103                          *       ORG     X'400'
003104      0CB3                CHLP2A EQU    $
003105                          *       LD      2               GET LINE CONFIGURATION
003108 0CB3 D602
003109                          *       OUT     6               OUTPUT IT
003110 0CB4 BD30
003113                          *       LD      =X'FA'          LOAD FILL CHAR.
003114 0CB5 38C6
003117                          *       OUT     4               OUTPUT SYNC. FILL CHAR.
003118 0CB6 FABD
003121 0CB7 3030
003122                          *       WAIT
003123 0CB8 BD31
003124                          *
003125                          *       LD      24              SEE IF SYNC. LINE
003126 0CB9 00D6
003129                          *       C       =X'E'           IF SYNC. LINE SEND OUT FILL CHAR (LR 4)
003130 0CBA 1886
```

```
003133  0CBB  0E10
003134  0CBC  9741
003135                        *       BET     SNC1            AND 3 SYNC. (OR FILL)
003136  0CBD  9641
003139  0CBE  2724
003140                        *       LD      =X'FA'          LOAD FILL CHAR.
003143  0CBF  C6FA
003144                        *       OUT     4
003145  0CC0  BD30
003148                        *       LD      =6
003149  0CC1  30C6
003152                        *       LOC     XMVV
003153        041F            XMVV EQU X'041F'
003154                        *       ST      63
003155  0CC2  06D7
003158                        *       LD      24              LOAD SYNC. CHAR
003159  0CC3  3FD6
003162                        *       SEND    0               SEND IT
003163  0CC4  18BD
003166  0CC5  3040
003167                        *       LD      63              DO SIX TIMES
003170  0CC6  D63F
003171                        *       DEC
003172                        *       BZT     SDVV
003173  0CC7  5A5D
003176  0CC8  2705
003177                        *       WAIT
003178  0CC9  BD31
003179                        *       B       XMVV
003180  0CCA  0020
003183                        *       LOC     SDVV
003184        0431            SDVV EQU X'0431'
003185                        *       WAIT
003186  0CCB  EEBD
003187  0CCC  3100
003188                        *       LD      =X'FF'
003191  0CCD  C6FF
003192                        *       SEND    0               XMIT START OF DATA CHAR. (FF)
003193  0CCE  BD30
003196                        *       WAIT
003197  0CCF  40BD
003198  0CD0  3100
003199                        *
003200                        *       LOC     SNC1
003201        043C            SNC1 EQU X'043C'
003202                        *       LD      23              LOAD FLAG FOR TYPE OF SEND INST.
003205  0CD1  D617
003206                        *       C       =1              IF TRUE TEST JUST CRC
003209  0CD2  8601
003210  0CD3  1097
003211                        *       BET     CPCRC
003212  0CD4  4196
003213  0CD5  4127
003216                        *       C       =X'F0'          IF TRUE TEST PARITY
003217  0CD6  1886
003220  0CD7  F010
003221  0CD8  9741
003222                        *       BET     CPPAR
003223  0CD9  9641
003226  0CDA  271E
003227                        *       LOC     SNC3            TEST BOTH CRC AND PARITY ON XMIT
003228        0450            SNC3 EQU X'0450'
003229                        *       LD,
003230  0CDB  BD30
003231                        *       SEND    3
003232  0CDC  A0BD
003235  0CDD  3058
003236                        *       BLCT    LCB1            BRANCH ON LAST CHAR.
003237  0CDE  DE63
003240  0CDF  2723
003241                        *       WAIT
003242  0CE0  BD31
003243                        *       B       SNC3            DUE NEXT CHAR.
003244  0CE1  0020
003247                        *
003248                        *       LOC     CPCRC
003249        045F            CPCRC EQU X'045F'
003250                        *       LD,     LOAD            CHAR. FROM DATA BUFFER
003251  0CE2  F1BD
003252  0CE3  30A0
003253                        *       SEND    1               XMIT CHAR. AND CALCULATE CRC
003254  0CE4  BD30
003257                        *       BLCT    LCB1            BRANCH ON LAST CHAR.
003258  0CE5  48DE
003259  0CE6  6327
003262                        *       WAIT
003263  0CE7  14BD
003264  0CE8  3100
003265                        *       B       CPCRC           DUE NEXT CHAR.
003268  0CE9  20F1
003269                        *
003270                        *       LOC     CPPAR
003271        046E            CPPAR EQU X'046E'
003272                        *       LD,     LOAD            CHAR. FROM DATA BUFFER
003273  0CEA  BD30
003274                        *       SEND    2               MIT CHAR. WITH PARITY
003275  0CEB  A0BD
003278  0CEC  3050
003279                        *       BLCT    LCB1
003280  0CED  DE63
003283  0CEE  2705
003284                        *       WAIT
003285  0CEF  BD31
003286                        *       B       CPPAR
003287  0CF0  0020
003290                        *
003291                        *       LOC     LCB1
003292        047D            LCB1 EQU X'047D'
003293                        *       WAIT
003294  0CF1  F1BD
003295  0CF2  3100
003296                        *       LD      =X'FF'          OUTPUT DUMMY CHAR. FOR SHIFT REG.
003299  0CF3  C6FF
```

```
003300                          *       SEND    0
003301  0CF4  BD30
003304                          *       BLBT    NWT             BRANCH ON LAST BLOCK
003305  0CF5  4096
003306  0CF6  6585
003307  0CF7  2026
003310                          *       B       SNC1
003311  0CF8  0220
003314                          *
003315                          *       LOC     NWT
003316        048D              NWT EQU X'048D'
003317                          *       LD      20
003318  0CF9  AFD6
003321                          *       AND     =X'FE'
003322  0CFA  14C4
003325                          *       OUT     2
003326  0CFB  FEBD
003329  0CFC  3028
003330                          *       GNB
003331  0CFD  BD31
003332                          *       WAIT
003333  0CFE  20BD
003334  0CFF  3100
003335                          *       B       SNC1
003338  0D00  20A0
003339                          *       NOP
003340                          *
003341                          *
003342                          *          RECEIVE PROGRAM FOR CHARACTER SIZE TEST
003343                          *
003344                          *
003345                          *       ORG     X'200'
003346  0D01  0100
003347        0D02              CHLP3   EQU     $
003348                          *       LOC     RCP3
003349        0200              RCP3 EQU X'0200'
003350                          *       LD      24              GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
003353  0D02  D618
003354                          *       OUT     4               OUTPUT IT
003355  0D03  BD30
003358                          *       LD      20
003359  0D04  30D6
003362                          *       OUT     2               OUTPUT LINE CONTROL
003363  0D05  14BD
003366  0D06  3028
003367                          *       RECV    0
003368  0D07  BD30
003371                          *       WAIT
003372  0D08  60BD
003373  0D09  3100
003374                          *
003375                          *       LD      24              SEE IF SYNC TO RECV.
003378  0D0A  D618
003379                          *       C       =X'E'
003382  0D0B  860E
003383  0D0C  1097
003384                          *       BET     RCCH
003385  0D0D  4196
003386  0D0E  4127
003389                          *       LOC     JHYY
003390        021B              JHYY EQU X'021B'
003391                          *       RECV    0               INPUT FIRST SYNC.
003392  0D0F  3EBD
003395  0D10  3060
003396                          *       WAIT
003397  0D11  BD31
003398                          *       LD      24
003399  0D12  00D6
003402                          *       AND     28
003403  0D13  18D4
003406                          *       ST      59
003407  0D14  1CD7
003410                          *       RECV    0               INPUT SECOND SYNC.
003411  0D15  3BBD
003414  0D16  3060
003415                          *       C       59              CHECK TO SEE IF TRUE SYNC.
003418  0D17  963B
003419  0D18  1097
003420                          *       BET     JFYY
003421  0D19  4196
003422  0D1A  4127
003425                          *       LD      20              SHUT OFF LINE
003426  0D1B  0ED6
003429                          *       AND     =X'FD'
003430  0D1C  14C4
003433                          *       OUT     2
003434  0D1D  FDBD
003437  0D1E  3028
003438                          *       LD      20              SHUT LINE BACK ON
003441  0D1F  D614
003442                          *       OUT     2
003443  0D20  BD30
003446                          *       B       JHYY
003447  0D21  2820
003450                          *       LOC     JFYY
003451        0241              JFYY EQU X'0241'
003452                          *       WAIT
003453  0D22  DABD
003454  0D23  3100
003455                          *       RECV    0               INPUT THIRD SYNC.
003456  0D24  BD30
003459                          *       LOC     RCKSY
003460        0247              RCKSY EQU X'0247'
003461                          *       WAIT
003462  0D25  60BD
003463  0D26  3100
003464                          *       RECV    0
003465  0D27  BD30
003468                          *       C       =X'1F'          SEE IF START OF DATA CHARACTER
003469  0D28  6086
003472  0D29  1F10
003473  0D2A  9741
003474                          *       BEF     RCKSY
```

```
003475  0D2B  9641
003478  0D2C  26F1
003479                    *       WAIT
003480  0D2D  BD31
003481                    *       LOC     RCCH
003482        0259        RCCH EQU X'0259'
003483                    *       LD      2                       CHECK FOR FIVE BIT CHAR.
003484  0D2E  00D6
003487                    *       BZT     NPRY
003488  0D2F  025D
003491  0D30  270F
003492                    *
003493                    *    RECV. CHAR WITH PARITY 6-7 BIT CHAR.
003494                    *
003495                    *       LOC     PRYT
003496        025E        PRYT EQU X'025E'
003497                    *       RECV    2
003498  0D31  BD30
003501                    *       ST
003502  0D32  70BD
003503  0D33  30C0
003504                    *       BLCT    SHFL
003505  0D34  DE63
003508  0D35  2714
003509                    *       WAIT
003510  0D36  BD31
003511                    *       B       PRYT
003512  0D37  0020
003515                    *
003516                    *    RECV. FIVE BIT CHAR. NO PARITY
003517                    *
003518                    *       LOC     NPRY
003519        026D        NPRY EQU X'026D'
003520                    *       RECV    0
003521  0D38  F1BD
003524  0D39  3060
003525                    *       ST
003526  0D3A  BD30
003527                    *       BLCT    SHFL
003528  0D3B  C0DE
003529  0D3C  6327
003532                    *       WAIT
003533  0D3D  05BD
003534  0D3E  3100
003535                    *       B       NPRY
003538  0D3F  20F1
003539                    *
003540                    *     TURN OFF THE LINE
003541                    *
003542                    *       LOC     SHFL
003543        027C        SHFL EQU X'027C'
003544                    *       LD      =0
003547  0D40  C600
003548                    *       OUT     2
003549  0D41  BD30
003552                    *       GNB
003553  0D42  28BD
003554  0D43  3120
003555                    *       WAIT
003556  0D44  BD31
003557                    *       JUMP    RCP3
003558  0D45  003F
003561  0D46  FF76          DC RCP3-X'028A'
003562                    *       NOP
003563                    *
003564                    *
003565                    *          TRANSMIT PROGRAM FOR CHARACTER SIZE TEST
003566                    *
003567                    *       ORG     X'400'
003568  0D47  0100
003569        0D48        CHLP3A EQU    $
003570                    *       LD      2                       GET LINE CONFIGURATION
003573  0D48  D602
003574                    *       OUT     6                       OUTPUT IT
003575  0D49  BD30
003578                    *       LD      =X'FA'                  LOAD FILL CHAR.
003579  0D4A  38C6
003582                    *       OUT     4                       OUTPUT SYNC. FILL CHAR.
003583  0D4B  FABD
003586  0D4C  3030
003587                    *       WAIT
003588  0D4D  BD31
003589                    *       LD      =X'1F'                  LOAD DEFAULT MASK TO STRIP 5 BITS
003590  0D4E  00C6
003593                    *       ST      27
003594  0D4F  1FD7
003597                    *       ST      28
003598  0D50  1BD7
003601                    *       LD      =X'FA'                  LOAD FILL CHAR.
003602  0D51  1CC6
003605                    *       OUT     4                       OUTPUT SYNC. CHAR.
003606  0D52  FABD
003609  0D53  3030
003610                    *       LD      2
003613  0D54  D602
003614                    *       BZT     XPYQ
003615  0D55  5D27
003618                    *
003619                    * SET MASK TO STRIP BITS FOR SYNC. CHAR. ON RECV. SIDE
003620                    *
003621                    *       C       =X'40'
003622  0D56  2B86
003625  0D57  4010
003626  0D58  9741
003627                    *       BEF     N61
003628  0D59  9641
003631  0D5A  2609
003632                    *       LD      =X'3F'
003635  0D5B  C63F
003636                    *       ST      28
003639  0D5C  D71C
003640                    *       SR
003641                    *       ST      27
```

```
003642  0D5D  54D7
003645                        *       B       XPYQ
003646  0D5E  1B20
003649                        *       LOC     N61
003650        042F            N61 EQU X'042F'
003651                        *       C       =X'80'
003652  0D5F  1986
003655  0D60  8010
003656  0D61  9741
003657                        *       BEF     N71
003658  0D62  9641
003661  0D63  2609
003662                        *       LD      =X'7F'
003665  0D64  C67F
003666                        *       ST      28
003669  0D65  D71C
003670                        *       SR
003671                        *       ST      27
003672  0D66  54D7
003675                        *       B       XPYQ
003676  0D67  1B20
003679                        *       LOC     N71
003680        0441            N71 EQU X'0441'
003681                        *       LD      =X'FF'
003682  0D68  07C6
003685                        *       ST      28
003686  0D69  FFD7
003689                        *       SR
003690  0D6A  1C54
003691                        *       ST      27
003694  0D6B  D71B
003695                        *       LOC     XPYQ
003696        0448            XPYQ EQU X'0448'
003697                        *       LD      24
003700  0D6C  D618
003701                        *       C       =X'E'
003704  0D6D  860E
003705  0D6E  1097
003706                        *       BET     XMNS            BRANCH ON ASYNC. LINE
003707  0D6F  4196
003708  0D70  4127
003711                        *       LD      =6
003712  0D71  1FC6
003715                        *       LOC     XMZZ
003716        0455            XMZZ EQU X'0455'
003717                        *       ST      63
003718  0D72  06D7
003721                        *       LD      24              LOAD SYNC. CHAR
003722  0D73  3FD6
003725                        *       SEND    0               SEND IT
003726  0D74  18BD
003729  0D75  3040
003730                        *       LD      63              DO SIX TIMES
003733  0D76  D63F
003734                        *       DEC
003735                        *       BZT     SDZZ
003736  0D77  5A5D
003739  0D78  2705
003740                        *       WAIT
003741  0D79  BD31
003742                        *       B       XMZZ
003743  0D7A  0020
003746                        *       LOC     SDZZ
003747        0467            SDZZ EQU X'0467'
003748                        *       WAIT
003749  0D7B  EEBD
003750  0D7C  3100
003751                        *       LD      =X'1F'
003754  0D7D  C61F
003755                        *       SEND    0               XMIT START OF DATA CHAR.
003756  0D7E  BD30
003759                        *       WAIT
003760  0D7F  40BD
003761  0D80  3100
003762                        *       LOC     XMNS
003763        0472            XMNS EQU X'0472'
003764                        *       LD      2
003767  0D81  D602
003768                        *       BZT     XNPY
003769  0D82  5D27
003772                        *
003773                        *       SEND DATA WITH PARITY
003774                        *
003775                        *       LOC     XPY
003776        0477            XPY EQU X'0477'
003777                        *       LD
003778  0D83  11BD
003779  0D84  30A0
003780                        *       AND     27
003783  0D85  D41B
003784                        *       SEND    2
003785  0D86  BD30
003788                        *       BLCT    EDXT
003789  0D87  50DE
003790  0D88  6327
003793                        *       WAIT
003794  0D89  16BD
003795  0D8A  3100
003796                        *       B       XPY
003799  0D8B  20EF
003800                        *
003801                        *     SEND DATA FIVE BIT NO PARITY
003802                        *
003803                        *       LOC     XNPY
003804        0488            XNPY EQU X'0488'
003805                        *       LD
003806  0D8C  BD30
003807                        *       AND     27
003808  0D8D  A0D4
003811                        *       SEND    0
003812  0D8E  1BBD
003815  0D8F  3040
003816                        *       BLCT    EDXT
```

```
003817  0D90  DE63
003820  0D91  2705
003821                          *       WAIT
003822  0D92  BD31
003823                          *       B       XNPY
003824  0D93  0020
003827                          *
003828                          *       XMIT DUMMY CHAR. TO SHIFT LAST DATA
003829                          *
003830                          *       LOC     EDXT
003831        0499              EDXT EQU X'0499'
003832                          *       WAIT
003833  0D94  EFBD
003834  0D95  3100
003835                          *       LD      27
003838  0D96  D61B
003839                          *       SEND    0
003840  0D97  BD30
003843                          *       BLBT    EOP
003844  0D98  4096
003845  0D99  6585
003846  0D9A  2026
003849                          *       WAIT
003850  0D9B  05BD
003851  0D9C  3100
003852                          *       B       XMNS
003855  0D9D  20C6
003856                          *
003857                          *       LOC     EOP
003858        04AC              EOP EQU X'04AC'
003859                          *       GNB
003860  0D9E  BD31
003861                          *       WAIT
003862  0D9F  20BD
003863  0DA0  3100
003864                          *       B       EDXT
003867  0DA1  20E5
003868                          *       NOP
003869                          *
003870                          *
003871                          *
003872                          *          RECEIVE PROGRAM FOR OVERRUN TEST
003873                          *
003874                          *
003875                          *       ORG     X'200'
003876  0DA2  0100
003877        0DA3              CHLP4  EQU    $
003878                          *
003879                          *       LD      =10
003882  0DA3  C60A
003883                          *       ST      28
003886  0DA4  D71C
003887                          *       LD      2
003890  0DA5  D602
003891                          *       OUT     6
003892  0DA6  BD30
003895                          *       LD      24                  GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
003896  0DA7  38D6
003899                          *       OUT     4                   OUTPUT IT
003900  0DA8  18BD
003903  0DA9  3030
003904                          *       LD      20
003907  0DAA  D614
003908                          *       OUT     2
003909  0DAB  BD30
003912                          *       RECV    0
003913  0DAC  28BD
003916  0DAD  3060
003917                          *       WAIT
003918  0DAE  BD31
003919                          *
003920                          *
003921                          *       LOC     BEG
003922        0219              BEG EQU X'0219'
003923                          *       LD      24
003924  0DAF  00D6
003927                          *       C       =X'E'
003928  0DB0  1886
003931  0DB1  0E10
003932  0DB2  9741
003933                          *       BET     RMCH
003934  0DB3  9641
003937  0DB4  2738
003938                          *       LOC     JHRR
003939        0224              JHRR EQU X'0224'
003940                          *       RECV    0                   1ST SYNC.
003941  0DB5  BD30
003944                          *       WAIT
003945  0DB6  60BD
003946  0DB7  3100
003947                          *       RECV    0                   2ND SYNC.
003948  0DB8  BD30
003951                          *       C       24                  CHECK TO SEE IF TRUE SYNC.
003952  0DB9  6096
003955  0DBA  1810
003956  0DBB  9741
003957                          *       BET     JFRR
003958  0DBC  9641
003961  0DBD  270E
003962                          *       LD      20                  SHUT OFF LINE
003965  0DBE  D614
003966                          *       AND     =X'FD'
003969  0DBF  C4FD
003970                          *       OUT     2
003971  0DC0  BD30
003974                          *       LD      20                  SHUT LINE BACK ON
003975  0DC1  28D6
003978                          *       OUT     2
003979  0DC2  14BD
003982  0DC3  3028
003983                          *       B       JHRR
003986  0DC4  20E0
003987                          *       LOC     JFRR
```

```
003988          0244            JFRR EQU X'0244'
003989                          *       WAIT
003990  0DC5    BD31
003991                          *       RECV    0                       3RD SYNC.
003992  0DC6    00BD
003995  0DC7    3060
003996                          *       LOC     NSCH
003997          024A            NSCH EQU X'024A'
003998                          *       WAIT
003999  0DC8    BD31
004000                          *       RECV    0
004001  0DC9    00BD
004004  0DCA    3060
004005                          *       C       =X'FF'                  SEE IF START DATA CHAR.
004008  0DCB    86FF
004009  0DCC    1097
004010                          *       BEF     NSCH
004011  0DCD    4196
004012  0DCE    4126
004015                          *       WAIT
004016  0DCF    F1BD
004017  0DD0    3100
004018                          *       LOC     RMCH
004019          025C            RMCH EQU X'025C'
004020                          *       RECV    0
004021  0DD1    BD30
004024                          *       ST
004025  0DD2    60BD
004026  0DD3    30C0
004027                          *       BLCT    REND
004028  0DD4    DE63
004031  0DD5    2705
004032                          *       WAIT
004033  0DD6    BD31
004034                          *       B       RMCH
004035  0DD7    0020
004038                          *
004039                          *
004040                          *       LOC     REND
004041          026B            REND EQU X'026B'
004042                          *       LD      =X'C5'
004043  0DD8    F1C6
004046                          *       OUT     2
004047  0DD9    C5BD
004050  0DDA    3028
004051                          *       GNB
004052  0DDB    BD31
004053                          *       WAIT
004054  0DDC    20BD
004055  0DDD    3100
004056                          *       B       BEG
004059  0DDE    20A1
004060                          *
004061                          *
004062                          *
004063                          *       TRANSMIT PROGRAM FOR UNDERRUN TEST
004064                          *
004065                          *
004066                          *       ORG     X'2C0'
004067          0DDE            CHLP4A EQU    $
004068                          *       LOC     TOP
004069          02C0            TOP EQU X'02C0'
004070                          *       WAIT
004071  0DDF    BD31
004072                          *       LD      24
004073  0DE0    00D6
004076                          *       C       =X'E'
004077  0DE1    1886
004080  0DE2    0E10
004081  0DE3    9741
004082                          *       BET     NSY
004083  0DE4    9641
004086  0DE5    274A
004087                          *       LD      =X'AA'                  LOAD FILL CHAR.
004090  0DE6    C6AA
004091                          *       OUT     4
004092  0DE7    BD30
004095                          *       LD      =6
004096  0DE8    30C6
004099                          *       LOC     XMRR
004100          02D5            XMRR EQU X'02D5'
004101                          *       ST      63
004102  0DE9    06D7
004105                          *       LD      24                      LOAD SYNC. CHAR
004106  0DEA    3FD6
004109                          *       SEND    0                       SEND IT
004110  0DEB    18BD
004113  0DEC    3040
004114                          *       LD      63                      DO SIX TIMES
004117  0DED    D63F
004118                          *       DEC
004119                          *       BZT     SDRR
004120  0DEE    5A5D
004123  0DEF    2705
004124                          *       WAIT
004125  0DF0    BD31
004126                          *       B       XMRR
004127  0DF1    0020
004130                          *       LOC     SDRR
004131          02E7            SDRR EQU X'02E7'
004132                          *       WAIT
004133  0DF2    EEBD
004134  0DF3    3100
004135                          *       LD      =X'FF'
004138  0DF4    C6FF
004139                          *       SEND    0
004140  0DF5    BD30
004143                          *       WAIT
004144  0DF6    40BD
004145  0DF7    3100
004146                          *       LD      =2
004149  0DF8    C602
004150                          *       ST      26
```

```
004153  0DF9  D71A
004154                        *       LOC     NZ7
004155        02F6            NZ7 EQU X'02F6'
004156                        *       LD
004157  0DFA  BD30
004158                        *       SEND    0
004159  0DFB  A0BD
004162  0DFC  3040
004163                        *       LD      26
004166  0DFD  D61A
004167                        *       DEC
004168                        *       ST      26
004169  0DFE  5AD7
004172                        *       BZT     CDLY
004173  0DFF  1A5D
004176  0E00  2705
004177                        *       WAIT
004178  0E01  BD31
004179                        *       B       NZ7
004180  0E02  0020
004183                        *
004184                        *
004185                        *       LOC     CDLY
004186        0309            CDLY EQU X'0309'
004187                        *       BS      DLAY            GO TO CHAR. DELAY
004188  0E03  EDBD
004189  0E04  3080
004192  0E05  2067
004193                        *       IN      5               SEE IF UNDERRUN BIT SET
004194  0E06  BD30
004197                        *       ST      60
004198  0E07  14D7
004201                        *       WAIT
004202  0E08  3CBD
004203  0E09  3100
004204                        *       B       NT3
004207  0E0A  2028
004208                        *
004209                        *
004210                        *       LOC     NSY
004211        0318            NSY EQU X'0318'
004212                        *       LD
004213  0E0B  BD30
004214                        *       SEND    0
004215  0E0C  A0BD
004218  0E0D  3040
004219                        *       LD      28
004222  0E0E  D61C
004223                        *       DEC
004224                        *       ST      28
004225  0E0F  5AD7
004228                        *       BZT     OVLY
004229  0E10  1C5D
004232  0E11  2705
004233                        *       WAIT
004234  0E12  BD31
004235                        *       B       NSY
004236  0E13  0020
004239                        *       LOC     OVLY
004240        032B            OVLY EQU X'032B'
004241                        *       BS      DLAY
004242  0E14  EDBD
004243  0E15  3080
004246  0E16  2045
004247                        *       SEND    0
004248  0E17  BD30
004251                        *       BS      DLAY
004252  0E18  40BD
004253  0E19  3080
004256  0E1A  203D
004257                        *       IN      5
004258  0E1B  BD30
004261                        *       ST      60
004262  0E1C  14D7
004265                        *       WAIT
004266  0E1D  3CBD
004267  0E1E  3100
004268                        *       LOC     NT3
004269        0340            NT3 EQU X'0340'
004270                        *       LD
004271  0E1F  BD30
004272                        *       SEND    0
004273  0E20  A0BD
004276  0E21  3040
004277                        *       BLCT    EXM1
004278  0E22  DE63
004281  0E23  2705
004282                        *       WAIT
004283  0E24  BD31
004284                        *       B       NT3
004285  0E25  0020
004288                        *
004289                        *
004290                        *       LOC     EXM1
004291        034F            EXM1 EQU X'034F'
004292                        *       WAIT
004293  0E26  F1BD
004294  0E27  3100
004295                        *       LD      =X'FF'
004298  0E28  C6FF
004299                        *       SEND    0
004300  0E29  BD30
004303                        *       BLBT    NSXM
004304  0E2A  4096
004305  0E2B  6585
004306  0E2C  2026
004309                        *       WAIT
004310  0E2D  06BD
004311  0E2E  3100
004312                        *       JUMP    TOP
004315  0E2F  3FFF
004316                        *
004317                        *       LOC     NSXM
```

```
004318          0363            NSXM EQU X'0363'
004319                          *       GNB
004320  0E30    5DBD
004321  0E31    3120
004322                          *       LD      20
004325  0E32    D614
004326                          *       AND     =X'FE'
004329  0E33    C4FE
004330                          *       ST      20
004333  0E34    D714
004334                          *       OUT     2
004335  0E35    BD30
004338                          *       WAIT
004339  0E36    28BD
004340  0E37    3100
004341                          *       B       NSXM
004344  0E38    20EF
004345                          *       NOP
004346                          *
004347                          *       LOC     DLAY
004348          0375            DLAY EQU X'0375'
004349                          *       LD      29
004350  0E39    01D6
004353                          *       BZF     EDLY
004354  0E3A    1D5D
004357  0E3B    2604
004358                          *       RET
004359  0E3C    DE52
004360  0E3D    6E00
004361                          *       LOC     EDLY
004362          037E            EDLY EQU X'037E'
004363                          *       ST      30
004366  0E3E    D71E
004367                          *       LD      =94
004370  0E3F    C65E
004371                          *       LOC     LESS
004372          0382            LESS EQU X'0382'
004373                          *       DEC
004374                          *       BZT     NXC
004375  0E40    5A5D
004378  0E41    2702
004379                          *       B       LESS
004382  0E42    20FA
004383                          *       LOC     NXC
004384          0388            NXC EQU X'0388'
004385                          *       LD      30
004388  0E43    D61E
004389                          *       DEC
004390                          *       BZF     EDLY
004391  0E44    5A5D
004394  0E45    26F0
004395                          *       RET
004396  0E46    DE52
004397  0E47    6E00
004398                          *       NOP
004399                          *
004400                          *
004401                          *         CHANNEL PROGRAM TO LOOP AND CHECK DSC INTO DSS
004402                          *
004403                          *
004404                          *       ORG     X'200'
004405  0E48    0100
004406          0E49            CHLP5  EQU    $
004407                          *       LOC     TOP1
004408          0200            TOP1 EQU X'0200'
004409                          *       LD      =7
004412  0E49    C607
004413                          *       ST      30
004416  0E4A    D71E
004417                          *       LD      29
004420  0E4B    D61D
004421                          *       C       =X'F0'
004424  0E4C    86F0
004425  0E4D    1097
004426                          *       BEF     TOP3
004427  0E4E    4196
004428  0E4F    4126
004431                          *       LD      =16
004432  0E50    08C6
004435                          *       ST      58
004436  0E51    10D7
004439                          *       LD      =2
004440  0E52    3AC6
004443                          *       B       TCMB1
004444  0E53    0220
004447                          *       LOC     TOP3
004448          0217            TOP3 EQU X'0217'
004449                          *       C       =X'FC'
004450  0E54    0F86
004453  0E55    FC10
004454  0E56    9741
004455                          *       BET     TOP2
004456  0E57    9641
004459  0E58    2704
004460                          *       LD      =8
004463  0E59    C608
004464                          *       ST      58
004467  0E5A    D73A
004468                          *       LOC     TOP2
004469          0224            TOP2 EQU X'0224'
004470                          *       LD      =0
004473  0E5B    C600
004474                          *       LOC     TCMB1
004475          0226            TCMB1 EQU X'0226'
004476                          *       ST      25
004479  0E5C    D719
004480                          *       C       =4
004483  0E5D    8604
004484  0E5E    1097
004485                          *       BEF     DSET
004486  0E5F    4196
004487  0E60    4126
004490                          *       LD      =X'80'
```

```
004491  0E61  08C6
004494                        *       OR      28
004495  0E62  80DA
004498                        *       ST      28
004499  0E63  1CD7
004502                        *       LD      =4
004503  0E64  1CC6
004506                        *       LOC     DSET
004507        0239            DSET EQU X'0239'
004508                        *       TLU     57
004509  0E65  04DE
004512  0E66  39BD
004513  0E67  30E0
004514                        *       OR      28
004517  0E68  DA1C
004518                        *       OUT     2
004519  0E69  BD30
004522                        *       CCH     TIME            DELAY FOR CABLE TURN AROUND
004523  0E6A  28BD
004524  0E6B  3160
004525                        *       NOP
004526                        *       NOP
004527  0E6C  0101
004528                        *       NOP
004529                        *       CCH
004530  0E6D  01BD
004531  0E6E  3160
004532                        *       CCH
004533  0E6F  BD31
004534                        *       CCH
004535  0E70  60BD
004536  0E71  3160
004537                        *       CCH
004538  0E72  BD31
004539                        *       IN      5
004540  0E73  60BD
004543  0E74  3014
004544                        *       ST
004545  0E75  BD30
004546                        *       LD      25
004547  0E76  C0D6
004550                        *       C       30
004551  0E77  1996
004554  0E78  1E10
004555  0E79  9741
004556                        *       BET     ENDTC
004557  0E7A  9641
004560  0E7B  2707
004561                        *       XOR     =X'FF'
004564  0E7C  C8FF
004565                        *       DEC
004566                        *       XOR     =X'FF'
004567  0E7D  5AC8
004570                        *       B       TCMB1
004571  0E7E  FF20
004574                        *       LOC     ENDTC
004575        026D            ENDTC EQU X'026D'
004576                        *       GNB
004577  0E7F  B9BD
004578  0E80  3120
004579                        *       WAIT
004580  0E81  BD31
004581                        *       B       TOP1
004582  0E82  0020
004585                        *       NOP
004586  0E83  8B01
004587                        *
004588                        *         CHANNEL PROGRAM FOR BROADBAND(RECV)
004589                        *    ADAPTER READY TEST & UNDERRUN OVERRUN TEST
004590                        *
004591                        *       ORG     X'200'
004592        0E84            CHLP6  EQU    $
004593                        *       LD      24              GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
004596  0E84  D618
004597                        *       OUT     4               OUTPUT IT
004598  0E85  BD30
004601                        *       LD      20
004602  0E86  30D6
004605                        *       OUT     2
004606  0E87  14BD
004609  0E88  3028
004610                        *       WAIT
004611  0E89  BD31
004612                        *       LOC     SINK
004613        020D            SINK EQU X'020D'
004614                        *       RECV    0               INPUT 1ST SYNC.
004615  0E8A  00BD
004618  0E8B  3060
004619                        *       WAIT
004620  0E8C  BD31
004621                        *       RECV    0               INPUT 2ND SYNC.
004622  0E8D  00BD
004625  0E8E  3060
004626                        *       C       24
004629  0E8F  9618
004630  0E90  1097
004631                        *       BET     NSYK
004632  0E91  4196
004633  0E92  4127
004636                        *       LD      20
004637  0E93  0ED6
004640                        *       AND     =X'FD'
004641  0E94  14C4
004644                        *       OUT     2
004645  0E95  FDBD
004648  0E96  3028
004649                        *       LD      20
004652  0E97  D614
004653                        *       OUT     2
004654  0E98  BD30
004657                        *       B       SINK
004658  0E99  2820
004661                        *       LOC     NSYK
```

```
004662          022D            NSYK EQU X'022D'
004663                          *       WAIT
004664  0E9A    E0BD
004665  0E9B    3100
004666                          *       RECV    0
004667  0E9C    BD30
004670                          *       WAIT
004671  0E9D    60BD
004672  0E9E    3100
004673                          *       LD      30
004676  0E9F    D61E
004677                          *       BZF     UDOR
004678  0EA0    5D26
004681                          *       BARF    RERR
004682  0EA1    3B20
004685                          *       LD      20
004686  0EA2    0ED6
004689                          *       OUT     2
004690  0EA3    14BD
004693  0EA4    3028
004694                          *       LOC     BUFE
004695          0242            BUFE EQU X'0242'
004696                          *       RECV    0
004697  0EA5    BD30
004700                          *       ST
004701  0EA6    60BD
004702  0EA7    30C0
004703                          *       BARF    BUFE
004706  0EA8    20F8
004707                          *       BART    CHRL
004708                          *       LOC     RERR
004709          024B            RERR EQU X'024B'
004710                          *       LD      =X'FF'
004711  0EA9    01C6
004714                          *       ST      61
004715  0EAA    FFD7
004718                          *       LOC     CHRL
004719          024F            CHRL EQU X'024F'
004720                          *       LD      =X'33'
004721  0EAB    3DC6
004724                          *       ST      62
004725  0EAC    33D7
004728                          *       LD      =127
004729  0EAD    3EC6
004732                          *       ST      59
004733  0EAE    7FD7
004736                          *       LOC     BEPY3
004737          0257            BEPY3 EQU X'0257'
004738                          *       WAIT
004739  0EAF    3BBD
004740  0EB0    3100
004741                          *       RECV    0
004742  0EB1    BD30
004745                          *       ST
004746  0EB2    60BD
004747  0EB3    30C0
004748                          *       LD      59
004751  0EB4    D63B
004752                          *       DEC
004753                          *       BZT     STP3
004754  0EB5    5A5D
004757  0EB6    2758
004758                          *       ST      59
004761  0EB7    D73B
004762                          *       IN      5               CHECK BUFFER EMPTY BIT
004763  0EB8    BD30
004766                          *       AND     =X'04'
004767  0EB9    14C4
004770                          *       BZF     BEPY3
004771  0EBA    045D
004774  0EBB    26E7
004775                          *       LD      =0
004778  0EBC    C600
004779                          *       ST      62
004782  0EBD    D73E
004783                          *       B       STP3
004786  0EBE    2048
004787                          *       LOC     UDOR
004788          0276            UDOR EQU X'0276'
004789                          *       C       =X'F'
004792  0EBF    860F
004793  0EC0    1097
004794                          *       BET     UND
004795  0EC1    4196
004796  0EC2    4127
004799                          *       RECV    0
004800  0EC3    0BBD
004803  0EC4    3060
004804                          *       ST
004805  0EC5    BD30
004806                          *       WAIT
004807  0EC6    C0BD
004808  0EC7    3100
004809                          *       B       STP3
004812  0EC8    2034
004813                          *
004814                          *
004815                          *       LOC     UND
004816          028A            UND EQU X'028A'
004817                          *       LD      =64
004820  0EC9    C640
004821                          *       LOC     FIFE
004822          028C            FIFE EQU X'028C'
004823                          *       ST      59
004826  0ECA    D73B
004827                          *       BARF    FIF3
004830  0ECB    200C
004831                          *       RECV    0
004832  0ECC    BD30
004835                          *       ST
004836  0ECD    60BD
004837  0ECE    30C0
004838                          *       LD      59
```

```
004841  0ECF  D63B
004842                        *       DEC
004843                        *       BZT     B1
004844  0ED0  5A5D
004847  0ED1  2702
004848                        *       LOC     FIF3
004849        029C            FIF3 EQU X'029C'
004850                        *       B       FIFE
004853  0ED2  20EE
004854                        *       LOC     B1
004855        029E            B1 EQU X'029E'
004856                        *       RECV    0
004857  0ED3  BD30
004860                        *       ST
004861  0ED4  60BD
004862  0ED5  30C0
004863                        *       IN      5
004864  0ED6  BD30
004867                        *       ST      60
004868  0ED7  14D7
004871                        *       LD      20
004872  0ED8  3CD6
004875                        *       OR      =1
004876  0ED9  14CA
004879                        *       ST      20
004880  0EDA  01D7
004883                        *       OUT     2               TURN ON XMIT
004884  0EDB  14BD
004887  0EDC  3028
004888                        *       WAIT
004889  0EDD  BD31
004890                        *       RECV    0
004891  0EDE  00BD
004894  0EDF  3060
004895                        *       ST
004896  0EE0  BD30
004897                        *       WAIT
004898  0EE1  C0BD
004899  0EE2  3100
004900                        *       LOC     STP3
004901        02BE            STP3 EQU X'02BE'
004902                        *       LD      =0
004905  0EE3  C600
004906                        *       OUT     2
004907  0EE4  BD30
004910                        *       GNB
004911  0EE5  28BD
004912  0EE6  3120
004913                        *       WAIT
004914  0EE7  BD31
004915                        *       NOP
004916  0EE8  0001
004917                        *
004918                        *         CHANNEL PROGRAM FOR BROADBAND LOOP(XMIT)
004919                        *      ADAPTER READY & UNDERRUN OVERRUN TEST
004920                        *
004921                        *
004922                        *       ORG     X'400'
004923        0EE9            CHLP6A EQU    $
004924                        *       LD      2               GET LINE CONFIGURATION
004927  0EE9  D602
004928                        *       OUT     6               OUTPUT IT
004929  0EEA  BD30
004932                        *       LD      24
004933  0EEB  38D6
004936                        *       OUT     4
004937  0EEC  18BD
004940  0EED  3030
004941                        *       WAIT
004942  0EEE  BD31
004943                        *       IN      5               CHECK BUFFER EMPTY BIT
004944  0EEF  00BD
004947  0EF0  3014
004948                        *       AND     =X'04'
004951  0EF1  C404
004952                        *       BZT     BEPY
004953  0EF2  5D27
004956                        *       LD      =X'11'
004957  0EF3  04C6
004960                        *       ST      62
004961  0EF4  11D7
004964                        *       LOC     BEPY
004965        0419            BEPY EQU X'0419'
004966                        *       LD      =3
004967  0EF5  3EC6
004970                        *       LOC     XMBB
004971        041B            XMBB EQU X'041B'
004972                        *       ST      59
004973  0EF6  03D7
004976                        *       LD      24              LOAD SYNC. CHAR
004977  0EF7  3BD6
004980                        *       SEND    0               SEND IT
004981  0EF8  18BD
004984  0EF9  3040
004985                        *       LD      59              DO SIX TIMES
004988  0EFA  D63B
004989                        *       DEC
004990                        *       BZT     SDBB
004991  0EFB  5A5D
004994  0EFC  2705
004995                        *       WAIT
004996  0EFD  BD31
004997                        *       B       XMBB
004998  0EFE  0020
005001                        *       LOC     SDBB
005002        042D            SDBB EQU X'042D'
005003                        *       WAIT
005004  0EFF  EEBD
005005  0F00  3100
005006                        *       LD      30
005009  0F01  D61E
005010                        *       BZF     ORUD
005011  0F02  5D26
```

```
005014                          *       LD      20
005015  0F03  48D6
005018                          *       AND     =X'FD'
005019  0F04  14C4
005022                          *       OUT     2
005023  0F05  FDBD
005026  0F06  3028
005027                          *       LD      =58
005030  0F07  C63A
005031                          *       LOC     PGCX
005032          043E            PGCX EQU X'043E'
005033                          *       ST      59
005036  0F08  D73B
005037                          *       LD
005038  0F09  BD30
005039                          *       SEND    0
005040  0F0A  A0BD
005043  0F0B  3040
005044                          *       LD      59
005047  0F0C  D63B
005048                          *       DEC
005049                          *       BZT     ALD
005050  0F0D  5A5D
005053  0F0E  2702
005054                          *       B       PGCX
005057  0F0F  20F0
005058                          *       LOC     ALD
005059          044E            ALD EQU X'044E'
005060                          *       IN      5               CHECK BUFFER EMPTY BIT
005061  0F10  BD30
005064                          *       AND     =X'04'
005065  0F11  14C4
005068                          *       BZF     BEPY1
005069  0F12  045D
005072  0F13  2604
005073                          *       LD      =X'22'
005076  0F14  C622
005077                          *       ST      62
005080  0F15  D73E
005081                          *       LOC     BEPY1
005082          045A            BEPY1 EQU X'045A'
005083                          *       LD      =70
005086  0F16  C646
005087                          *       LOC     AL76
005088          045C            AL76 EQU X'045C'
005089                          *       ST      59
005092  0F17  D73B
005093                          *       LD
005094  0F18  BD30
005095                          *       SEND    0
005096  0F19  A0BD
005099  0F1A  3040
005100                          *       LD      59
005103  0F1B  D63B
005104                          *       DEC
005105                          *       BZT     AL77
005106  0F1C  5A5D
005109  0F1D  2706
005110                          *       BART    AL76
005111                          *       WAIT
005112  0F1E  01BD
005113  0F1F  3100
005114                          *       B       TRCV
005117  0F20  2004
005118                          *       LOC     AL77
005119          0470            AL77 EQU X'0470'
005120                          *       LD      =X'FF'
005123  0F21  C6FF
005124                          *       ST      60
005127  0F22  D73C
005128                          *       LOC     TRCV
005129          0474            TRCV EQU X'0474'
005130                          *       LD      20
005133  0F23  D614
005134                          *       AND     =X'FC'          TURN OFF XMIT AND RECV.
005137  0F24  C4FC
005138                          *       OUT     2
005139  0F25  BD30
005142                          *       B       ALD1
005143  0F26  2820
005146                          *       LOC     ORUD
005147          047D            ORUD EQU X'047D'
005148                          *       C       =X'F'
005149  0F27  4386
005152  0F28  0F10
005153  0F29  9741
005154                          *       BEF     OVR
005155  0F2A  9641
005158  0F2B  261A
005159                          *       LD      20
005162  0F2C  D614
005163                          *       AND     =X'FE'
005166  0F2D  C4FE
005167                          *       ST      20
005170  0F2E  D714
005171                          *       OUT     2               TURN OFF XMIT
005172  0F2F  BD30
005175                          *       WAIT
005176  0F30  28BD
005177  0F31  3100
005178                          *       LD
005179  0F32  BD30
005180                          *       SEND    0
005181  0F33  A0BD
005184  0F34  3040
005185                          *       GNB
005186  0F35  BD31
005187                          *       WAIT
005188  0F36  20BD
005189  0F37  3100
005190                          *       B       ALD1
005193  0F38  2020
005194                          *
```

```
005195                          *
005196                          *       LOC     OVR
005197          04A0            OVR EQU X'04A0'
005198                          *       LD
005199  0F39    BD30
005200                          *       SEND    0
005201  0F3A    A0BD
005204  0F3B    3040
005205                          *       BART    OVR
005206                          *       LD      =74
005207  0F3C    01C6
005210                          *       LOC     CNT
005211          04A9            CNT EQU X'04A9'
005212                          *       ST      30
005213  0F3D    4AD7
005216                          *       BARF    ROTR
005217  0F3E    1E20
005220                          *       LD
005221  0F3F    0CBD
005222  0F40    30A0
005223                          *       SEND    0
005224  0F41    BD30
005227                          *       LD      30
005228  0F42    40D6
005231                          *       BZT     OVCK
005232  0F43    1E5D
005235  0F44    2703
005236                          *       DEC
005237                          *       LOC     ROTR
005238          04B9            ROTR EQU X'04B9'
005239                          *       B       CNT
005240  0F45    5A20
005243                          *       LOC     OVCK
005244          04BB            OVCK EQU X'04BB'
005245                          *       IN      5
005246  0F46    EEBD
005249  0F47    3014
005250                          *       ST      60
005253  0F48    D73C
005254                          *
005255                          *
005256                          *       LOC     ALD1
005257          04C0            ALD1 EQU X'04C0'
005258                          *       LD      20
005261  0F49    D614
005262                          *       AND     =X'FE'
005265  0F4A    C4FE
005266                          *       ST      20
005269  0F4B    D714
005270                          *       OUT     2
005271  0F4C    BD30
005274                          *       GNB
005275  0F4D    28BD
005276  0F4E    3120
005277                          *       LOC     TAMPA
005278          04CC            TAMPA EQU X'04CC'
005279                          *       WAIT
005280  0F4F    BD31
005281                          *       B       TAMPA
005282  0F50    0020
005285                          *       NOP
005286  0F51    FB01
005287                          *
005288                          *
005289                          *         CHANNEL PROGRAM FOR STOP I/O(BROADBAND RECV.)
005290                          *
005291                          *
005292                          *       ORG     X'200'
005293          0F52            CHLP7   EQU     $
005294                          *       LD      24              GET LINE SPEED (ASYNC) OR SYNC CHAR (SYNC)
005297  0F52    D618
005298                          *       OUT     4               OUTPUT IT
005299  0F53    BD30
005302                          *       LD      20
005303  0F54    30D6
005306                          *       OUT     2
005307  0F55    14BD
005310  0F56    3028
005311                          *       WAIT
005312  0F57    BD31
005313                          *       LOC     SINK1
005314          020D            SINK1 EQU X'020D'
005315                          *       RECV    0               INPUT 1ST SYNC.
005316  0F58    00BD
005319  0F59    3060
005320                          *       WAIT
005321  0F5A    BD31
005322                          *       RECV    0               INPUT 2ND SYNC.
005323  0F5B    00BD
005326  0F5C    3060
005327                          *       C       24
005330  0F5D    9618
005331  0F5E    1097
005332                          *       BET     NSYK1
005333  0F5F    4196
005334  0F60    4127
005337                          *       LD      20
005338  0F61    0ED6
005341                          *       AND     =X'FD'
005342  0F62    14C4
005345                          *       OUT     2
005346  0F63    FDBD
005349  0F64    3028
005350                          *       LD      20
005353  0F65    D614
005354                          *       OUT     2
005355  0F66    BD30
005358                          *       B       SINK1
005359  0F67    2820
005362                          *       LOC     NSYK1
005363          022D            NSYK1 EQU X'022D'
005364                          *       WAIT
005365  0F68    E0BD
```

```
005366  0F69  3100
005367                    *       RECV    0
005368  0F6A  BD30
005371                    *       LOC     LKB1
005372        0233        LKB1 EQU X'0233'
005373                    *       WAIT
005374  0F6B  60BD
005375  0F6C  3100
005376                    *       RECV    0
005377  0F6D  BD30
005380                    *       C       =X'DB'
005381  0F6E  6086
005384  0F6F  DB10
005385  0F70  9741
005386                    *       BEF     LKB1
005387  0F71  9641
005390  0F72  26F1
005391                    *       WAIT
005392  0F73  BD31
005393                    *       LOC     T25A
005394        0245        T25A EQU X'0245'
005395                    *       RECV    0
005396  0F74  00BD
005399  0F75  3060
005400                    *       ST
005401  0F76  BD30
005402                    *       BLCT    KER
005403  0F77  C0DE
005404  0F78  6327
005407                    *       BART    T25A
005408  0F79  0601
005409                    *       WAIT
005410  0F7A  BD31
005411                    *       B       T25A
005412  0F7B  0020
005415                    *       LOC     KER
005416        0255        KER EQU X'0255'
005417                    *       LD      =0
005418  0F7C  F0C6
005421                    *       OUT     2
005422  0F7D  00BD
005425  0F7E  3028
005426                    *       GNB
005427  0F7F  BD31
005428                    *       WAIT
005429  0F80  20BD
005430  0F81  3100
005431                    *       B       SINK1
005434  0F82  20AB
005435                    *       NOP
005436                    *
005437                    *
005438                    *
005439                    *
005440                    *         CHANNEL PROGRAM FOR STOP I/O (BROADBAND XMIT)
005441                    *
005442                    *
005443                    *       ORG     X'400'
005444  0F83  0100
005445        0F84        CHLP7A EQU    $
005446                    *       LD      2                 GET LINE CONFIGURATION
005449  0F84  D602
005450                    *       OUT     6                 OUTPUT IT
005451  0F85  BD30
005454                    *       LD      =X'AA'
005455  0F86  38C6
005458                    *       OUT     4
005459  0F87  AABD
005462  0F88  3030
005463                    *       WAIT
005464  0F89  BD31
005465                    *       LD      =6
005466  0F8A  00C6
005469                    *       LOC     XMWW
005470        040F        XMWW EQU X'040F'
005471                    *       ST      63
005472  0F8B  06D7
005475                    *       LD      24                LOAD SYNC. CHAR
005476  0F8C  3FD6
005479                    *       SEND    0                 SEND IT
005480  0F8D  18BD
005483  0F8E  3040
005484                    *       LD      63                DO SIX TIMES
005487  0F8F  D63F
005488                    *       DEC
005489                    *       BZT     SDWW
005490  0F90  5A5D
005493  0F91  2705
005494                    *       WAIT
005495  0F92  BD31
005496                    *       B       XMWW
005497  0F93  0020
005500                    *       LOC     SDWW
005501        0421        SDWW EQU X'0421'
005502                    *       WAIT
005503  0F94  EEBD
005504  0F95  3100
005505                    *       LD      =X'DB'
005508  0F96  C6DB
005509                    *       SEND    0
005510  0F97  BD30
005513                    *       WAIT
005514  0F98  40BD
005515  0F99  3100
005516                    *       LOC     T25B
005517        042C        T25B EQU X'042C'
005518                    *       LD
005519  0F9A  BD30
005520                    *       SEND    0
005521  0F9B  A0BD
005524  0F9C  3040
005525                    *       BART    T25B
005526                    *       WAIT
```

```
005527  0F9D  01BD
005528  0F9E  3100
005529                    *       LOC     HANG
005530        0436        HANG EQU X'0436'
005531                    *       LD
005532  0F9F  BD30
005533                    *       SEND    0
005534  0FA0  A0BD
005537  0FA1  3040
005538                    *       BLCT    ERR
005539  0FA2  DE63
005542  0FA3  2705
005543                    *       WAIT
005544  0FA4  BD31
005545                    *       B       HANG
005546  0FA5  0020
005549                    *       LOC     ERR
005550        0445        ERR EQU X'0445'
005551                    *       WAIT
005552  0FA6  F1BD
005553  0FA7  3100
005554                    *       LD      20
005557  0FA8  D614
005558                    *       AND     =X'FE'
005561  0FA9  C4FE
005562                    *       OUT     2
005563  0FAA  BD30
005566                    *       GNB
005567  0FAB  28BD
005568  0FAC  3120
005569                    *       WAIT
005570  0FAD  BD31
005571                    *       B       T25B
005572  0FAE  0020
005575                    *       NOP
005576  0FAF  D501
005577                    *
005578                    *
005579                    *          CHANNEL PROGRAM FOR CHARACTER SIZE AND
005580                    *           CRC, PARITY TESTS (BROADBAND RECV.)
005581                    *
005582                    *
005583                    *       ORG     X'200'
005584        0FB0        CHLP8  EQU    $
005585                    *       LD      24
005588  0FB0  D618
005589                    *       OUT     4               OUTPUT SYNC. CHARACTER
005590  0FB1  BD30
005593                    *       LD      20
005594  0FB2  30D6
005597                    *       OUT     2               OUTPUT LINE CONTROL
005598  0FB3  14BD
005601  0FB4  3028
005602                    *       WAIT
005603  0FB5  BD31
005604                    *
005605                    *       LOC     SYCH            GET IN RECV. IN SYNC.
005606        020D        SYCH EQU X'020D'
005607                    *       RECV    0
005608  0FB6  00BD
005611  0FB7  3060
005612                    *       BART    SCY2
005613                    *       WAIT
005614  0FB8  01BD
005615  0FB9  3100
005616                    *       LOC     SCY2
005617        0214        SCY2 EQU X'0214'
005618                    *       RECV    0
005619  0FBA  BD30
005622                    *       C       28
005623  0FBB  6096
005626  0FBC  1C10
005627  0FBD  9741
005628                    *       BET     SYC
005629  0FBE  9641
005632  0FBF  2705
005633                    *       SFS
005634  0FC0  BD31
005635                    *       B       SYCH
005636  0FC1  4020
005639                    *       LOC     SYC
005640        0225        SYC EQU X'0225'
005641                    *       BART    SCY1
005642  0FC2  E801
005643                    *       WAIT
005644  0FC3  BD31
005645                    *       LOC     SCY1
005646        0229        SCY1 EQU X'0229'
005647                    *       RECV    0
005648  0FC4  00BD
005651  0FC5  3060
005652                    *       C       =X'19'          CHECK FOR START OF MESSAGE SHAR.
005655  0FC6  8619
005656  0FC7  1097
005657                    *       BEF     SYC
005658  0FC8  4196
005659  0FC9  4126
005662                    *
005663                    *       LD      24
005664  0FCA  F0D6
005667                    *       C       =X'32'
005668  0FCB  1886
005671  0FCC  3210
005672  0FCD  9741
005673                    *       BET     CPRT            IF EQUAL GO TO CRC AND PARITY ROUTINES
005674  0FCE  9641
005677  0FCF  2760
005678                    *
005679                    *       LD      2               START OF CHAR. SIZE TEST
005682  0FD0  D602
005683                    *       BZT     FVBC            BRANCH TO FIVE BIT CHAR. OUTPUT LOOP
005684  0FD1  5D27
005687                    *       C       =X'40'
```

```
005688  0FD2  3686
005691  0FD3  4010
005692  0FD4  9741
005693                        *       BEF     NFBA
005694  0FD5  9641
005697  0FD6  2619
005698                        *       LOC     SXBC
005699        024E            SXBC EQU X'024E'
005700                        *       BART    SXB1
005701                        *       WAIT
005702  0FD7  01BD
005703  0FD8  3100
005704                        *       LOC     SXB1
005705        0252            SXB1 EQU X'0252'
005706                        *       RECV    2
005707  0FD9  BD30
005710                        *       C       30
005711  0FDA  7096
005714  0FDB  1E10
005715  0FDC  9741
005716                        *       BET     SXBC
005717  0FDD  9641
005720  0FDE  27F0
005721                        *       ST
005722  0FDF  BD30
005723                        *       BLCT    EXT
005724  0FE0  CODE
005725  0FE1  6327
005728                        *       B       SXBC
005729  0FE2  7F20
005732                        *
005733                        *       LOC     NFBA
005734        0267            NFBA EQU X'0267'
005735                        *       BART    NFVB
005736  0FE3  E701
005737                        *       WAIT
005738  0FE4  BD31
005739                        *       LOC     NFVB
005740        026B            NFVB EQU X'026B'
005741                        *       RECV    2
005742  0FE5  00BD
005745  0FE6  3070
005746                        *       ST
005747  0FE7  BD30
005748                        *       BLCT    EXT
005749  0FE8  CODE
005750  0FE9  6327
005753                        *       B       NFBA
005754  0FEA  6F20
005757                        *       LOC     FVBY
005758        0277            FVBY EQU X'0277'
005759                        *       BART    FVBC
005760  0FEB  F001
005761                        *       WAIT
005762  0FEC  BD31
005763                        *       LOC     FVBC
005764        027B            FVBC EQU X'027B'
005765                        *       RECV    0
005766  0FED  00BD
005769  0FEE  3060
005770                        *       C       30
005773  0FEF  961E
005774  0FF0  1097
005775                        *       BET     FVBY
005776  0FF1  4196
005777  0FF2  4127
005780                        *       ST
005781  0FF3  F0BD
005782  0FF4  30C0
005783                        *       C       =X'1E'
005786  0FF5  861E
005787  0FF6  1097
005788                        *       BET     TKF
005789  0FF7  4196
005790  0FF8  4127
005793                        *       LOC     TKFN
005794        0293            TKFN EQU X'0293'
005795                        *       BLCT    EXT
005796  0FF9  06DE
005797  0FFA  6327
005800                        *       B       FVBY
005801  0FFB  4D20
005804                        *       LOC     TKF
005805        0299            TKF EQU X'0299'
005806                        *       LD      30
005807  0FFC  DED6
005810                        *       ST
005811  0FFD  1EBD
005812  0FFE  30C0
005813                        *       B       TKFN
005816  0FFF  20F3
005817                        *       LOC     CPRT
005818        02A0            CPRT EQU X'02A0'
005819                        *       LD      23
005822  1000  D617
005823                        *       C       =1
005826  1001  8601
005827  1002  1097
005828                        *       BET     RE1A
005829  1003  4196
005830  1004  4127
005833                        *       C       =X'F0'
005834  1005  1986
005837  1006  F010
005838  1007  9741
005839                        *       BET     RE2A
005840  1008  9641
005843  1009  2720
005844                        *       LOC     RE3A
005845        02B4            RE3A EQU X'02B4'
005846                        *       BART    RE3
005847                        *       WAIT
005848  100A  01BD
```

```
005849  100B  3100
005850                        *       LOC     RE3
005851        02B8            RE3 EQU X'02B8'
005852                        *       RECV    3
005853  100C  BD30
005856                        *       ST
005857  100D  78BD
005858  100E  30C0
005859                        *       BLCT    EXT
005860  100F  DE63
005863  1010  2722
005864                        *       B       RE3A
005867  1011  20F0
005868                        *       LOC     RE1A
005869        02C4            RE1A EQU X'02C4'
005870                        *       BART    RE1
005871                        *       WAIT
005872  1012  01BD
005873  1013  3100
005874                        *
005875                        *       LOC     RE1
005876        02C8            RE1 EQU X'02C8'
005877                        *       RECV    1
005878  1014  BD30
005881                        *       ST
005882  1015  68BD
005883  1016  30C0
005884                        *       BLCT    EXT
005885  1017  DE63
005888  1018  2712
005889                        *       B       RE1A
005892  1019  20F0
005893                        *       LOC     RE2A
005894        02D4            RE2A EQU X'02D4'
005895                        *       BART    RE2
005896                        *       WAIT
005897  101A  01BD
005898  101B  3100
005899                        *
005900                        *       LOC     RE2
005901        02D8            RE2 EQU X'02D8'
005902                        *       RECV    2
005903  101C  BD30
005906                        *       ST
005907  101D  70BD
005908  101E  30C0
005909                        *       BLCT    EXT
005910  101F  DE63
005913  1020  2702
005914                        *       B       RE2A
005917  1021  20F0
005918                        *
005919                        *       LOC     EXT
005920        02E4            EXT EQU X'02E4'
005921                        *       LD      =0
005924  1022  C600
005925                        *       OUT     2
005926  1023  BD30
005929                        *       GNB
005930  1024  28BD
005931  1025  3120
005932                        *       WAIT
005933  1026  BD31
005934                        *       NOP
005935  1027  0001
005936                        *
005937                        *
005938                        *          CHANNEL PROGRAM FOR CHARACTER SIZE AND
005939                        *           CRC,PARITY TESTS (BROADBAND XMIT)
005940                        *
005941                        *
005942                        *       ORG     X'400'
005943        1028            CHLP8A EQU    $
005944                        *       LD      2                       GET LINE CONFIGURATION
005947  1028  D602
005948                        *       OUT     6                       OUTPUT IT
005949  1029  BD30
005952                        *       LD      =X'FF'
005953  102A  38C6
005956                        *       OUT     4
005957  102B  FFBD
005960  102C  3030
005961                        *       ST      28
005964  102D  D71C
005965                        *       WAIT
005966  102E  BD31
005967                        *       LD      24
005968  102F  00D6
005971                        *       C       =X'32'
005972  1030  1886
005975  1031  3210
005976  1032  9741
005977                        *       BET     Z48
005978  1033  9641
005981  1034  273C
005982                        *       LD      =X'1F'
005985  1035  C61F
005986                        *       ST      27
005989  1036  D71B
005990                        *       ST      28
005993  1037  D71C
005994                        *       LD      2
005997  1038  D602
005998                        *       BZT     Z47
005999  1039  5D27
006002                        *       C       =X'40'
006003  103A  2B86
006006  103B  4010
006007  103C  9741
006008                        *       BEF     N611
006009  103D  9641
006012  103E  2609
006013                        *       LD      =X'3F'
```

```
006016  103F  C63F
006017                          *       ST      28
006020  1040  D71C
006021                          *       SR
006022                          *       ST      27
006023  1041  54D7
006026                          *       B       Z47
006027  1042  1B20
006030                          *       LOC     N611
006031        0437              N611 EQU X'0437'
006032                          *       C       =X'80'
006033  1043  1986
006036  1044  8010
006037  1045  9741
006038                          *       BEF     N711
006039  1046  9641
006042  1047  2609
006043                          *       LD      =X'7F'
006046  1048  C67F
006047                          *       ST      28
006050  1049  D71C
006051                          *       SR
006052                          *       ST      27
006053  104A  54D7
006056                          *       B       Z47
006057  104B  1B20
006060                          *       LOC     N711
006061        0449              N711 EQU X'0449'
006062                          *       LD      =X'FF'
006063  104C  07C6
006066                          *       ST      28
006067  104D  FFD7
006070                          *       SR
006071  104E  1C54
006072                          *       ST      27
006075  104F  D71B
006076                          *
006077                          *       LOC     Z47
006078        0450              Z47 EQU X'0450'
006079                          *       LD      28
006082  1050  D61C
006083                          *       AND     =X'FF'
006086  1051  C4FF
006087                          *       ST      30
006090  1052  D71E
006091                          *       LOC     Z48
006092        0456              Z48 EQU X'0456'
006093                          *       LD      28
006096  1053  D61C
006097                          *       AND     24
006100  1054  D418
006101                          *       ST      28
006104  1055  D71C
006105                          *
006106                          *       LD      =6
006109  1056  C606
006110                          *       LOC     Z2Z
006111        045E              Z2Z EQU X'045E'
006112                          *       ST      63
006115  1057  D73E
006116                          *       LD      24
006119  1058  D618
006120                          *       SEND    0
006121  1059  BD30
006124                          *       LD      63
006125  105A  40D6
006128                          *       DEC
006129  105B  3F5A
006130                          *       BZT     Z3Z
006131  105C  5D27
006134                          *       BART    Z2Z
006135  105D  0101
006136                          *       LOC     Z3Z
006137        046C              Z3Z EQU X'046C'
006138                          *       LD      =X'19'
006141  105E  C619
006142                          *       SEND    0
006143  105F  BD30
006146                          *
006147                          *       LD      24
006148  1060  40D6
006151                          *       C       =X'32'
006152  1061  1886
006155  1062  3210
006156  1063  9741
006157                          *       BET     CPRTX
006158  1064  9641
006161  1065  2729
006162                          *       LD      2
006165  1066  D602
006166                          *       BZT     ZXZ1
006167  1067  5D27
006170                          *       LOC     XPYZ
006171        0481              XPYZ EQU X'0481'
006172                          *       LD
006173  1068  12BD
006174  1069  30A0
006175                          *       AND     27
006178  106A  D41B
006179                          *       SEND    2
006180  106B  BD30
006183                          *       BLCT    ZEXT
006184  106C  50DE
006185  106D  6327
006188                          *       BART    XPYZ
006189  106E  5C01
006190                          *       WAIT
006191  106F  BD31
006192                          *       B       XPYZ
006193  1070  0020
006196                          *
006197                          *       LOC     ZXZ1
006198        0493              ZXZ1 EQU X'0493'
```

```
006199                          *       LD
006200  1071  EEBD
006201  1072  30A0
006202                          *       AND     27
006205  1073  D41B
006206                          *       SEND    0
006207  1074  BD30
006210                          *       BLCT    ZEXT
006211  1075  40DE
006212  1076  6327
006215                          *       BART    ZXZ1
006216  1077  4A01
006217                          *       WAIT
006218  1078  BD31
006219                          *       B       ZXZ1
006220  1079  0020
006223                          *
006224                          *       LOC     CPRTX
006225        04A5              CPRTX EQU X'04A5'
006226                          *       LD      23
006227  107A  EED6
006230                          *       C       =1
006231  107B  1786
006234  107C  0110
006235  107D  9741
006236                          *       BET     BBCR
006237  107E  9641
006240  107F  2719
006241                          *       C       =X'F0'
006244  1080  86F0
006245  1081  1097
006246                          *       BET     BBPR
006247  1082  4196
006248  1083  4127
006251                          *
006252                          *       LOC     BBPCR
006253        04B9              BBPCR EQU X'04B9'
006254                          *       LD
006255  1084  20BD
006256  1085  30A0
006257                          *       SEND    3
006258  1086  BD30
006261                          *       BLCT    ZEXT
006262  1087  58DE
006263  1088  6327
006266                          *       BART    BBPCR
006267  1089  2601
006268                          *       WAIT
006269  108A  BD31
006270                          *       B       BBPCR
006271  108B  0020
006274                          *
006275                          *       LOC     BBCR
006276        04C9              BBCR EQU X'04C9'
006277                          *       LD
006278  108C  F0BD
006279  108D  30A0
006280                          *       SEND    1
006281  108E  BD30
006284                          *       BLCT    ZEXT
006285  108F  48DE
006286  1090  6327
006289                          *       BART    BBCR
006290  1091  1601
006291                          *       WAIT
006292  1092  BD31
006293                          *       B       BBCR
006294  1093  0020
006297                          *
006298                          *       LOC     BBPR
006299        04D9              BBPR EQU X'04D9'
006300                          *       LD
006301  1094  F0BD
006302  1095  30A0
006303                          *       SEND    2
006304  1096  BD30
006307                          *       BLCT    ZEXT
006308  1097  50DE
006309  1098  6327
006312                          *       BART    BBPR
006313  1099  0601
006314                          *       WAIT
006315  109A  BD31
006316                          *       B       BBPR
006317  109B  0020
006320                          *
006321                          *       LOC     ZEXT
006322        04E9              ZEXT EQU X'04E9'
006323                          *       LD      =0
006324  109C  F0C6
006327                          *       SEND    0
006328  109D  00BD
006331  109E  3040
006332                          *       SEND    0
006333  109F  BD30
006336                          *       LD      20
006337  10A0  40D6
006340                          *       AND     =X'FE'
006341  10A1  14C4
006344                          *       OUT     2
006345  10A2  FEBD
006348  10A3  3028
006349                          *       GNB
006350  10A4  BD31
006351                          *       WAIT
006352  10A5  20BD
006353  10A6  3100
006354                          *       NOP
006355                          *
006356                          *         CHANNEL PROGRAM FOR MARK & SPACE (REC)
006357                          *
006358                          *
006359                          *       ORG     X'200'
```

```
006360  10A7  0100
006361        10A8              CHLP9  EQU    $
006362                          *      LD     2                GET LINE CONFIG.
006365  10A8  D602
006366                          *      OUT    6                OUTPUT IT
006367  10A9  BD30
006370                          *      LD     =X'D'            LOAD LINE SPEED
006371  10AA  38C6
006374                          *      OUT    4                OUTPUT IT
006375  10AB  0DBD
006378  10AC  3030
006379                          *      LD     20               GET LINE CONTROL
006382  10AD  D614
006383                          *      OUT    2
006384  10AE  BD30
006387                          *      WAIT
006388  10AF  28BD
006389  10B0  3100
006390                          *      RECV   0
006391  10B1  BD30
006394                          *      ST
006395  10B2  60BD
006396  10B3  30C0
006397                          *      WAIT
006398  10B4  BD31
006399                          *      LOC    RAN
006400        021B              RAN EQU X'021B'
006401                          *      RECV   0
006402  10B5  00BD
006405  10B6  3060
006406                          *      ST
006407  10B7  BD30
006408                          *      BZF    BINTA
006409  10B8  C05D
006412  10B9  260C
006413                          *      IN     5
006414  10BA  BD30
006417                          *      AND    =1
006418  10BB  14C4
006421                          *      BZT    BINTA
006422  10BC  015D
006425  10BD  2704
006426                          *      LD     =X'FE'
006429  10BE  C6FE
006430                          *      ST     60
006433  10BF  D73C
006434                          *      LOC    BINTA
006435        0230              BINTA EQU X'0230'
006436                          *      BLCT   TONI
006437  10C0  DE63
006440  10C1  2705
006441                          *      WAIT
006442  10C2  BD31
006443                          *      B      RAN
006444  10C3  0020
006447                          *      LOC    TONI
006448        0239              TONI EQU X'0239'
006449                          *      LD     =0
006450  10C4  E2C6
006453                          *      OUT    2
006454  10C5  00BD
006457  10C6  3028
006458                          *      GNB
006459  10C7  BD31
006460                          *      WAIT
006461  10C8  20BD
006462  10C9  3100
006463                          *      NOP
006464                          *
006465                          *
006466                          *        CHANNEL PROGRAM FOR MARK & SPACE (XMIT)
006467                          *
006468                          *
006469                          *      ORG    X'400'
006470  10CA  0100
006471        10CB              CHLP9A EQU    $
006472                          *      LOC    CLRK
006473        0400              CLRK EQU X'0400'
006474                          *      LD,    GET              CHARACTER  TO SEND OUT
006475  10CB  BD30
006476                          *      C      =5               CHECK TO SET MARK AND SPACE CONDITION
006477  10CC  A086
006480  10CD  0510
006481  10CE  9741
006482                          *      BEF    DENW
006483  10CF  9641
006486  10D0  260B
006487                          *      LD     20               SET MARK OR SPACE CONDITION
006490  10D1  D614
006491                          *      OR     26
006494  10D2  DA1A
006495                          *      ST     20
006498  10D3  D714
006499                          *      OUT    2
006500  10D4  BD30
006503                          *      LD     =5
006504  10D5  28C6
006507                          *      LOC    DENW
006508        0417              DENW EQU X'0417'
006509                          *      C      =7               CHECK  TO RESET MARK AND SPACE CONDITION
006510  10D6  0586
006513  10D7  0710
006514  10D8  9741
006515                          *      BEF    BAMB
006516  10D9  9641
006519  10DA  2622
006520                          *      LD     20               RESET MARK OR SPACE CONDITION
006523  10DB  D614
006524                          *      XOR    26
006527  10DC  D81A
006528                          *      ST     20
006531  10DD  D714
006532                          *      OUT    2
```

```
006533  10DE  BD30
006536                        *       LD      26
006537  10DF  28D6
006540                        *       C       =X'10'          IF SPACE CONDITION CHECK FOR FRAMING ERROR
006541  10E0  1A86
006544  10E1  1010
006545  10E2  9741
006546                        *       BEF     KUNTA
006547  10E3  9641
006550  10E4  260C
006551                        *       IN      5
006552  10E5  BD30
006555                        *       AND     =1              MASK OTHER BITS OUT
006556  10E6  14C4
006559                        *       BZT     KUNTA           BRANCH ON FRAMING ERROR SET
006560  10E7  015D
006563  10E8  2704
006564                        *       LD      =X'EE'
006567  10E9  C6EE
006568                        *       ST      61
006571  10EA  D73D
006572                        *       LOC     KUNTA
006573        0440            KUNTA EQU X'0440'
006574                        *       LD      =7
006577  10EB  C607
006578                        *       LOC     BAMB
006579        0442            BAMB EQU X'0442'
006580                        *       SEND    0
006581  10EC  BD30
006584                        *       BLCT    RAY
006585  10ED  40DE
006586  10EE  6327
006589                        *       WAIT
006590  10EF  05BD
006591  10F0  3100
006592                        *       B       CLRK
006595  10F1  20B2
006596                        *       LOC     RAY
006597        044E            RAY EQU X'044E'
006598                        *       WAIT
006599  10F2  BD31
006600                        *       GNB
006601  10F3  00BD
006602  10F4  3120
006603                        *       LOC     MARK
006604        0454            MARK EQU X'0454'
006605                        *       LD      =X'FF'
006608  10F5  C6FF
006609                        *       SEND    0
006610  10F6  BD30
006613                        *       WAIT
006614  10F7  40BD
006615  10F8  3100
006616                        *       B       MARK
006619  10F9  20F6
006620                        *       NOP
006621        10FA            CHLP10 EQU    $
006622                        *
006623                        *=============================================================
006624                        *                       I/O FUNCTION CODES
006625                        *
006626  10FA  0001            FOMC    DC      X'1'            OUTPUT DLCP CONTROL
006627  10FB  0003            FOIC    DC      X'3'            OUTPUT INTERRUPT CONTROL
006628  10FC  0005            FOCC    DC      X'5'            OUTPUT CHANNEL CONTROL
006629  10FD  0009            FOLD    DC      X'9'            OUTPUT ADDRESS AND RANGE
006630  10FE  000B            FOBL    DC      X'B'            OUTPUT BYTE INTO LCT
006631  10FF  000F            FOCB    DC      X'F'            OUTPUT CCB CONTROL
006632  1100  000C            FIR     DC      X'C'            INPUT RANGE
006633  1101  0018            FIS     DC      X'18'           INPUT STATUS
006634  1102  001A            FINS    DC      X'1A'           INPUT NEXT STATUS
006635  1103  001C            FIDSS   DC      X'1C'           INPUT DATA SET STATUS
006636  1104  001E            FILS    DC      X'1E'           INPUT LCT ATAFUS
006637  1105  0026            FID     DC      X'26'           INPUT ID
006638                        *
006639                        *=============================================================
006640                        *                       STORAGE CONSTANTS
006641                        *
006642  1106  0000            ARFG    DC      0               ADAPTER READY TEST FLAG
006643  1107  0000            BYTP    DC      0               BYTE OFFSET FOR V$CALLS
006644  1108  0000            CHANL   DC      0
006645  1109  0000            CHSZFG  DC      0               CHARACTER SIZE TEST FLAG
006646  110A  0000            CHNZ    DC      0
006647  110B  0000            COUNT   DC      0
006648  110C  0000            CS      DC      0               CHARACTER SIZE
006649  110D  0000            CSB     DC      0               CHARACTER SIZE INCLUDING START AND STOP BITS
006650  110E  0000            CRCAC   DC      0               CRC ACUMMULATOR
006651  110F  0000            CRCFG   DC      0
006652  1110  0000            DIV1    DC      0               DIVISORS FOR TICKS TO MILLSEC SUB-ROUTINE
006653  1111  0000            DIV2    DC      0
006654  1112  0000            DIV3    DC      0
006655  1113  0000            DIV4    DC      0
006656  1114  0034            ERF     DC      52
006657  1115  0000            FBRG    DC      0
006658  1116  0000            FWFG    DC      0               FIRMWARE PRINT FLAG
006659  1117  0000            HGSPD   DC      0               HIGH SPEED FLAG
006660  1118  0000            HISPD   DC      0
006661  1119  003C            HRTZ    DC      60              LINE FREQUENCY
006662  111A  0000            MASK    DC      0
006663  111B  0000            MSBS    RESV    2,0             BITS/SEC
006664  111D  0000            PARFG   DC      0               PARITY ERROR TEST FLAG
006665  111E  0000            PTTY    DC      0               PARITY TYPE FLAG
006666  111F  0000            PSFG    DC      0               SYNC. PRINT SPEED FLAG
006667  1120  0000            RANGE   DC      0
006668  1121  0000            RCRC    DC      0               RECEIVE CRC RESIDUE
006669  1122  0000            RLCTAD  DC      0               RAM ADDRESS TO READ
006670  1123  0000            RLCTRG  DC      0               RANGE FOR READ FROM RAM
006671  1124  0000            RUNFG   DC      0               OVERRUN AND UNDERRUN TEST FLAG
006672  1125  0000            TLOC    DC      0               TEMPORARY LOCATION
006673  1126  0000            TP1     DC      0
006674  1127  0000            TP2     DC      0
006675  1128  0000            TRMRK   DC      0
006676  1129  0000            SBTFG   DC      0               STOP BIT TIME TEST FLAG
006677  112A  0000            SHORT   DC      0               TEST FLAG
```

```
1128  0000            STPFG  DC     0                  STOP I/O AND CHANNEL RESET FLAG
112C  0000            WDBFG  DC     0                  BROADBAND TEST FLAG
112D  0000            XCRC   DC     0                  TRANSMIT CRC RESIDUE
112E  FFFF            LCTV   RESV   3,-1
1131  0000            ERAR   RESV   4,0                ERROR ARRAY FOR ZV$C
                      *
1135                  SAV1   RESV   9+7*$AF
1145                  SAV2   RESV   9+7*$AF
1155                  SAV3   RESV   9+7*$AF
1165                  SAV4   RESV   9+7*$AF
                      *
                      *=====================================================================
                      *
                      *                         DEFINED CONSTANTS
                      *
1175  3108            ASYN   DC     X'3108'            ASYNC. ID
1176  3118            ASYN1  DC     X'3118'            ASYNC. ID NEW BAUD RATE
1177  3110            ASYNC  DC     X'3110'            CURRENT LOOP ASYNC.
1178  3158            SYNC   DC     X'3158'            SYNC. ID
1179  3150            BISYNC DC     X'3150'            BISYNC. ID
117A  3160            SYN188 DC     X'3160'            SYNC. MIL 188 ID
117B  3138            WDB3   DC     X'3138'            WIDEBAND WITH 301/303
117C  3130            WDB3B  DC     X'3130'            BI-SYNC. WIDEBAND WITH 301/303
117D  3168            WDBV35 DC     X'3168'            WIDEBAND WITH V35
117E  3128            WDBV5B DC     X'3128'            BI-SYNC. WIDEBAND WITH V35
                      *
117F  8000            BIT0   DC     Z'8000'
1180  4000            BIT1   DC     Z'4000'
1181  2000            BIT2   DC     X'2000'
1182  0800            BIT4   DC     X'800'
1183  0200            BIT6   DC     X'200'
1184  0100            BIT7   DC     X'100'
                      *
1185  A001            CRC16  DC     Z'A001'
                      *
1186  0000            ATLT   RESV   8,0                ACTIVE LINE TABLE
                      *
118E  0000            ADDRS  RESV   9,0                LINE ADDRESS TABLE
                      *
1197  0117            CPFLG  DC     X'117'             LCT FLAG FOR CRC ONLY
1198  F017                   DC     Z'F017'            PARITY ONLY
1199  0017                   DC     X'017'             CRC AND PARITY
119A  0000                   DC     0
                      *
119B  805C            CRLF   DC     Z'805C'
119C  2C00            COMMA  DC     X'2C00'
119D  005C            BKLS   DC     X'5C'
                      *
119E  0001 0203 0405  DOUT   TEXT   Z'000102030405060708090A0B0C0D0E0F'
11A1  0607 0809 0A0B
      0C0D 0E0F
11A6  1011 1213 1415         TEXT   Z'101112131415161718191A1B1C1D1E1F'
11A9  1617 1819 1A1B
      1C1D 1E1F
11AE  2021 2223 2425  DOUT1  TEXT   Z'202122232425262728292A2B2C2D2E2F'
11B1  2627 2829 2A2B
      2C2D 2E2F
11B6  3031 3233 3435         TEXT   Z'303132333435363738393A3B3C3D3E3F'
11B9  3637 3839 3A3B
      3C3D 3E3F
11BE  4041 4243 4445         TEXT   Z'404142434445464748494A4B4C4D4E4F'
11C1  4647 4849 4A4B
      4C4D 4E4F
11C6  5051 5253 5455         TEXT   Z'505152535455565758595A5B5C5D5E5F'
11C9  5657 5859 5A5B
      5C5D 5E5F
11CE  6061 6263 6465         TEXT   Z'606162636465666768696A6B6C6D6E6F'
11D1  6667 6869 6A6B
      6C6D 6E6F
11D6  7071 7273 7475         TEXT   Z'707172737475767778797A7B7C7D'
11D9  7677 7879 7A7B
      7C7D
11DD  5555            DFLT   DC     X'5555'            DEFAULT CHARACTER
                      *
11DE  0000            FIVSEC DC     0                  5 SECONDS OF TICK ON RTC
11DF  0000            SECL   DC     0
11E0  0000            SECH   DC     0
                      *
11E1  0000            FRIMRV DC     0                  FIRMWARE REV NUMBER
                      *
                      *      LINE SPEED IS IN SECOND WORD FOR LOW AND MEDIUM
                      *      SPEED LINES.  FOR HIGH SPEED LINES THE TWO
                      *      WORDS ARE PUT TOGETHER FORMING A SINGLE HEX WORD
                      *      CONSISTING OF 32 BITS.
                      *
11E2  0000            SYLSP0 RESV   2,0                SYNC. LINE 0 SPEED
11E4  0000            SYLSP1 RESV   2,0                SYNC. LINE 1 SPEED
11E6  0000            SYLSP2 RESV   2,0                SYNC. LINE 2 SPEED
11E8  0000            SYLSP3 RESV   2,0                SYNC. LINE 3 SPEED
11EA  0000            SYLSP4 RESV   2,0                SYNC. LINE 4 SPEED
11EC  0000            SYLSP5 RESV   2,0                SYNC. LINE 5 SPEED
11EE  0000            SYLSP6 RESV   2,0                SYNC. LINE 6 SPEED
11F0  0000            SYLSP7 RESV   2,0                SYNC. LINE 7 SPEED
                      *
11F2  000A            NSPTBL DC     10
11F3  000F                   DC     15
11F4  0016                   DC     22
11F5  001B                   DC     27
11F6  001E                   DC     30
11F7  0028                   DC     40
11F8  003C                   DC     60
11F9  0078                   DC     120
11FA  00D2                   DC     210
11FB  00F0                   DC     240
11FC  0168                   DC     360
11FD  0190                   DC     400
11FE  01E0                   DC     480
11FF  03C0                   DC     960
1200  0780                   DC     1920
1201  0F00                   DC     3840
                      *
1202  000A            OSPTBL DC     10
1203  000F                   DC     15
```

```
006775  1204  0016                    DC    22
006776  1205  001B                    DC    27
006777  1206  001E                    DC    30
006778  1207  003C                    DC    60
006779  1208  0078                    DC    120
006780  1209  00B4                    DC    180
006781  120A  00F0                    DC    240
006782  120B  0168                    DC    360
006783  120C  01E0                    DC    480
006784  120D  02D0                    DC    720
006785  120E  03C0                    DC    960
006786  120F  05A0                    DC    1440
006787  1210  0780                    DC    1920
006788  1211  0000                    DC    0
006789                          *
006790  1212  0032              ALSPBL DC    50              OLD BAUD RATE LINE SPEED TABLE
006791  1213  004B                    DC    75
006792  1214  006E                    DC    110
006793  1215  0087                    DC    135
006794  1216  0096                    DC    150
006795  1217  012C                    DC    300
006796  1218  0258                    DC    600
006797  1219  0384                    DC    900
006798  121A  04B0                    DC    1200
006799  121B  0708                    DC    1800
006800  121C  0960                    DC    2400
006801  121D  0E10                    DC    3600
006802  121E  12C0                    DC    4800
006803  121F  1C20                    DC    7200
006804  1220  2580                    DC    9600
006805                          *
006806  1221  0032              CMSPBL DC    50              NEW BAUD RATE LINE SPEED TABLE
006807  1222  004B                    DC    75
006808  1223  006E                    DC    110
006809  1224  0087                    DC    135
006810  1225  0096                    DC    150
006811  1226  00C8                    DC    200
006812  1227  012C                    DC    300
006813  1228  0258                    DC    600
006814  1229  041A                    DC    1050
006815  122A  04B0                    DC    1200
006816  122B  0708                    DC    1800
006817  122C  07D0                    DC    2000
006818  122D  0960                    DC    2400
006819  122E  12C0                    DC    4800
006820  122F  2580                    DC    9600
006821  1230  4B00                    DC    19200
006822                          *
006823  1231  0021              SBITBL DC    33
006824  1232  001E                    DC    30
006825  1233  0020                    DC    32
006826  1234  001E                    DC    30
006827                          *
006828  1235  0018              TLTBLA DC    X'18'           SHOULD BE TABLE FOR DSS TO DSC LOOP ASYC.
006829  1236  6078                    DC    X'6078'
006830  1237  8098                    DC    Z'8098'
006831  1238  E0F8                    DC    Z'E0F8'
006832                          *
006833  1239  B0B0              TLTBLC DC    Z'B0B0'         SHOULD BE TABLE FOR CURRENT LOOP ADAPT.
006834  123A  E0E0                    DC    Z'E0E0'
006835  123B  B0B0                    DC    Z'B0B0'
006836  123C  F0F0                    DC    Z'F0F0'
006837                          *
006838  123D  0111              TLTBLS DC    X'111'          SHOULD BE TABLE FOR DSS TO DSC LOOP SYNC.
006839  123E  6171                    DC    X'6171'
006840  123F  8191                    DC    Z'8191'
006841  1240  E1F1                    DC    Z'E1F1'
006842                          *
006843  1241  00E0              TLUWB3 DC    X'E0'           SHOULD BE TABLE FOR BROADBAND 301
006844  1242  10F0                    DC    X'010F0'
006845                          *
006846  1243  0060              TLUV35 DC    X'60'           SHOULD BE TABLE FOR BROADBAND V35
006847  1244  90F0                    DC    Z'90F0'
006848                          *
006849  1245  2020              SPACE  DC    Z'2020'
006850                          *
006851  1246  0020              TLUTB2 DC    X'20'
006852  1247  4060                    DC    X'4060'
006853  1248  0020                    DC    X'20'
006854  1249  4060                    DC    X'4060'
006855  124A  0010                    DC    X'10'
006856  124B  4050                    DC    X'4050'
006857  124C  0010                    DC    X'10'
006858  124D  4050                    DC    X'4050'
006859  124E  0000                    DC    0
006860  124F  0040                    DC    X'40'
006861  1250  0040                    DC    X'40'
006862                          *
006863  1251  0001              XMSB   DC    1               XMIT MARK SHOULD BE
006864  1252  0203                    DC    X'0203'
006865  1253  FF07                    DC    Z'FF07'
006866  1254  0809                    DC    X'0809'
006867                          *
006868  1255  0001              XSSB   DC    1               XMIT SPACE SHOULD BE
006869  1256  0203                    DC    X'0203'
006870  1257  0007                    DC    X'0007'
006871  1258  0809                    DC    X'0809'
006872                          *
006873  1259  0002              C2     DC    2
006874  125A  0004              C4     DC    4
006875  125B  0008              C8     DC    8
006876  125C  0010              C16    DC    16
006877  125D  0200              HX200  DC    X'200'
006878  125E  0400              HX400  DC    X'400'
006879                          *
006880                          *=============================================================
006881                          *
006882                          *                    TRAP SAVE AREAS
006883                          *
006884  125F  1267              TSA1   DC    <TSA2
006885  1260  0000                     RESV  4+3*$AF,0
006886  1267  126F              TSA2   DC    <TSA3
006887  1268  0000                     RESV  4+3*$AF,0
```

```
006888  126F  0000              TSA3   RESV  8,0
006889                          *
006890                          *=============================================================
006891                          *
006892                          *              INTERRUPT SAVE AREAS
006893                          *
006894  1277  0000              ISA1   RESV  3+$AF,0          INTERRUPT SAVE AREA FOR LEVEL 1 (RTC)
006895  127B  03D6              ISA1P  DC    <NOSPD
006896  127C  6000              DC    X'6000'
006897                          *
006898  127D  0000              ISA4   RESV  3+$AF,0
006899  1281  02CD              ISA4P  DC    <SPSYT1
006900  1282  6000              DC    X'6000'
006901                          *
006902                          *=============================================================
006903                          *
006904                          *                    LINE CONTROL TABLES
006905                          *
006906  1283  0206              LCT    DC    X'206'          STARTING ADDRESS OF CHANNEL PROG.
006907  1284  0414              DC    X'414'          RECV. LINE CONTROL SET TEST MODE
006908  1285  0000              DC    0               TABLE END
006909                          *
006910  1286  0206              LCT1   DC    X'206'
006911  1287  0426              DC    X'426'
006912  1288  C802              DC    Z'C802'
006913  1289  C822              DC    Z'C822'
006914  128A  0E18              LCT1S  DC    X'E18'
006915  128B  C714              LC1LCR DC    Z'C714'         LINE CONTROL
006916  128C  0000              LC1SYF DC    0               SYNC. FLAG
006917  128D  0000              DC    0               END OF TABLE
006918                          *
006919  128E  0206              LCT2   DC    X'206'
006920  128F  0426              DC    X'426'
006921  1290  C802              LC2CFR DC    Z'C802'
006922  1291  C822              LC2CFX DC    Z'C822'
006923  1292  0E18              LCT2S  DC    X'E18'
006924  1293  C714              DC    Z'C714'
006925  1294  0117              LCT2FG DC    X'117'          FLAG FOR TYPE OF SEND AND RECV
006926  1295  0000              DC    0
006927                          *
006928  1296  0206              LCT3   DC    X'206'          RAM STARTING ADDRESS (RECV)
006929  1297  0426              DC    X'426'          RAM STARTING ADDRESS (XMIT)
006930  1298  C002              LC3CFR DC    Z'C002'         CONFIGURATION (RECV)
006931  1299  C022              LC3CFX DC    Z'C022'         CONFIGURATION (XMIT)
006932  129A  0000              LCT3S  DC    0               SPEED IN SCRATCH LOCATION
006933  129B  C714              DC    Z'C714'         LINE CONTROL (RECV)
006934  129C  0000              LC3SYF DC    0               SYNC FLAG
006935  129D  0000              RESV  1,0
006936                          *
006937  129E  0206              LCT4   DC    X'206'
006938  129F  0426              DC    X'426'
006939  12A0  0002              LC4CFR DC    2
006940  12A1  0022              LC4CFX DC    X'22'
006941  12A2  0000              LCT4S  DC    0
006942  12A3  C714              DC    Z'C714'
006943  12A4  0000              RESV  1,0
006944                          *
006945  12A5  0206              LCT5   DC    X'206'
006946  12A6  0426              DC    X'426'
006947  12A7  C802              DC    Z'C802'
006948  12A8  C822              DC    Z'C822'
006949  12A9  0E18              LCT5S  DC    X'E18'
006950  12AA  C714              DC    Z'C714'
006951  12AB  0F10              DC    X'F10'          LCT STATUS BYTES
006952  12AC  0F11              DC    X'F11'          "
006953  12AD  0000              LC5SYF DC    0               SYNC. LINE FLAG
006954  12AE  0000              DC    0
006955                          *
006956  12AF  0206              LCT6   DC    X'206'
006957  12B0  0226              DC    X'226'
006958  12B1  C027              DC    Z'C027'
006959  12B2  C802              DC    Z'C802'
006960  12B3  C822              DC    Z'C822'
006961  12B4  0E18              LCT6S  DC    X'E18'
006962  12B5  C714              DC    Z'C714'
006963  12B6  021D              LCT6TF DC    X'21D'
006964  12B7  0125              DC    X'125'          DISABLE PAUSE
006965  12B8  0000              DC    0
006966                          *
006967  12B9  0206              LCT7   DC    X'206'
006968  12BA  0226              DC    X'226'
006969  12BB  C802              DC    Z'C802'
006970  12BC  C822              DC    Z'C822'
006971  12BD  FC1D              LC7SM  DC    Z'FC1D'         MASK TO CHECK BITS(FC-ASY.  F8-SYC.)
006972  12BE  0439              DC    X'439'          TLU POINTER TABLE  1
006973  12BF  0105              DC    X'105'
006974  12C0  0125              DC    X'125'
006975  12C1  0000              RESV  1,0
006976                          *
006977  12C2  0206              LCT8   DC    X'206'
006978  12C3  0426              DC    X'426'
006979  12C4  C802              DC    Z'C802'
006980  12C5  C822              DC    Z'C822'
006981  12C6  C518              DC    Z'C518'
006982  12C7  C714              DC    Z'C714'
006983  12C8  0105              DC    X'105'          DISABLE PAUSE
006984  12C9  0125              DC    X'125'          DISABLE PAUSE
006985  12CA  0000              LCT8FG DC    0
006986  12CB  0000              RESV  1,0
006987                          *
006988  12CC  0206              LCT9 * DC    X'206'
006989  12CD  0426              DC    X'426'
006990  12CE  C802              DC    Z'C802'
006991  12CF  C822              DC    Z'C822'
006992  12D0  0000              LCT9MS DC    0
006993  12D1  C314              DC    Z'C314'
006994  12D2  0000              DC    0
006995                          *
006996                          *
006997                          *=============================================================
006998                          *
006999                          *                    FORMAT  FOR DRIVER
007000                          *
```

```
007001  12D3  0006              WCP     DC      6                       RANGE OF PROGRAM IN BYTES
007002                          *
007003  12D4  0BF8              WCP1    DC      <CHLP1                  CHANNEL PROGRAM ADDRESS
007004  12D5  0C34                      DC      <CHLP1A                 XMIT CHANNEL PROGRAM ADDRESS
007005  12D6  0078              RAG1    DC      (CHLP1A-CHLP1)*2        RANGE OF CHANNEL PROGRAM
007006  12D7  005C              RAG2    DC      (CHLP2-CHLP1A)*2        RANGE OF PROGRAM IN BYTES
007007  12D8  0400                      DC      X'400'                  RAM ADDRESS
007008                          *
007009  12D9  0C62              WCP2    DC      <CHLP2                  CHANNEL PROGRAM ADDRESS
007010  12DA  0CB3                      DC      <CHLP2A                 XMIT CHANNEL PROGRAM ADDRESS
007011  12DB  00A2                      DC      (CHLP2A-CHLP2)*2        RANGE OF CHANNEL PROGRAM
007012  12DC  009E                      DC      (CHLP3-CHLP2A)*2        RANGE OF CHANNEL PROGRAM
007013  12DD  0400                      DC      X'400'                  RAM STARTING ADDRESS
007014                          *
007015  12DE  0D02              WCP3    DC      <CHLP3
007016  12DF  0D48                      DC      <CHLP3A
007017  12E0  008C                      DC      (CHLP3A-CHLP3)*2
007018  12E1  00B6                      DC      (CHLP4-CHLP3A)*2
007019  12E2  0400                      DC      X'400'
007020                          *
007021  12E3  0DA3              WCP4    DC      <CHLP4
007022  12E4  0DDF                      DC      <CHLP4A
007023  12E5  0078                      DC      (CHLP4A-CHLP4)*2
007024  12E6  00D4                      DC      (CHLP5-CHLP4A)*2
007025  12E7  02C0                      DC      X'2C0'
007026                          *
007027  12E8  0076              WCP5    DC      (CHLP6-CHLP5)*2
007028  12E9  0016              WCP5B   DC      22
007029                          *
007030  12EA  0E84              WCP6    DC      <CHLP6
007031  12EB  0EE9                      DC      <CHLP6A
007032  12EC  00CA                      DC      (CHLP6A-CHLP6)*2
007033  12ED  00D2                      DC      (CHLP7-CHLP6A)*2
007034  12EE  0400                      DC      X'400'
007035                          *
007036  12EF  0F52              WCP7    DC      <CHLP7
007037  12F0  0F84                      DC      <CHLP7A
007038  12F1  0064                      DC      (CHLP7A-CHLP7)*2
007039  12F2  0058                      DC      (CHLP8-CHLP7A)*2
007040  12F3  0400                      DC      X'400'
007041  12F4  0000                      RESV    2,0
007042                          *
007043  12F6  0FB0              WCP8    DC      <CHLP8
007044  12F7  1028                      DC      <CHLP8A
007045  12F8  00F0                      DC      (CHLP8A-CHLP8)*2
007046  12F9  0100                      DC      (CHLP9-CHLP8A)*2
007047  12FA  0400                      DC      X'400'
007048                          *
007049  12FB  10A8              WCP9    DC      <CHLP9
007050  12FC  10CB                      DC      <CHLP9A
007051  12FD  0046                      DC      (CHLP9A-CHLP9)*2
007052  12FE  005E                      DC      (CHLP10-CHLP9A)*2
007053  12FF  0400                      DC      X'400'
007054                          *
007055                          *=============================================================
007056                          *
007057                          *                       CONSOLE MESSAGES
007058                          *
007059  1300  434C 4120 5445    MESG    TEXT    'CLA TEST'
        1303  5354
007060                                  IFZ     ($AF-2),PLAF
007061  1304  2020 4443 4D43            TEXT    '  DCMC2,REV-A '
        1307  322C 5245 562D
              4120
007065  130B  2020 2041 5052    SLAF    TEXT    '   APR 06 1978$'
        130E  2030 3620 3139
              3738 2400
007066  1313  4153 594E 4348    MESG2   TEXT    'ASYNCHRONOUS LINE$'
        1316  524F 4E4F 5553
              204C 494E 4524
007067  131C  2053 5045 4544    MESG4   TEXT    ' SPEED IS  $'
        131F  2049 5320 2024
007068  1322  2053 484F 554C    MESG5   TEXT    ' SHOULD BE $'
        1325  4420 4245 2024
007069  1328  4E45 5854 2024    MESG6   TEXT    'NEXT $'
007070  132B  444C 4320 4348    MESG7   TEXT    'DLC CHANNEL(S) $'
        132E  414E 4E45 4C28
              5329 2024
007071  1333  2044 434D 4332    MESG8   TEXT    ' DCMC2 PASS$'
        1336  2050 4153 5324
007072  1339  504F 5745 5220    MESG9   TEXT    'POWER FREQ(HZ) $'
        133C  4652 4551 2848
              5A29 2024
007073  1341  434F 5059 5249            TEXT    'COPYRIGHT 1976 BY HONEYWELL  INFORMATION SYSTEMS INC.'
        1344  4748 5420 3139
              3736 2042 5920
              484F 4E45 5957
              454C 4C20 2049
              4E46 4F52 4D41
              5449 4F4E 2053
              5953 5445 4D53
              2049 4E43 2E00
007074  135C  4649 524D 5741    FREV    TEXT    'FIRMWARE REV.$'
        135F  5245 2052 4556
              2E24
007075  1363  2020 2020 2024    ERMG    TEXT    '     $'
007076  1366  434C 4120 4E4F    ERMG1   TEXT    'CLA NOT FOUND ON THIS CHANNEL  $'
        1369  5420 464F 554E
              4420 4F4E 2054
              4849 5320 4348
              414E 4E45 4C20
              2024
007077  1376  494E 434F 5252    ERMG2   TEXT    'INCORRECT ID ON THIS CHANNEL  $'
        1379  4543 5420 4944
              204F 4E20 5448
              4953 2043 4841
              4E4E 454C 2020
              2400
007078        1386              V$LB    EQU     $
007079  1C00                            ORG     ZERO+X'1C00'
007082        1C00              DIN     EQU     $
007083  1D00                            ORG     DIN+X'100'
007084        1D00              DIN1    EQU     $
```

```
007085  1386                          ORG    V$LB
007086                        *
007087  1386  0100                    END    DCMC2,STRT
0000  ERR COUNT
```

TITLE DCMC2, 'REV A' CLA TEST (SAF)

$AF	597	598C	606	607C	673	1091	1107	1619C	1621C	1635C
	1640C	1661	1758C	1791	1806C	1808C	1809C	1811C	1813C	1822
	1846C	1849	2086C	2415	2416C	2420	2421	2425	2431	2432
	2436	2438	2443C	2446	2452	2460	2461	6684	6685	6686
	6687	6885	6887	6894	6898	7060	7062	7080		
$B1	373	379	494	508	524	534	552	562	569	619
	626	646	651	656	670C	841	851	862	872	879
	931	1022	1032	1043	1053	1060	1117	1122	1126	1160
	1206	1216	1227	1237	1244	1294	1304	1317	1327	1334
	1479	1489	1503	1513	1520	1618	1619C	1620	1621C	1628
	1631	1634	1635C	1636	1639	1640C	1643	1646	1649	1658C
	1724	1733	1735	1735	1788C	2061C	2273	2276	2303	2304
	2431	2436	2437C	2465C						
$B2	331	336	372	375	440B	441B	442B	443B	444B	445B
	446B	447B	448B	457B	458B	459B	460B	461B	470B	471B
	472B	476B	577B	662B	900B	908	909	925	934	951
	978	1080B	1131B	1253B	1344B	1532B	1651B	1745B	2305	2308
	2384	2390	2420	2424C	2425	2427C	2451C	2455C	2461	2464C
	2479B									
$B3	486B	495B	525B	553B	570B	585B	621B	628B	647B	652B
	657B	674B	825B	842B	863B	880B	930	977	981	1005B
	1023B	1044B	1061B	1088B	1119B	1123B	1127B	1192B	1207B	1228B
	1245B	1283B	1295B	1318B	1335B	1466B	1480B	1504B	1521B	1616B
	1629B	1632B	1637B	1644B	1647B	1650B	1662B	1710B	1725B	1792B
	1799	1799	1801	1818	1839	1843	1843	1844	1949B	1993
	2003	2027	2430	2437C	2442	2443C	2465C			
$B4	332	333C	334	335C	406B	595	596C	597	598C	603
	604C	606	607C	634	635C	683B	684B	723B	732B	741B
	743B	770B	788B	808B	940B	984B	1089	1090C	1091	1092C
	1104	1105C	1107	1108C	1259B	1267B	1352B	1354B	1358B	1366B
	1367B	1370B	1373B	1386B	1394B	1547B	1558B	1663B	1682B	1688B
	1798B	1800B	1805	1806C	1807	1808C	1825B	1833B	1861B	1872B
	1886B	1893B	2008B	2064B	2074	2082B	2085	2086C	2088B	2098B
	2105B	2216B	2225B	2231B	2258B	2263B	2283B	2345B	2369B	2388B
	2394B	2419	2426	2451C	2463	2464C				
$B5	390	393	395C	424B	672B	691B	696B	705B	713B	727B
	798	799C	814	815C	950	996B	1179	1180C	1380B	1660B
	1673B	1677B	1686B	1753	1757	1758C	1790B	1801	1805	1807
	1809C	1810	1810	1811C	1812	1812	1813C	1818	1819C	1821B
	1844	1845C	1846C	1848B	1853B	1867B	1876B	1902B	2093B	2103B
	2132B	2173B	2185B	2187B	2229B	2273	2280B	2303	2312	2312
	2314B	2333B	2384	2385	2400B	2423	2424C			
$B6	407B	620	627	632	694	832	833	887	896	897
	953B	959B	967B	975	1118	1401B	1407B	1415B	1540B	1566B
	1717	1733	1770B	1774B	1958B	1987	2140	2202	2206	2208
	2348	2350	2354B	2415	2416C	2418	2427C	2454	2455C	
$B7	726B	740B	746B	769B	948B	990B	994B	1139B	1142B	1178B
	1262B	1265B	1270B	1273B	1357B	1360B	1376B	1390B	1398B	1413B
	1421B	1442B	1452B	1550B	1553B	1561B	1565B	1574B	1580B	1685B
	1691B	1698B	1765B	1781B	1829B	1837B	1866B	1875B	1890B	1896B
	1913B	1928B	2101B	2228B	2261B	2398B	2415	2459B		
$R1	295	296	336	337B	338	339	342	343	348	352
	353C	358	359C	371C	375	376	383C	384	385	389
	392	394	395C	676C	708C	730C	752B	809C	810B	812B
	1422	1423C	1424	1425	1426	1427C	1428C	1432	1436	1440
	1443C	1444	1449	1450	1453C	1454	1988C	1996C	1998	2000B
	2022	2023C	2113C	2116	2121	2126	2140	2141	2143	2144
	2151	2162	2168	2211	2212C	2274C	2276	2277C	2323	2324C
	2325	2327	2413	2456B						
$R2	330C	336	357C	358	374	379	381B	388	393	395C
	396B	677C	694	774C	777	1149C	1152	1158C	1284	1723
	1735	1990C	1998	1999	2001	2006C	2025C	2027	2061C	2140
	2208	2240	2276	2278B	2284	2304	2308	2385	2395	
$R3	348	349	350	351C	405	669C	695C	699C	721C	728C
	780	1266C	1351C	1361B	1362C	1392C	1539C	1554C	1659C	1700
	1789C	1831C	1852C	1858C	1869C	1884C	1892C	1932C	1955	1989C
	1993	2003	2073	2157	2180	2181C	2208	2209C	2302	2306
	2308	2310C	2311	2366	2367	2368	2409	2410	2411	2412C
	2414C	2430	2433	2441C	2442	2450C	2452			
$R4	344	351C	375	378B	379	391	393	394	412	413C
	414B	425	426	427	429	431	433	438	439C	454
	455C	456C	464	465C	467	468C	482	483C	484	485C
	506	507C	508	509	515	516C	522	523C	534	535
	539	540	542	543C	544	545C	546	547C	562	563
	583	584C	586	587	588	589C	590	591C	592	593
	594C	600	601C	611	612C	618	625	630	631C	638
	639C	640	641C	642	678C	680	681	685	686	687C
	694	700B	701C	702	704	709	710C	722	724	731
	738	742	744	755	756	763	765	772	773C	775
	776	789	790C	813	823	824C	828	829C	830	831C
	833	834C	851	852	872	873	887	888B	889	890
	892	893C	894	895C	897	909	910	911	912C	916
	918	920	922	923C	925	926	934	935	936	937
	938C	963	964	965C	971	972	973C	978	979	1003
	1004C	1007	1008C	1009	1010C	1011	1012C	1016	1017C	1032
	1033	1053	1054	1070	1071	1073C	1074	1075C	1086	1087C
	1094	1095	1096	1097C	1098	1099C	1101	1102C	1112	1113C
	1114	1115C	1116	1121	1125	1137	1140	1154	1155C	1156
	1157C	1159	1163	1166	1167C	1168	1169C	1190	1191C	1194
	1195C	1196	1197C	1198	1199C	1216	1217	1237	1238	1281
	1282C	1287	1288C	1304	1305	1309	1310C	1311	1312C	1327
	1328	1372	1374	1385	1387	1393	1395	1433	1434	1435
	1436	1437	1438C	1439	1440	1448	1450	1464	1465C	1470
	1471C	1473	1474C	1489	1490	1513	1514	1546	1548	1557
	1559	1614	1615C	1622	1623C	1624	1625C	1641	1642C	1667
	1668	1669C	1676	1681	1683	1708	1709C	1711	1712C	1714
	1718C	1738	1739	1743	1744C	1754	1755	1767	1768	1795
	1796	1802	1803	1804C	1824	1826	1832	1834	1839	1840C
	1841	1860	1864	1871	1873	1987	2001	2027	2039	2040C
	2045C	2046C	2048	2050C	2051C	2053	2054	2056C	2057	2059
	2060	2081	2083	2087	2089	2090C	2097	2099	2113C	2114
	2115C	2116	2117	2118	2119	2120C	2121	2122	2123	2124
	2125C	2126	2127	2128	2129	2147	2160	2167	2203	2204
	2224	2226	2241	2248	2257	2259	2282	2284	2344	2346
	2347C	2348	2349C	2350	2351C	2367	2386	2387	2390	2393
	2395									
$R5	343	344	345	346	363C	701C	709	710C	744	747
	749	750B	757C	811	945	954	955C	956	957C	987

```
2027  CPAR2    2062B
2040  CPAR3    2029B   2032B
2038  CPAR4    2035B
2039  CPAR5    2037B
3249  CPCRC    3214    3215    3217    3266    3267    3268
6717  CPFLG     326     832     896    1921
3271  CPPAR    3224    3225    3226    3288    3289    3294
5818  CPRT     5675    5676    5677
6225  CPRTX    6159    6160    6161
1986  CRC       984B
6711  CRC16     975
1996  CRCA     2002B   2005B
6650  CRCAC     986    2006C
2002  CRCB     2000B
6651  CRCFG     826C    899C   1069C   1078C   1900   1941C
6722  CRLF     2418    2454
 923  CRPR1     921B
 925  CRPR2     917B
 907  CRPRT    1902B
 823  CRPTB     442B    459B
 834  CRPTB1    888B    898B
 899  CRPTB2    891B
6648  CS        912C    976    1918    2022    2031    2034
6649  CSB       612C    641C    773C    790C    797C   2146
 379  CYN       381B
      DCMC2       1    7087
6508  DENW     6484    6485    6486
 356  DEVNF     341B    347B
6734  DFLT      499     557     574     846     867     884    1027    1048    1065    1211
               1232    1249    1448    1484    1508    1525    1794
7082  DIN       706     714     942    1440    1450    1678    1692    1759    1794    1854
               1907    1922    7080    7083
7084  DIN1      931     942     981    1921    1922
6652  DIV1      309C   2114
6653  DIV2      312C   2117
6654  DIV3      313C   2122
6655  DIV4      318C   2127
4348  DLAY     4190    4191    4192    4244    4245    4246    4254    4255    4256
 702  DLCCB     711B
6726  DOUT      499     500     697     846     847    1027    1028    1211    1212    1300
               1484    1485   1712C   1730   1744C
6728  DOUT1     530     557     558     574     575     867     868     884     885    1048
               1049    1065    1066    1232    1233    1249    1250    1323    1340    1508
               1509    1525    1526
4507  DSET     4488    4489    4491
1831  DT2      1827B
1839  DT3      1835B
1799  DTAL     1797B
1787  DTALOP    495B    525B    553B    570B    842B    863B    880B   1023B   1044B   1061B
               1207B   1228B   1245B   1295B   1318B   1335B   1480B   1504B   1521B   1725B
 995  EDCK      952B    993B
 326  EDLINT    291B    361B
4362  EDLY     4355    4356    4357    4392    4393    4394
2475  EDPCK     448B    461B    476B
3831  EDXT     3791    3792    3794    3818    3819    3820    3865    3866    3867
4575  ENDTC    4558    4559    4560
3858  EOP      3847    3848    3850
2427  ER11     2421
6682  ERAR      942     943     945     946     947    1692    1693    1695    1696    1697
               1759    1760    1762    1763    1764    1907    1908    1910    1911    1912
               1922    1923    1925    1926    1927
6656  ERF       767     768C   771C    2428
7075  ERMG      483C    584C    824C   1004C   1087C   1191C   1282C   1465C   1615C   1709C
               2408    2411   2412C   2417
7076  ERMG1     356
7077  ERMG2     382
5550  ERR      5540    5541    5542
2417  ERR1     2416C
2430  ERR2     2456B
2437  ERR21    2432    2435B
2436  ERR21A
2445  ERR3     2420    2431    2438   2440B   2443C   2461   2466B
2444  ERR31    2425    2436
2450  ERR34    2446    2448B
2456  ERR4     2453B
2458  ERR5     2429B
2407  ERROR     726B    740B    746B    769B    948B    990B    994B   1139B   1142B   1178B
               1262B   1265B   1270B   1273B   1357B   1360B   1376B   1390B   1398B   1413B
               1421B   1442B   1452B   1550B   1553B   1561B   1565B   1574B   1580B   1685B
               1691B   1698B   1765B   1781B   1829B   1837B   1866B   1875B   1890B   1896B
               1913B   1928B   2101B   2228B   2261B   2398B
4291  EXM1     4279    4280    4281
5920  EXT      5726    5727    5729    5751    5752    5754    5798    5799    5801    5861
               5862    5863    5886    5887    5888    5911    5912    5913
6657  FBRG    1423C   1454
6637  FID       390
6635  FIDSS
4849  FIF3     4828    4829    4830
4822  FIFE     4851    4852    4853
6636  FILS
6634  FINS     2087    2248
6632  FIR      2257    2344
6633  FIS       742    1372    1546    1557
6736  FIVSEC    594C    733     755     813
2301  FLN       672B   1660B   1790B
2306  FLN2     2309B
2313  FLN3     2307B
6630  FOBL     2282
6631  FOCB     2393
6628  FOCC      722     731    1385    1393    1681    1824    1832    1860    1871    2081
               2097    2241
6627  FOIC
6629  FOLD     2387
6626  FOMC     2224
7074  FREV      415
6740  FRIMRV    413C
5764  FVBC     5685    5686    5688
5758  FVBY     5778    5779    5781    5802    5803    5807
6658  FWFG      290     417C
2222  GENITZ    406B    684B   1663B   1800B
2648  GETC     2561    2562    2566    2667    2668    2669    2682    2683    2687
2662  GETC1    2638    2639    2640
2685  GETX     2675    2676    2677
```

```
5530  HANG      5547    5548    5552
6659  HGSPD      398C   2155C   2188
6660  HISPD      399C   1013    2212C
2530  HLP1      2701    2702    2703
2671  HLP1A     2659    2660    2664
6661  HRTZ       299     305     317     586     592    1094
6877  HX200      688    1670    1794    1815    2092
6878  HX400      689    1671
2256  INRG       808B   1358B   1367B
2613  INSY      2585    2586    2588
 421  INVD       401B    414B    434B    452B
2247  INXT       741B   1259B   1267B   1354B   1366B   1688B   1886B   1893B
 807  IRRG       634     817B
 811  IRRG1      812B
 813  IRRG2      810B
6894  ISA1       606    1107
6895  ISA1P      604C    635C   1105C
6898  ISA4       597    1091
6899  ISA4P      596C    799C    815C   1090C   1180C
2950  JFQQ      2922    2923    2925
3988  JFRR      3959    3960    3961
3451  JFYY      3423    3424    3426
2901  JHQQ      2947    2948    2952
3939  JHRR      3984    3985    3986
3390  JHYY      3448    3449    3453
1876  JP2       1874B
1880  JP2A      1878B
1886  JP2B      1881B
1892  JP3       1889B
1907  JP4       1456B   1846C   1901B
1897  JP4A      1895B
1915  JP5        950    1909B
1932  JP5A      1274B   1581B   1783B   1916B   1919B
1930  JP5B      1924B
 741  JPT1       739B    752B
 747  JPT2       745B
 672  JPT3       782B
 761  JPT4       798
 752  JPT5       748B
5416  KER       5405    5406    5408
6573  KUNTA     6548    6549    6550    6561    6562    6563
6915  LC1LCR     439C    455C    468C     539    543C
6916  LC1SYF     488C    547C
6921  LC2CFR     829C    889     893C   1195C
6922  LC2CFX     831C    895C   1197C
6930  LC3CFR    1113C   1143    1146C   1155C   1167C
6931  LC3CFX    1115C   1148C   1157C   1169C
6934  LC3SYF     613C    639C    761    1110C   2200
6939  LC4CFR    1008C   1070    1073C
6940  LC4CFX    1010C   1075C
6953  LC5SYF    1286C   1312C   1430
6971  LC7SM     1623C   1642C
3292  LCB1      3238    3239    3240    3260    3261    3263    3281    3282    3283
6906  LCT       2094
6910  LCT1       498     528     556     573
6914  LCT1S      485C    507C    516C    523C    545C
6919  LCT2       845     866     883    1210    1231    1248
6925  LCT2FG     834C   1199C
6923  LCT2S      836C    857C   1201C   1222C
6928  LCT3       692
2284  LCT3A     2285B
6932  LCT3S      678C    775
6937  LCT4      1026    1047    1064
2276  LCT4A     2286B
6941  LCT4S     1017C   1038C
6945  LCT5      1298    1321    1338
2282  LCT5A     2278B
6949  LCT5S     1288C   1310C
6956  LCT6      1483    1507    1524
6961  LCT6S     1474C   1498C
6963  LCT6TF    1471C   1496C
6967  LCT7      1674
6977  LCT8
6985  LCT8FG    1467C   1531C
6988  LCT9      1728
6992  LCT9MS    1718C   1739    1754    1768
6681  LCTV       411C    412     416     963     971    1411    1419    1572    1578    1779
                2352
4372  LESS      4380    4381    4382
5372  LKB1      5388    5389    5390
1660  LOPD      1701B
1661  LOPD1     1658C
1686  LOPD2     1684B
1700  LOPD3     1694B
1692  LOPD7     1619C   1621C   1635C   1640C   1690B
1658  LOPDS     1629B   1632B   1637B   1644B   1647B   1650B
2714  LP1A      2850    2851    2853
2808  LP1AA     2839    2840    2844
2817  LP1AB     2805    2806    2810
2794  LP1ABD    2740    2741    2742    2813    2814    2815
2842  LP1AC1    2832    2833    2834
1614  LPCBLK     471B
 482  LPTA       440B    457B    470B
1815  LPWRT     1806C   1809C
1816  LPWRT1    1808C   1811C   1813C
6604  MARK      6617    6618    6619
6662  MASK       487C    827C    923C    942    1012C   1617C   1692    1713C   1759    1907
                1920C   1922    1930C
2395  MCB2      2396B
2389  MCB5      2392B
2399  MCB6      2397B
2382  MCCB       696B    705B    713B   1677B   1848B   1853B
2398  MCER      2389B
1867  MELCO     1863B   1865B
 727  MELCO1     725B
7059  MESG       289
7066  MESG2     2182
7067  MESG4     2184
7068  MESG5     2207
7069  MESG6      421
7070  MESG7      328
7071  MESG8     2478
7072  MESG9      298
```

```
6663 MSBS     2156   2164   2171C  2190   2192C  2196C  2197   2199   2211   2213C
              2214C
2262 N1       2260B
2258 N1A      2242B  2249B
2229 N2       2227B
3650 N61      3629   3630   3631
6031 N611     6010   6011   6012
3680 N71      3659   3660   3661
6061 N711     6040   6041   6042
1177 NBITT    1104   1182B
 544 NCBL      514B   541B
5734 NFBA     5695   5696   5697   5755   5756   5760
5740 NFVB
 363 NODEVF    334
 796 NOSPD     603    801B   6895
3519 NPRY     3489   3490   3491   3536   3537   3538
3997 NSCH     4013   4014   4016
6756 NSPTBL    620    627
2959 NSTR     2975   2976   2977
4318 NSXM     4307   4308   4310   4342   4343   4344
4211 NSY      4084   4085   4086   4237   4238   4242
4662 NSYK     4634   4635   4637
5363 NSYK1    5335   5336   5338
4269 NT3      4205   4206   4207   4286   4287   4293
 290 NTYR      288B
3316 NWT      3308   3309   3311
4384 NXC      4376   4377   4378
1790 NXLT     1934B
 678 NXSPD     778B   1162B  1164B
 680 NXSPD1   1150B   1179
 684 NXSPD2    682B
4155 NZ7      4181   4182   4188
5147 ORUD     5012   5013   5015
6773 OSPTBL
5244 OVCK     5233   5234   5235
4240 OVLY     4230   4231   4232
5197 OVR      5156   5157   5158
1259 PAR1     1883B
1263 PAR2     1261B
1266 PAR3     1264B
1271 PAR4     1269B
1274 PAR5     1272B
1190 PAREKF    445B   460B
6664 PARFG    1193C  1252C  1715C  1741C  1880   1942C
 451 PCH       432B
 359 PDRS      386B
5032 PGCX     5055   5056   5057
7064 PLAF     7060
2211 PRLT1    2201B
2199 PRLT3    2189B
2196 PRLT5    2194B
2207 PRLT7    2205B
2179 PRTLSP    770B   788B
3496 PRYT     3513   3514   3521
6666 PSFG      397C   659C   784
6665 PTTY      938C  2057
 553 QA1       565B
 863 QA2       875B
1044 QA3      1056B
1228 QA4      1240B
1318 QA5      1330B
1504 QA6      1516B
1400 QT1      1396B
1392 QT2      1388B
7005 RAG1      688
7006 RAG2      689
6400 RAN      6445   6446   6450
6667 RANGE     631C   632    932    942    982    1422   1428C  1840C  1907   1921
              1922
6597 RAY      6587   6588   6590
3482 RCCH     3387   3388   3392
 342 RCK1      340B
3460 RCKSY    3476   3477   3478
3349 RCP3     3559   3560   3561
6668 RCRC      965C   987
 991 RCRCG     989B
 960 RCRRL     955C
 968 RCRXL     957C
3078 RCV      3021   3022   3024   3043   3044   3045   3066   3067   3069
2982 RCVA     2898   2899   2903   3094   3095   3097
2343 RDLCT     407B   959B   967B  1407B  1415B  1566B  1774B
5876 RE1
5869 RE1A     5831   5832   5834   5890   5891   5892
5901 RE2
5894 RE2A     5841   5842   5843   5915   5916   5917
5851 RE3
5845 RE3A     5865   5866   5867
3033 RECV1    2995   2996   2997   3050   3051   3057
3055 RECV2    3006   3007   3012   3072   3073   3074
3010 RECV3    3027   3028   3029
4041 REND     4029   4030   4031
4709 RERR     4683   4684   4686
6669 RLCTAD   2349C  2352
6670 RLCTRG   2351C  2352
4019 RMCH     3935   3936   3937   4036   4037   4043
1567 RNCK1    1542C
1554 RNCK4    1545B  1552B
1551 RNCK4A   1549B
1563 RNCK4B   1560B
1566 RNCK5    1556B  1564B
1581 RNCK6    1573B  1576B  1579B
1578 RNCK9    1571B
 375 RON       378B
5238 ROTR     5218   5219   5221
 438 RTST      449B
 467 RTST1     475B
1539 RUNCK    1898B
6671 RUNFG    1469C  1530C  1897   1948C
2469 SAFF     2460
 336 SALT      354B
6684 SAV1     1986C  2007   2021C  2063   2111C  2131   2138C  2172   2239C  2247C
              2256C  2262   2272C  2279   2301C  2313   2322C  2332   2343C  2353
              2382C  2399
```

```
6685  SAV2      907C   995   1752C  1782   2072C  2080C  2104   2179C  2215   2222C
                2230
6686  SAV3      668C   781   1787C  1933
6687  SAV4      2407C  2430  2442   2458
1101  SBIT1     1089
6823  SBITBL    1118
1137  SBTE       759B
1140  SBTE1     1138B
1143  SBTE2     1141B
1152  SBTE3     1145B
1166  SBTE4     1153B
6676  SBTFG      758   1111C  1130C  1943C
2080  SCLS       683B  1798B
2083  SCLS1     2084B
5646  SCY1
5617  SCY2
5002  SDBB      4992   4993   4994
4131  SDRR      4121   4122   4123
2779  SDSY      2769   2770   2772
3184  SDVV      3174   3175   3176
5501  SDWW      5491   5492   5493
3747  SDZZ      3737   3738   3739
6738  SECH       591C   765   1099C  1137
6737  SECL       589C   763   1097C  1140
2272  SETLCT     691B  1673B  1821B  2093B
2461  SETNR     2434B
 713  SGCCB      703B
3543  SHFL      3506   3507   3508   3530   3531   3533
6677  SHORT      437C   456C   466C   513    609    786
 454  SHTST      428B   462B
4613  SINK      4659   4660   4664
5314  SINK1     5360   5361   5365   5432   5433   5434
7065  SLAF      7062
3201  SNC1      3137   3138   3139   3312   3313   3318   3336   3337   3338
3228  SNC3      3245   3246   3251
6849  SPACE     2419   2463
2467  SPACE1    2449
 668  SPDTLP     621B   628B   647B   652B   657B  1119B  1123B  1127B
 600  SPSYT1     595   6899
 583  SPSYTC     441B   458B
 673  ST1        670C   680   2204
 744  ST3        751B
1086  STBITE     444B
2239  STCR      1352B  1370B
1281  STIOG      446B
1351  STIO      1879B
1352  STIO1     1381B
1379  STIO2     1365B
1367  STIO3     1369B
1377  STIO4     1375B
1355  STIO5     1377B
2072  STMD
2103  STMD1     2100B
2086  STMD2     2075B
2092  STMD3     2086C
 279  STOP      2475C  2477B
 289  STP        284B
4901  STP3      4755   4756   4757   4784   4785   4786   4810   4811   4812
6678  STPFG     1285C  1343C  1877   1944C
 283  STRT      7087
 630  STSP1      610B
2468  SWT       2422C  2439   2447   2462C
5705  SXB1
5699  SXBC      5718   5719   5720   5730   5731   5736
5640  SYC       5630   5631   5632   5660   5661   5664
5606  SYCH      5637   5638   5642
 736  SYCT       737B
6747  SYLSP0    2157
6748  SYLSP1
6749  SYLSP2
6750  SYLSP3
6751  SYLSP4
6752  SYLSP5
6753  SYLSP6
6754  SYLSP7
6698  SYN188     569    656    879   1060   1244   1334   1520   1649
6696  SYNC       552    651    862   1043   1227   1317   1503   1646
 784  SYNSP      762B   814
 789  SYNSP1     785B   787B
 494  T1
 872  T10        864B
 879  T12        874B
1022  T14
1023  T14A      1035B
1032  T15       1024B
1036  T15A      1034B
1053  T16       1045B
1060  T17       1055B
1294  T19
1295  T19A      1307B
 495  T1A        511B
 508  T2         505B
1304  T20       1296B
1308  T20A      1306B
1327  T21       1319B
1334  T22       1329B
1479  T24
1480  T24Z      1492B
1489  T25       1481B
5394  T25A      5413   5414   5418
5517  T25B      5573   5574   5576
1493  T25Z      1491B
1513  T26       1505B
1520  T27       1515B
 512  T2A        510B
 504  T2AA       496B
 522  T3
 577  T30        571B
1529  T32       1522B
1343  T34       1336B
1069  T36       1014B  1062B
 887  T38        881B
1206  T39
```

```
1207  T39A     1219B
 525  T3A       537B
 534  T4        526B
1216  T40      1208B
1220  T40A     1218B
1237  T41      1229B
1244  T42      1239B
1252  T45      1246B
 538  T4A       536B
 562  T5        554B
 569  T6        564B
 841  T8
 842  T8A       854B
 851  T9        843B
 855  T9A       853B
5278  TAMPA    5283   5284   5286
4475  TCMB1    4445   4446   4450   4572   4573   4577
2323  TDLA     2331B
2327  TDLA1    2329B
2330  TDLA3    2328B
2322  TDLAY     424B   727B  1380B  1686B  1867B  1876B  2103B  2229B
1791  TG1      1788C  1795
1822  TG2       908  1819C
1849  TG3       930   977   1845C
5805  TKF      5791   5792   5796
5794  TKFN     5814   5815   5816
2111  TKSEC    2185B
6672  TLOC      306C   315    359C   360    422    425   1992C  2004   2045C  2048
               2050C  2053   2115C  2119   2120C  2124  2125C  2129   2181C  2183
               2209C  2210
6828  TLTBLA   1618
6833  TLTBLC   1634
6838  TLTBLS   1639
6851  TLUTB2   1671
6846  TLUV35
6843  TLUWB3
6448  TONI     6438   6439   6440
4069  TOP      4313   4314   4315   4320
4408  TOP1     4583   4584   4586
4469  TOP2     4457   4458   4459
4448  TOP3     4429   4430   4432
6673  TP1      1363C  1425   1433   2148   2158
6674  TP2       422   1400C  1434   1438C  1444   1625C  1676   2150
5129  TRCV     5115   5116   5117
6675  TRMRK    1716C  1742C  1862
2564  TRNX     2610   2611   2615
 767  TRYH      764B
6884  TSA1      332
6886  TSA2     6884
6888  TSA3     6886
 780  TSNL      700B   791B  1170B
1422  TSTF2    1420B
1408  TSTF2B   1403C
1358  TSTF3    1356B
1361  TSTF4    1359B
1385  TSTF4A   1361B
1448  TSTF4C   1431B  1445B
1450  TSTF4D   1455B
1453  TSTF4E   1451B
1443  TSTF5    1441B
1440  TSTF5A   1446B
1415  TSTF7K   1412B
1416  TSTF8B   1405C
 774  TYXL      766B
1732  U1       1726B
1717  U2       1740B
1725  U3       1736B
1738  U5       1734B
4788  UDUR     4679   4680   4682
4816  UND      4797   4798   4800
1464  UORNTH    447B
7078  V$LB     7085
7001  WCP      2092
7003  WCP1      497    527    555    572   1297   1320   1337
7009  WCP2      844    865    882   1209   1230   1247
7015  WCP3     1025   1046   1063
7021  WCP4     1482   1506   1523
7027  WCP5     1670
7028  WCP5B    1671
7030  WCP6
7036  WCP7
7043  WCP8
7049  WCP9     1727
6699  WDB3
6700  WDB3B
6679  WDBFG    1364   1468C  1494C  1529C  1544   1555   1570   1575   1945C
6701  WDBV35
6702  WDBV5B
6680  XCRC      973C   991
4971  XMBB     4999   5000   5004
1708  XMMSL     472B
3763  XMNS     3709   3710   3712   3853   3854   3855
4100  XMRR     4128   4129   4133
6863  XMSB     1753
1751  XMSCK    1885B
1758  XMSCK1   1756B
1759  XMSCK2   1758C
1767  XMSCK3   1761B
1775  XMSCK4   1772C
1782  XMSCK5   1769B  1780B
2748  XMSY     2775   2776   2777
3153  XMVV     3181   3182   3186
5470  XMWW     5498   5499   5503
3716  XMZZ     3744   3745   3749
3804  XNPY     3770   3771   3778   3825   3826   3833
3776  XPY      3797   3798   3799
3696  XPYQ     3616   3617   3622   3647   3648   3652   3677   3678   3682
6171  XPYZ     6194   6195   6200
6868  XSSB     1757
6111  Z2Z
6137  Z3Z      6132   6133   6135
6078  Z47      6000   6001   6003   6028   6029   6033   6058   6059   6063
6092  Z48      5979   5980   5981
```

```
  272  ZERO      278    285    7079    7081
 6322  ZEXT     6186   6187    6189    6213    6214    6216    6264    6265    6267    6287
                6288   6290    6310    6311    6313
       ZHCOMM    277
       ZHIAFB    274
       ZHISAZ    274    598C    607C   1092C   1108C
       ZHNTSA    273    333C
       ZHPFR     277
       ZHRTCC    275    734C    736     756    2324C   2327
       ZHRTCI    276
       ZHRTCL    276    601C   1102C
       ZHTH15    273    335C
       ZHWDTC    275
       ZV$C      942B   1692B   1759B   1907B   1922B
       ZV$CQ     949B   1699B   1766B   1914B   1929B
       ZV$ER    2417B
       ZV$F      326B   1794B   1921B
       ZV$HR     273
       ZV$IA     422B
       ZV$ID     299B
       ZV$IH     329B
       ZV$IZ     286B
       ZV$MLR   2352B
       ZV$MLW    688B    689B   1670B   1671B   1815B   1816B   2092B
       ZV$PCH    451B
       ZV$QC     298B    328B    421B
       ZV$RD     286B    289B    660B   1129B
       ZV$T     2184B   2198B   2207B   2449B
       ZV$TC     356B    382B    415B   2182B   2478B
       ZV$TD     416B   2183B   2197B   2199B   2210B
       ZV$TH     360B   2444B
       ZV$THZ   2445B
       ZV$TTY    275    287C    2476
 6198  ZXZ1     6168   6169    6173    6221    6222    6227
 678 LABELS
3109 REFERENCES
7087 RECORDS
   1 U FLAGS
   0 M FLAGS
  43 N FLAGS
6 CROSS REF VERSION L - 24 SEPT, 1976
RS LINKER VERSION 5.00 04/12/78  2052.6 EST WED
LINK MAP FOR DCMC2
 START        0100
 LOW          0000
 HIGH         1CFF
 CURRENT      1C9A
*LOC DEFS
  ZHCOMM      0000
 *DCMC2       0000  REV A
  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$F        1386
  ZV$F        1386
 *ZV$IH       1394
  ZV$ID       1399
  ZV$IH       1394
  ZV$IAD      139E
  ZV$=-2      13B6
  ZV$=-3      13C8
 *ZV$TH       142D
  ZV$TH       142D
  ZV$TD       1462
  ZV$THZ      1455
```

```
*ZV$IA      147D  REV. 7
 ZV$IA      1480
 ZV$ARG     152F
 ZV$ABF     1531
 ZV$--1     14EC
 ZV$IAV     147E
*ZV$PCH     153C
 ZV$PCH     153C
*ZV$MLW     163E  REV. 0
 ZV$MLW     163E
 ZV$MLR     166D
*ZV$C       1685  REV. 5
 ZV$C       1685
 ZV$CO      16A8
*ZV$T       16B9  REV. 5.0
 ZV$QC      16D6
 ZV$TC      16C2
 ZV$T       16B9
 ZV$Q       16CB
*ZV$ER      16EA  REV. 5.0
 ZV$ER      16EA
 ZV$TA      1716
 ZV$--0     16FD
*ZV$GP      175A
 ZV$GP      175A
 ZV$--4     177A
*ZV$HA      1786
 ZV$HA      1786
 ZV$HZ      1790
 ZV$HS      178B
*ZV$HD      17BF
 ZV$HD      17BF
*ZV$BRK     17F1
 ZV$BRK     17F1
*ZV$RD      180B  REV. 7
 ZV$IZ      1845
 ZV$TTY     181E
 ZV$RD      180B
 ZV$SV1     19E0
 ZV$SV3     1A00
 ZV$AF      181C
 ZV$SV2     19F0
 ZV$TID     181D
 ZV$CF2     1827
 ZV$TK      1823
 ZV$RAR     1824
 ZV$ST1     1828
 ZV$RCC     1829
 ZV$BUD     181F
 ZV$OLB     182B
 ZV$RCB     182C
 ZV$NSR     1830
 ZV$STR     182E
 ZV$BKF     1833
 ZV$OTP     18B2
 ZV$BKS     1832
 ZV$HR      183A
 ZV$LR      1837
 ZV$DAT     181A
 ZV$HM      1881
 ZV$HRU     1834
 ZV$HRL     1835
 ZV$LRU     1836
 ZV$LRL     1837
 ZV$HBD     1838
 ZV$CF1     1826
 ZV$--5     183D
 ZV$RMD     181B
 ZV$MCP     1839
 HIBAUD     1838
 ZV$RAW     1825
 ZV$RDT     1A3C
 ZV$CTL     1822
 ZV$B1      195D
 ZV$TST     1A92
 ZV$MDC     1A66
 ZV$R99     1C64
 ZV$ISA     1840
 ZV$UIH     183B
 ZV$ZRO     18BF
 ZV$BSH     18C1
 ZV$CPU     1821
 ZV$R50     189F
 ZV$R60     18AA
 ZV$RT      1BA1
 ZV$ALL     1820
*MLCHPG     1C69  T+V
 MLCHPG     1C69
 ENDCHP     1C9A
```