DH11

MULTI-LINE PARITY

MD-11-DZDHG-B

EP-DZDHG-B-DL-A

COPYRIGHT © 1976

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA

```
                                ;DH11 SINGLE LINE PARITY CHECK & MULTI-LINE DATA TEST
                                ;COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

                                        ;STARTING PROCEDURE
                                        ;LOAD PROGRAM
                                        ;LOAD ADDRESS 000200
                                        ;PRESS START
                                        ;PROGRAM WILL TYPE DH11 SINGLE LINE PARITY CHECK & MULTI-LINE DATA TEST
                                        ;PROGRAM WILL TYPE "VECTOR ADDRESS-"
                                        ;TYPE IN THE ADDRESS OF THE RECEIVER INTERRUPT VECTOR
                                        ;FOR THE DH11 TO BE TESTED, FOLLOWED BY <CARRIAGE RETURN>
                                        ;PROGRAM WILL TYPE "CONTROL REGISTER ADDRESS-"
                                        ;TYPE IN THE ADDRESS OF THE SYSTEM CONTROL REGISTER
                                        ;FOR THE DH11 TO BE TESTED, FOLLOWED BY <CARRIAGE RETURN>
                                        ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                        ;AT THE END OF A PASS, PROGRAM WILL TYPE " DZDHG "
                                        ;AND THEN RESUM TESTING

                                ;SWITCH REGISTER OPTIONS

        100000                  SW15=100000             ;=1,HALT ON ERROR
        040000                  SW14=40000              ;=1,LOOP ON CURRENT TEST
        020000                  SW13=20000              ;=1,INHIBIT ERROR TYPEOUT
        010000                  SW12=10000
        004000                  SW11=4000               ;=1,INHIBIT ITERATIONS
        002000                  SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
        001000                  SW09=1000               ;=1,LOOP WITH CURRENT DATA
        000400                  SW08=400
        000100                  SW06=100
        000040                  SW05=40
        000020                  SW04=20
        000010                  SW03=10
        000004                  SW02=4
        000002                  SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
        000001                  SW00=1                  ;RESELECT VECTOR AND CONTROL REGISTER
                                                        ;ADDRESS AFTER PROGRAM RESTART
```

```
                                :REGISTER DEFINITIONS

000000                          R0=%0           ;GENERAL REGISTER
000001                          R1=%1           ;GENERAL REGISTER
000002                          R2=%2           ;GENERAL REGISTER
000003                          R3=%3           ;GENERAL REGISTER
000004                          R4=%4           ;GENERAL REGISTER
000005                          R5=%5           ;GENERAL REGISTER
000006                          SP=%6           ;PROCESSOR STACK POINTER
000007                          PC=%7           ;PROGRAM COUNTER

                                ;LOCATION EQUIVALENCIES

177570                          SWR=177570      ;CONSOLE SWITCH REGISTER
177570                          LIGHTS=177570   ;PDP-11/45 DISPLAY REGISTER
177776                          PS=177776       ;PROCESSOR STATUS WORD
013564                          STACK=ENDCOD+200;START OF PROCESSOR STACK

                                ;INSTRUCTION DEFINITIONS

005746                          PUSH1SP=5746    ;DECREMENT PROCESSOR STACK 1 WORD
005726                          POP1SP=5726     ;INCREMENT PROCESSOR STACK 1 WORD
010046                          PUSHR0=10046    ;SAVE R0 ON STACK
012600                          POPR0=12600     ;RESTORE R0 FROM STACK
024646                          PUSH2SP=24646   ;DECREMENT STACK TWICE
022626                          POP2SP=22626    ;INCREMENT STACK TWICE
                                .EQUIV  EMT,HLT ;BASIC DEFINITION OF ERROR CALL


100000                          BIT15=100000
040000                          BIT14=40000
020000                          BIT13=20000
010000                          BIT12=10000
004000                          BIT11=4000
002000                          BIT10=2000
001000                          BIT09=1000
000400                          BIT08=400
000200                          BIT07=200
000100                          BIT06=100
000040                          BIT05=40
000020                          BIT04=20
000010                          BIT03=10
000004                          BIT02=4
000002                          BIT01=2
000001                          BIT00=1
```

```
                                                ;TRAPCATCHER FOR ILLEGAL INTERRUPTS
                 000000                 .=0
 000000  000002                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000002  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000004  000006                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000006  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000010  000012                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000012  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000014  000016                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000016  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000020  000022                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000022  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000024  000026                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000026  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000030  000032                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000032  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000034  000036                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000036  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000040  000042                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000042  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000044  000046                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000046  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000050  000052                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000052  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000054  000056                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000056  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000060  000062                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000062  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000064  000066                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000066  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000070  000072                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000072  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000074  000076                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000076  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000100  000102                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000102  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000104  000106                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000106  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000110  000112                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000112  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000114  000116                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000116  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000120  000122                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000122  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000124  000126                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000126  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000130  000132                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000132  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000134  000136                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000136  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000140  000142                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000142  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000144  000146                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000146  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
 000150  000152                         .+2             ;UNEXPECTED TRAP TO THIS LOCATION
 000152  000000                         HALT            ;EXAMINE STACK TO FIND CAUSE
```

```
142  000154  000156                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
143  000156  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
144  000160  000162                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
145  000162  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
146  000164  000166                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
147  000166  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
148  000170  000172                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
149  000172  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
150  000174  000176                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
151  000176  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
152  000200  000202                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
153  000202  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
154  000204  000206                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
155  000206  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
156  000210  000212                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
157  000212  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
158  000214  000216                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
159  000216  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
160  000220  000222                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
161  000222  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
162  000224  000226                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
163  000226  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
164  000230  000232                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
165  000232  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
166  000234  000236                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
167  000236  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
168  000240  000242                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
169  000242  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
170  000244  000246                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
171  000246  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
172  000250  000252                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
173  000252  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
174  000254  000256                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
175  000256  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
176  000260  000262                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
177  000262  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
178  000264  000266                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
179  000266  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
180  000270  000272                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
181  000272  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
182  000274  000276                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
183  000276  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
184  000300  000302                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
185  000302  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
186  000304  000306                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
187  000306  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
188  000310  000312                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
189  000312  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
190  000314  000316                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
191  000316  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
192  000320  000322                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
193  000322  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
194  000324  000326                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
195  000326  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
196  000330  000332                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
197  000332  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
```

```
198  000334  000336          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
199  000336  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
200  000340  000342          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
201  000342  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
202  000344  000346          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
203  000346  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
204  000350  000352          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
205  000352  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
206  000354  000356          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
207  000356  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
208  000360  000362          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
209  000362  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
210  000364  000366          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
211  000366  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
212  000370  000372          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
213  000372  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
214  000374  000376          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
215  000376  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
216  000400  000402          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
217  000402  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
218  000404  000406          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
219  000406  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
220  000410  000412          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
221  000412  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
222  000414  000416          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
223  000416  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
224  000420  000422          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
225  000422  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
226  000424  000426          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
227  000426  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
228  000430  000432          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
229  000432  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
230  000434  000436          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
231  000436  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
232  000440  000442          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
233  000442  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
234  000444  000446          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
235  000446  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
236  000450  000452          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
237  000452  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
238  000454  000456          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
239  000456  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
240  000460  000462          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
241  000462  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
242  000464  000466          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
243  000466  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
244  000470  000472          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
245  000472  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
246  000474  000476          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
247  000476  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
248  000500  000502          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
249  000502  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
250  000504  000506          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
251  000506  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
252  000510  000512          .+2     ;UNEXPECTED TRAP TO THIS LOCATION
253  000512  000000          HALT    ;EXAMINE STACK TO FIND CAUSE
```

```
254  000514  000516                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
255  000516  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
256  000520  000522                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
257  000522  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
258  000524  000526                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
259  000526  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
260  000530  000532                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
261  000532  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
262  000534  000536                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
263  000536  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
264  000540  000542                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
265  000542  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
266  000544  000546                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
267  000546  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
268  000550  000552                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
269  000552  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
270  000554  000556                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
271  000556  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
272  000560  000562                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
273  000562  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
274  000564  000566                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
275  000566  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
276  000570  000572                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
277  000572  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
278  000574  000576                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
279  000576  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
280  000600  000602                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
281  000602  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
282  000604  000606                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
283  000606  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
284  000610  000612                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
285  000612  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
286  000614  000616                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
287  000616  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
288  000620  000622                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
289  000622  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
290  000624  000626                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
291  000626  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
292  000630  000632                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
293  000632  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
294  000634  000636                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
295  000636  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
296  000640  000642                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
297  000642  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
298  000644  000646                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
299  000646  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
300  000650  000652                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
301  000652  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
302  000654  000656                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
303  000656  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
304  000660  000662                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
305  000662  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
306  000664  000666                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
307  000666  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
308  000670  000672                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
309  000672  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
```

```
000674  000676                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000676  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000700  000702                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000702  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000704  000706                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000706  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000710  000712                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000712  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000714  000716                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000716  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000720  000722                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000722  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000724  000726                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000726  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000730  000732                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000732  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000734  000736                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000736  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000740  000742                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000742  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000744  000746                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000746  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000750  000752                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000752  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000754  000756                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000756  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000760  000762                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000762  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000764  000766                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000766  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000770  000772                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000772  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
000774  000776                    .+2        ;UNEXPECTED TRAP TO THIS LOCATION
000776  000000                    HALT       ;EXAMINE STACK TO FIND CAUSE
```

```
                                        ;STANDARD INTERRUPT VECTORS


        000024                  .=24
000024  012354                          PFAIL                   ;POWER FAIL HANDLER
000026  000340                          340                     ;SERVICE AT LEVEL 7
000030  010552                          ERRORS                  ;ERROR HANDLER
000032  000340                          340                     ;SERVICE AT LEVEL 7
000034  010754                          TRPSRV                  ;GENERAL HANDLER DISPATCH SERVICE
000036  000340                          340                     ;SERVICE AT LEVEL 7
        000200                  .=200
000200  000167  000574                  JMP     START           ;GO TO START OF PROGRAM


                                        ;DEFINITIONS FOR TRAP SUBROUTINE CALLS
                                        ;POINTERS TO SUBROUTINES CAN BE FOUND STARTING
                                        ;AT LOCATION "TRPTAB"

        104400                          SCOPE=TRAP+Y            ;SCOPE LOOP AND ITERATION HANDLER
        104401                          TYPE=TRAP+Y             ;TELETYPE OUTPUT ROUTINE
        104402                          OCTASC=TRAP+Y           ;OCTAL TO ASCII CONVERSION
        104403                          INSTR=TRAP+Y            ;INPUT ASCII STRING
        104404                          INSTER=TRAP+Y           ;STRING INPUT ERROR
        104405                          PARAM=TRAP+Y            ;CONVERT STRING TO OCTAL, CHECK LIMITS
        104406                          SAVDSP=TRAP+Y           ;SAVE R0-R5, PC
        104407                          RESDS=TRAP+Y            ;RESTORE R0-R5
        104410                          SCOPE1=TRAP+Y           ;CHECK FOR FREEZE ON CURRENT DATA
```

```
                001000                  .=1000

                                                ;PROGRAM INITIALIZATION
                                                ;LOCK OUT INTERRUPTS
                                                ;SET UP PROCESSOR STACK
                                                ;SET UP POWER FAIL VECTOR
                                                ;CLEAR PROGRAM FLAGS AND COUNTS
                                                ;TYPE TITLE MESSAGE

001000  012767  000340  176770  START:  MOV     #340,PS                 ;LOCK OUT INTERRUPTS
001006  012706  013564                  MOV     #STACK,SP               ;SET UP PROCESSOR STACK
001012  012737  012354  000024          MOV     #PFAIL,@#24             ;SET UP POWER FAIL TRAP
001020  005067  010656                  CLR     STFLG                   ;CLEAR TEST START FLAG
001024  005067  010612                  CLR     PASCNT                  ;CLEAR PASS COUNT
001030  005067  010610                  CLR     ERRCNT                  ;CLEAR ERROR COUNT
001034  005067  010600                  CLR     ERRFLG                  ;CLEAR ERROR FLAG
001040  005067  010574                  CLR     ERRFLG                  ;CLEAR LAST ERROR PC
001044  104401  012520                  TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
001050  005767  010624                  TST     INIFLG                  ;CHECK INITIALIZATION FLAG
001054  001001                          BNE     VEC1                    ;IF NOT 0, CHECK SWITCHES
                                                                        ;FOR REINITIALIZATION
001056  000404                          BR      VEC2
001060  032767  000001  176502  VEC1:   BIT     #SW00,SWR               ;IF SW00=1, GET NEW VECTOR
001066  001445                          BEQ     BEGIN                   ;AND CSR
001070  012701  000300          VEC2:   MOV     #300,R1
001074  012702  000302                  MOV     #302,R2
001100  012703  000004                  MOV     #4,R3
001104  010211                  1$:     MOV     R2,(R1)                 ;RESTORE TRAPCATCHER
001106  005012                          CLR     (R2)                    ;IN FLOATING VECTOR AREA
001110  060301                          ADD     R3,R1
001112  060302                          ADD     R3,R2
001114  020127  001000                  CMP     R1,#1000
001120  001371                          BNE     1$
001122  104403                          INSTR                           ;INPUT ADDRESS OF DEVICE VECTOR
001124  012613                          MVECTOR                         ;MESSAGE "VECTOR ADDRESS-"
001126  104405                          PARAM                           ;CONVERT STRING TO OCTAL
001130  000300                          300                             ;LOW LIMIT
001132  000770                          770                             ;HIGH  LIMIT
001134  011630                          DHRVEC                          ;LOCATIONS TO BE FILLED
001136     003                  .BYTE   3                               ;NUMBER OF LOCATIONS
001137     004                  .BYTE   4                               ;LSB MASK
001140  104403                          INSTR                           ;INPUT ADDRESS OF DEVICE CSR
001142  012635                          MREGAD                          ;MESSAGE "CONTROL REGISTER ADDRESS-"
001144  104405                          PARAM                           ;CONVERT STRING TO OCTAL
001146  000000                          0                               ;LOW LIMIT
001150  177776                          177776                          ;HIGH LIMIT
001152  011606                          DHSCR                           ;LOCATIONS TO BE FILLED
001154     007                  .BYTE   7                               ;NUMBER OF LOCATIONS
001155     010                  .BYTE   10                              ;LSB MASK
001156  016767  010442  010442          MOV     DHSSR,DHSLR             ;SET UP ADDRESS OF SILO
001164  005267  010436                  INC     DHSLR                   ;STATUS REGISTER HIGH BYTE
001170  005767  010504                  TST     INIFLG                  ;IF INITIALIZATION FLAG
001174  001002                          BNE     BEGIN                   ;IS CLEARED
001176  005167  010476                  COM     INIFLG                  ;SET IT

                                        ;PROGRAM START
```

```
                                                ;CHECK FOR PROGRAM START AT SELECTED ADDRESS

001202  012767  000340  176566  BEGIN:  MOV     #340,PS                 ;LOCK OUT INTERRUPTS
001210  012706  013564                  MOV     #STACK,SP               ;SET UP PROCESSOR STACK
001214  032767  000002  176346          BIT     #SW01,SWR               ;IF SW01=1
001222  001410                          BEQ     1$                      ;GET PC FOR PROGRAM START
001224  104403                          INSTR                           ;GET PC
001226  013001                          MTSTPC                          ;MESSAGE "TEST PC"
001230  104405                          PARAM                           ;CONVERT STRING TO OCTAL
001232  000000                          0
001234  017500                          17500
001236  000207                          RETURN
001240     001                  .BYTE   1
001241     001                  .BYTE   1
001242  000410                          BR      2$
001244  012767  001274  010374  1$:     MOV     #T1,RETURN              ;NORMAL START, TEST 1
001252  005767  010424                  TST     STFLG                   ;IF LOOPING, BYPASS TYPEOUT
001256  001004                          BNE     3$
001260  005167  010416                  COM     STFLG
001264  104401  012775          2$:     TYPE    ,MR                     ;TYPE "R" TO INDICATE START
001270  000177  010352          3$:     JMP     @RETURN                 ;START TESTING
```

```
                                        ;RECEIVER PARITY LOGIC TEST ON LINE 0.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

001274  012767  000340  176474  T1:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
001302  012767  000010  010344          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
001310  012767  001434  010332          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
001316  012767  001364  010326          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
001324  012777  004000  010254          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
001332  012703  000000                  MOV     #0,R3   ;SET UP LINE #
001336  012705  130000                  MOV     #0*400+130000,R5        ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
001342  012767  000377  010340          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
001350  012777  000000  010230          MOV     #0,@DHSCR       ;SELECT LINE 0
001356  012777  073563  010226          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 0
001364  012777  177777  010224  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
001372  012777  011710  010214          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
001400  012777  000001  010214          MOV     #1,@DHBCR       ;SET BREAK FOR LINE 0.
001406  012777  000001  010204          MOV     #1,@DHBAR       ;START TRANSMITTER
001414  105777  010166          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
001420  100375                          BPL     2$              ;
001422  017701  010162                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
001426  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
001430  001401                          BEQ     .+4
001432  104003                          HLT     3       ;DATA ERROR
001434  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 1.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

001436  012767  000340  176332  T2:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
001444  012767  000010  010202          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
001452  012767  001576  010170          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
001460  012767  001526  010164          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
001466  012777  004000  010112          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
001474  012703  000001                  MOV     #1,R3   ;SET UP LINE #
001500  012705  130400                  MOV     #1*400+130000,R5        ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
001504  012767  000377  010176          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
```

```
001512  012777  000001  010066          MOV     #1,@DHSCR          ;SELECT LINE 1
001520  012777  073563  010064          MOV     #73563,@DHLPR      ;SELECT 8 BITS/CHAR,ODD PARITY HDX.
                                                                   ;9600 BAUD SPEED FOR LINE 1
001526  012777  177777  010062  1$:     MOV     #-1,@DHBC          ;TRANSMIT 1 CHARACTER
001534  012777  011710  010052          MOV     #TDATA,@DHBA       ;ADDRESS OF TRANSMITTED DATA
001542  012777  000002  010052          MOV     #2,@DHBCR          ;SET BREAK FOR LINE 1.
001550  012777  000002  010042          MOV     #2,@DHBAR          ;START TRANSMITTER
001556  105777  010024          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
001562  100375                          BPL     2$                 ;
001564  017701  010020                  MOV     @DHNRC,R1          ;GET RECEIVED CHARACTER
001570  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
001572  001401                          BEQ     .+4
001574  104003                          HLT     3       ;DATA ERROR
001576  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 2.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

001600  012767  000340  176170  T3:     MOV     #340,PS                    ;DISABLE ALL INTERRUPTS
001606  012767  000010  010040          MOV     #10,ICOUNT                 ;SET UP FOR 10 ITERATIONS
001614  012767  001740  010026          MOV     #4$,ESCAPE                 ;SET UP TO ESCAPE TO NEXT TEST
001622  012767  001670  010022          MOV     #1$,FREEZ1                 ;SET UP TO LOOP WITH DATA
001630  012777  004000  007750          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
001636  012703  000002                  MOV     #2,R3   ;SET UP LINE #
001642  012705  131000                  MOV     #2*400+130000,R5           ;SET EXPECTED LINE NUMBER
                                                                           ;VALID DATA FLAG,
                                                                           ;PARITY ERROR,DATA OVERRUN,
                                                                           ;AND EXPECTED DATA
001646  012767  000377  010034          MOV     #377,TDATA         ;ACTUAL DATA TO BE TRANSMITTED
001654  012777  000002  007724          MOV     #2,@DHSCR          ;SELECT LINE 2
001662  012777  073563  007722          MOV     #73563,@DHLPR      ;SELECT 8 BITS/CHAR,ODD PARITY HDX.
                                                                   ;9600 BAUD SPEED FOR LINE 2
001670  012777  177777  007720  1$:     MOV     #-1,@DHBC          ;TRANSMIT 1 CHARACTER
001676  012777  011710  007710          MOV     #TDATA,@DHBA       ;ADDRESS OF TRANSMITTED DATA
001704  012777  000004  007710          MOV     #4,@DHBCR          ;SET BREAK FOR LINE 2.
001712  012777  000004  007700          MOV     #4,@DHBAR          ;START TRANSMITTER
001720  105777  007662          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
001724  100375                          BPL     2$                 ;
001726  017701  007656                  MOV     @DHNRC,R1          ;GET RECEIVED CHARACTER
001732  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
001734  001401                          BEQ     .+4
001736  104003                          HLT     3       ;DATA ERROR
001740  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 3.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
```

```
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

001742  012767  000340  176026  T4:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
001750  012767  000010  007676          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
001756  012767  002102  007664          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
001764  012767  002032  007660          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
001772  012777  004000  007606          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
002000  012703  000003                  MOV     #3,R3   ;SET UP LINE #
002004  012705  131400                  MOV     #3*400+130000,R5        ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
002010  012767  000377  007672          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
002016  012777  000003  007562          MOV     #3,@DHSCR       ;SELECT LINE 3
002024  012777  073563  007560          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 3
002032  012777  177777  007556  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
002040  012777  011710  007546          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
002046  012777  000010  007546          MOV     #10,@DHBCR      ;SET BREAK FOR LINE 3.
002054  012777  000010  007536          MOV     #10,@DHBAR      ;START TRANSMITTER
002062  105777  007520          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
002066  100375                          BPL     2$              ;
002070  017701  007514                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
002074  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
002076  001401                          BEQ     .+4
002100  104003                          HLT     3       ;DATA ERROR
002102  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 4.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

002104  012767  000340  175564  T5:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002112  012767  000010  007534          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002120  012767  002244  007522          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
002126  012767  002174  007516          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
002134  012777  004000  007444          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
002142  012703  000004                  MOV     #4,R3   ;SET UP LINE #
002146  012705  132000                  MOV     #4*400+130000,R5        ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
002152  012767  000377  007530          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
002160  012777  000004  007420          MOV     #4,@DHSCR       ;SELECT LINE 4
002166  012777  073563  007416          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 4
002174  012777  177777  007414  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
002202  012777  011710  007404          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
002210  012777  000020  007404          MOV     #20,@DHBCR      ;SET BREAK FOR LINE 4.
002216  012777  000020  007374          MOV     #20,@DHBAR      ;START TRANSMITTER
```

```
002224  105777  007356          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
002230  100375                          BPL     2$              ;
002232  017701  007352                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
002236  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
002240  001401                          BEQ     .+4
002242  104003                          HLT     3       ;DATA ERROR
002244  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 5.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

002246  012767  000340  175522  T6:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002254  012767  000010  007372          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002262  012767  002406  007360          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
002270  012767  002336  007354          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
002276  012777  004000  007302          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
002304  012703  000005                  MOV     #5,R3   ;SET UP LINE #
002310  012705  132400                  MOV     #5*400+130000,R5        ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
002314  012767  000377  007366          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
002322  012777  000005  007256          MOV     #5,@DHSCR       ;SELECT LINE 5
002330  012777  073563  007254          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 5
002336  012777  177777  007252  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
002344  012777  011710  007242          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
002352  012777  000040  007242          MOV     #40,@DHBCR      ;SET BREAK FOR LINE 5.
002360  012777  000040  007232          MOV     #40,@DHBAR      ;START TRANSMITTER
002366  105777  007214          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
002372  100375                          BPL     2$              ;
002374  017701  007210                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
002400  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
002402  001401                          BEQ     .+4
002404  104003                          HLT     3       ;DATA ERROR
002406  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 6.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

002410  012767  000340  175360  T7:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002416  012767  000010  007230          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002424  012767  002550  007216          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
002432  012767  002500  007212          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
```

```
002440  012777  004000  007140          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
002446  012703  000006                  MOV     #6,R3   ;SET UP LINE #
002452  012705  133000                  MOV     #6*400+130000,R5        ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
002456  012767  000377  007224          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
002464  012777  000006  007114          MOV     #6,@DHSCR       ;SELECT LINE 6
002472  012777  073563  007112          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 6
002500  012777  177777  007110  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
002506  012777  011710  007100          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
002514  012777  000100  007100          MOV     #100,@DHBCR     ;SET BREAK FOR LINE 6.
002522  012777  000100  007070          MOV     #100,@DHBAR     ;START TRANSMITTER
002530  105777  007052          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
002534  100375                          BPL     2$              ;
002536  017701  007046                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
002542  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
002544  001401                          BEQ     .+4
002546  104003                          HLT     3       ;DATA ERROR
002550  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 7.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

002552  012767  000340  175216  T10:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002560  012767  000010  007066          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002566  012767  002712  007054          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
002574  012767  002642  007050          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
002602  012777  004000  006776          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
002610  012703  000007                  MOV     #7,R3   ;SET UP LINE #
002614  012705  133400                  MOV     #7*400+130000,R5        ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
002620  012767  000377  007062          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
002626  012777  000007  006752          MOV     #7,@DHSCR       ;SELECT LINE 7
002634  012777  073563  006750          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 7
002642  012777  177777  006746  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
002650  012777  011710  006736          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
002656  012777  000200  006736          MOV     #200,@DHBCR     ;SET BREAK FOR LINE 7.
002664  012777  000200  006726          MOV     #200,@DHBAR     ;START TRANSMITTER
002672  105777  006710          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
002676  100375                          BPL     2$              ;
002700  017701  006704                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
002704  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
002706  001401                          BEQ     .+4
002710  104003                          HLT     3       ;DATA ERROR
002712  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP
```

```
                                        ;RECEIVER PARITY LOGIC TEST ON LINE 10.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

002714  012767  000340  175054  T11:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
002722  012767  000010  006724          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
002730  012767  003054  006712          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
002736  012767  003004  006706          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
002744  012777  004000  006634          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
002752  012703  000010                  MOV     #10,R3  ;SET UP LINE #
002756  012705  134000                  MOV     #10*400+130000,R5       ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
002762  012767  000377  006720          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
002770  012777  000010  006610          MOV     #10,@DHSCR      ;SELECT LINE 10
002776  012777  073563  006606          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 10
003004  012777  177777  006604  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
003012  012777  011710  006574          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
003020  012777  000400  006574          MOV     #400,@DHBCR     ;SET BREAK FOR LINE 10.
003026  012777  000400  006564          MOV     #400,@DHBAR     ;START TRANSMITTER
003034  105777  006546          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
003040  100375                          BPL     2$              ;
003042  017701  006542                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
003046  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
003050  001401                          BEQ     .+4
003052  104003                          HLT     3       ;DATA ERROR
003054  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 11.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

003056  012767  000340  174712  T12:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
003064  012767  000010  006562          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
003072  012767  003216  006550          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
003100  012767  003146  006544          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
003106  012777  004000  006472          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
003114  012703  000011                  MOV     #11,R3  ;SET UP LINE #
003120  012705  134400                  MOV     #11*400+130000,R5       ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
003124  012767  000377  006556          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
```

```
003132  012777  000011  006446          MOV     #11,@DHSCR        ;SELECT LINE 11
003140  012777  073563  006444          MOV     #73563,@DHLPR     ;SELECT 8 BITS/CHAR,C . PARITY HDX,
                                                                  ;9600 BAUD SPEED FOR . . 11
003146  012777  177777  006442  1$:     MOV     #-1,@DHBC         ;TRANSMIT 1 CHARACTER
003154  012777  011710  006432          MOV     #TDATA,@DHBA      ;ADDRESS OF TRANSMITTED DATA
003162  012777  001000  006432          MOV     #1000,@DHBCR      ;SET BREAK FOR LINE 11.
003170  012777  001000  006422          MOV     #1000,@DHBAR      ;START TRANSMITTER
003176  105777  006404          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
003202  100375                          BPL     2$                ;
003204  017701  006400                  MOV     @DHNRC,R1         ;GET RECEIVED CHARACTER
003210  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
003212  001401                          BEQ     .+4
003214  104003                          HLT     3       ;DATA ERROR
003216  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 12.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

003220  012767  000340  174550  T13:     MOV     #340,PS                   ;DISABLE ALL INTERRUPTS
003226  012767  000010  006420          MOV     #10.,ICOUNT               ;SET UP FOR 10 ITERATIONS
003234  012767  003360  006406          MOV     #4$,ESCAPE                ;SET UP TO ESCAPE TO NEXT TEST
003242  012767  003310  006402          MOV     #1$,FREEZI                ;SET UP TO LOOP WITH DATA
003250  012777  004000  006330          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
003256  012703  000012                  MOV     #12,R3  ;SET UP LINE #
003262  012705  135000                  MOV     #12*400+130000,R5         ;SET EXPECTED LINE NUMBER
                                                                          ;VALID DATA FLAG,
                                                                          ;PARITY ERROR,DATA OVERRUN,
                                                                          ;AND EXPECTED DATA
003266  012767  000377  006414          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
003274  012777  000012  006304          MOV     #12,@DHSCR      ;SELECT LINE 12
003302  012777  073563  006302          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 12
003310  012777  177777  006300  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
003316  012777  011710  006270          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
003324  012777  002000  006270          MOV     #2000,@DHBCR    ;SET BREAK FOR LINE 12.
003332  012777  002000  006260          MOV     #2000,@DHBAR    ;START TRANSMITTER
003340  105777  006242          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
003344  100375                          BPL     2$              ;
003346  017701  006236                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
003352  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
003354  001401                          BEQ     .+4
003356  104003                          HLT     3       ;DATA ERROR
003360  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 13.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
```

```
                                                ;CHARACTER LENGTH: 8 BITS
                                                ;LINE SPEED: 9600 BAUD

003362  012767  000340  174406  T14:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
003370  012767  000010  006256          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
003376  012767  003522  006244          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
003404  012767  003452  006240          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
003412  012777  004000  006166          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
003420  012703  000013                  MOV     #13,R3  ;SET UP LINE #
003424  012705  135400                  MOV     #13*400+130000,R5       ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
003430  012767  000377  006252          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
003436  012777  000013  006142          MOV     #13,@DHSCR      ;SELECT LINE 13
003444  012777  073563  006140          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 13
003452  012777  177777  006136  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
003460  012777  011710  006126          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
003466  012777  004000  006126          MOV     #4000,@DHBCR    ;SET BREAK FOR LINE 13.
003474  012777  004000  006116          MOV     #4000,@DHBAR    ;START TRANSMITTER
003502  105777  006100          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
003506  100375                          BPL     2$              ;
003510  017701  006074                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
003514  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
003516  001401                          BEQ     .+4
003520  104003                          HLT     3       ;DATA ERROR
003522  104400          4$:             SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                                ;RECEIVER PARITY LOGIC TEST ON LINE 14.
                                                ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                                ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                                ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                                ;WITH EVEN PARITY AND THEREBY
                                                ;CAUSE A PARITY ERROR AND OVERRUN
                                                ;CHARACTER LENGTH: 8 BITS
                                                ;LINE SPEED: 9600 BAUD

003524  012767  000340  174244  T15:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
003532  012767  000010  006114          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
003540  012767  003664  006102          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
003546  012767  003614  006076          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
003554  012777  004000  006024          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
003562  012703  000014                  MOV     #14,R3  ;SET UP LINE #
003566  012705  136000                  MOV     #14*400+130000,R5       ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
003572  012767  000377  006110          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
003600  012777  000014  006000          MOV     #14,@DHSCR      ;SELECT LINE 14
003606  012777  073563  005776          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 14
003614  012777  177777  005774  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
003622  012777  011710  005764          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
003630  012777  010000  005764          MOV     #10000,@DHBCR   ;SET BREAK FOR LINE 14.
003636  012777  010000  005754          MOV     #10000,@DHBAR   ;START TRANSMITTER
```

```
003644  105777  005736          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
003650  100375                          BPL     2$              ;
003652  017701  005732                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
003656  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
003660  001401                          BEQ     .+4
003662  104003                          HLT     3       ;DATA ERROR
003664  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 15.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

003666  012767  000340  174102  T16:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
003674  012767  000010  005752          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
003702  012767  004026  005740          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
003710  012767  003756  005734          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
003716  012777  004000  005662          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
003724  012703  000015                  MOV     #15,R3  ;SET UP LINE #
003730  012705  136400                  MOV     #15*400+130000,R5       ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
003734  012767  000377  005746          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
003742  012777  000015  005636          MOV     #15,@DHSCR      ;SELECT LINE 15
003750  012777  073563  005634          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 15
003756  012777  177777  005632  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
003764  012777  011710  005622          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
003772  012777  020000  005622          MOV     #20000,@DHBCR   ;SET BREAK FOR LINE 15.
004000  012777  020000  005612          MOV     #20000,@DHBAR   ;START TRANSMITTER
004006  105777  005574          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
004012  100375                          BPL     2$              ;
004014  017701  005570                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
004020  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
004022  001401                          BEQ     .+4
004024  104003                          HLT     3       ;DATA ERROR
004026  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 16.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

004030  012767  000340  173740  T17:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
004036  012767  000010  005610          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
004044  012767  004170  005576          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
004052  012767  004120  005572          MOV     #1$,FREEZ1              ;SET UP TO LOOP WITH DATA
```

```
004060  012777  004000  005520          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
004066  012703  000016                  MOV     #16,R3  ;SET UP LINE #
004072  012705  137000                  MOV     #16*400+130000,R5       ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
004076  012767  000377  005604          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
004104  012777  000016  005474          MOV     #16,@DHSCR      ;SELECT LINE 16
004112  012777  073563  005472          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 16
004120  012777  177777  005470  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
004126  012777  011710  005460          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
004134  012777  040000  005460          MOV     #40000,@DHBCR   ;SET BREAK FOR LINE 16.
004142  012777  040000  005450          MOV     #40000,@DHBAR   ;START TRANSMITTER
004150  105777  005432          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
004154  100375                          BPL     2$              ;
004156  017701  005426                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
004162  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
004164  001401                          BEQ     .+4
004166  104003                          HLT     3       ;DATA ERROR
004170  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP

                                        ;RECEIVER PARITY LOGIC TEST ON LINE 17.
                                        ;TRANSMIT ONE CHARACTER TO THE RECEIVER
                                        ;WITH ODD PARITY SET,BREAK SET,HDX SET.
                                        ;THE RECEIVER WILL RECEIVE AN ALL 0'S CHARACTER
                                        ;WITH EVEN PARITY AND THEREBY
                                        ;CAUSE A PARITY ERROR AND OVERRUN
                                        ;CHARACTER LENGTH: 8 BITS
                                        ;LINE SPEED: 9600 BAUD

004172  012767  000340  173576  T20:     MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
004200  012767  000010  005446          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
004206  012767  004332  005434          MOV     #4$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
004214  012767  004262  005430          MOV     #1$,FREEZI              ;SET UP TO LOOP WITH DATA
004222  012777  004000  005356          MOV     #BIT11,@DHSCR   ;MASTER CLEAR INTERFACE
004230  012703  000017                  MOV     #17,R3  ;SET UP LINE #
004234  012705  137400                  MOV     #17*400+130000,R5       ;SET EXPECTED LINE NUMBER
                                                                        ;VALID DATA FLAG,
                                                                        ;PARITY ERROR,DATA OVERRUN,
                                                                        ;AND EXPECTED DATA
004240  012767  000377  005442          MOV     #377,TDATA      ;ACTUAL DATA TO BE TRANSMITTED
004246  012777  000017  005332          MOV     #17,@DHSCR      ;SELECT LINE 17
004254  012777  073563  005330          MOV     #73563,@DHLPR   ;SELECT 8 BITS/CHAR,ODD PARITY HDX,
                                                                ;9600 BAUD SPEED FOR LINE 17
004262  012777  177777  005326  1$:     MOV     #-1,@DHBC       ;TRANSMIT 1 CHARACTER
004270  012777  011710  005316          MOV     #TDATA,@DHBA    ;ADDRESS OF TRANSMITTED DATA
004276  012777  100000  005316          MOV     #100000,@DHBCR  ;SET BREAK FOR LINE 17.
004304  012777  100000  005306          MOV     #100000,@DHBAR  ;START TRANSMITTER
004312  105777  005270          2$:     TSTB    @DHSCR  ;WAIT FOR CHARACTER TO BE RECEIVED
004316  100375                          BPL     2$              ;
004320  017701  005264                  MOV     @DHNRC,R1       ;GET RECEIVED CHARACTER
004324  020501                          CMP     R5,R1   ;COMPARE EXPECTED AND RECEIVED DATA
004326  001401                          BEQ     .+4
004330  104003                          HLT     3       ;DATA ERROR
004332  104400                  4$:     SCOPE   ;CHECK FOR ITERATIONS,LOOP
```

```
                                                ;MULTI-LINE PARITY DATA TEST
                                                ;TRANSMIT A BINARY COUNT WITH ODD PARITY
                                                ;PATTERN ON ALL LINES.
                                                ;SILO ALARM LEVEL SET TO 0
                                                ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                                ;IN INTERRUPT MODE.
                                                ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                                ;LINE SPEED: 2400 BAUD
                                                ;CHARACTER LENGTH: 8 + PARITY

004334  012767  000340  173434  T21:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
004342  012767  000010  005304          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
004350  012767  004602  005272          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
004356  012777  004000  005222          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
004364  012700  000020                  MOV     #16.,R0                 ;SET UP TO START 16. LINES
004370  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
004372  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
004376  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
004402  012777  011712  005204  1$:     MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
004410  012777  177400  005200          MOV     #-400,@DHBC             ;LOAD BYTE COUNT
004416  012777  027363  005166          MOV     #27363,@DHLPR           ;SET LINE SPEED AND
                                                        ;CHARACTER LENGTH 8 + ODD PARITY
004424  105061  012312                  CLPB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
004430  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
004434  005277  005146                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
004440  005202                          INC     R2                      ;UPDATE POINTER TO
004442  062701  000002                  ADD     #2,R1                   ;LOW AND HIGH BYTE OF
004446  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
004452  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
004454  001352                          BNE     1$                      ;NOT SET UP
004456  012777  004516  005144          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
004464  012777  000240  005140          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
004472  012777  000100  005106          MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
004500  012777  177777  005112          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
004506  005067  173264                  CLR     PS                      ;ENABLE INTERRUPTS
004512  000167  000076                  JMP     12$                     ;GO  O TRANSMITTER WAITING ROUTINE

                                                ;RECEIVER INTERRUPT SERVICE ROUTINE
                                                ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                                ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                                ;VERIFY THAT RECEIVED DATA IS CORRECT
                                                ;CHECK FOR END OF PASS

004516  105777  005064          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
004522  100026                          BPL     10$
004524  032777  040000  005054          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
004532  001401                          BEQ     .+4
004534  104002                          HLT     2                       ;SILO OVERRUN, ERROR
004536  017701  005046          7$:     MOV     @DHNRC,R1               ;READ CHARACTER FROM SILO
004542  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
004544  000302                          SWPB    R2
004546  042702  177760                  BIC     #177760,R2
004552  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
004554  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
```

```
004556  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
004562  001403                          BEQ     .+10
004564  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
004570  104003                          HLT     3                       ;DATA ERROR
004572  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
004576  000002                          RTI                             ;CONTINUE
004600  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
004602  022626                  11$:    POP2SP                          ;RESTORE STACK
004604  012777  004030  004774          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
004612  000411                          BR      13$                     ;RESTART TEST


                                        ;WAIT FOR ALL BAR BITS TO CLEAR
                                        ;WHEN ALL BAR BITS HAVE CLEARED,
                                        ;WAIT FOR SILO TO EMPTY
                                        ;WHEN SILO IS EMPTY, SCOPE ON TEST

004614  005777  005000          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
004620  001375                          BNE     .-4
004622  105777  005000                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
004626  001375                          BNE     .-4
004630  012767  000340  173140          MOV     #340,PS ;PREVENT INTERRUPTS
004636  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                        ;MULTI-LINE PARITY DATA TEST
                                        ;TRANSMIT A BINARY COUNT WITH EVEN PARITY
                                        ;PATTERN ON ALL LINES.
                                        ;SILO ALARM LEVEL SET TO 0
                                        ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                        ;IN INTERRUPT MODE.
                                        ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                        ;LINE SPEED: 2400 BAUD
                                        ;CHARACTER LENGTH: 8 + PARITY

004640  012767  000340  173130  T22:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
004646  012767  000010  005000          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
004654  012767  005106  004756          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
004662  012777  004000  004716          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
004670  012700  000020                  MOV     #16.,R0                 ;SET UP TO START 16. LINES
004674  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
004676  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
004702  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
004706  012777  011712  004700  1$:     MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
004714  012777  177400  004674          MOV     #-400,@DHBC             ;LOAD BYTE COUNT
004722  012777  027323  004662          MOV     #27323,@DHLPR           ;SET LINE SPEED AND
                                                                ;CHARACTER LENGTH 8 + EVEN PARITY
004730  105061  012312                  CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
004734  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
004740  005277  004642                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
004744  005202                          INC     R2                      ;UPDATE POINTER TO
004746  062701  000002                  ADD     #2,R1                   ;LOW AND HIGH BYTE OF
004752  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
004756  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
004760  001352                          BNE     1$                      ;NOT SET UP
004762  012777  005022  004640          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
```

```
004770  012777  000240  004634          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
004776  012777  000100  004602          MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
005004  012777  177777  004606          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
005012  005067  172760                  CLR     PS                      ;ENABLE INTERRUPTS
005016  000167  000076                  JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE

                                        ;RECEIVER INTERRUPT SERVICE ROUTINE
                                        ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                        ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                        ;VERIFY THAT RECEIVED DATA IS CORRECT
                                        ;CHECK FOR END OF PASS

005022  105777  004560          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
005026  100026                          BPL     10$
005030  032777  040000  004550          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
005036  001401                          BEQ     .+4
005040  104002                          HLT     2                       ;SILO OVERRUN, ERROR
005042  017701  004542          7$:     MOV     @DHNRC,R1               ;READ CHARACTER FROM SILO
005046  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
005050  000302                          SWAB    R2
005052  042702  177760                  BIC     #177760,R2
005056  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
005060  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
005062  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
005066  001403                          BEQ     .+10
005070  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
005074  104003                          HLT     3                       ;DATA ERROR
005076  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
005102  000002                          RTI                             ;CONTINUE
005104  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
005106  022626                  11$:    POP2SP                          ;RESTORE STACK
005110  012777  004000  004470          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
005116  000411                          BR      13$                     ;RESTART TEST


                                        ;WAIT FOR ALL BAR BITS TO CLEAR
                                        ;WHEN ALL BAR BITS HAVE CLEARED,
                                        ;WAIT FOR SILO TO EMPTY
                                        ;WHEN SILO IS EMPTY, SCOPE ON TEST

005120  005777  004474          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
005124  001375                          BNE     .-4
005126  105777  004474                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
005132  001375                          BNE     .-4
005134  012767  000340  172634          MOV     #340,PS ;PREVENT INTERRUPTS
005142  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                        ;MULTI-LINE PARITY DATA TEST
                                        ;TRANSMIT A BINARY COUNT WITH ODD PARITY
                                        ;PATTERN ON ALL LINES.
                                        ;SILO ALARM LEVEL SET TO 0
                                        ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                        ;IN INTERRUPT MODE.
                                        ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                        ;LINE SPEED: 2400 BAUD
                                        ;CHARACTER LENGTH: 7 + PARITY
```

```
1177
1178  005144  012767  000340  172624  T23:   MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
1179  005152  012767  000010  004474         MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
1180  005160  012767  005412  004462         MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
1181  005166  012777  004000  004412         MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1182  005174  012700  000020                 MOV     #16.,R0                 ;SET UP TO START 16. LINES
1183  005200  005001                         CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
1184                                                                         ;AND RECEIVED DATA STORAGE
1185  005202  012702  000200                 MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
1186  005206  012703  000001                 MOV     #1,R3                   ;OF EXPECTED DATA
1187  005212  012777  011712  004374  1$:    MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
1188  005220  012777  177600  004370         MOV     #-200,@DHBC             ;LOAD BYTE COUNT
1189  005226  012777  027362  004356         MOV     #27362,@DHLPR           ;SET LINE SPEED AND
1190                                                         ;CHARACTER LENGTH 7 + ODD PARITY
1191  005234  105061  012312                 CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
1192  005240  110263  012312                 MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
1193  005244  005277  004336                 INC     @DHSCR                  ;ADVANCE TO NEXT LINE
1194  005250  005202                         INC     R2                      ;UPDATE POINTER TO
1195  005252  062701  000002                 ADD     #2,R1                   ;LOW AND HIGH BYTE OF
1196  005256  062703  000002                 ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
1197  005262  005300                         DEC     R0                      ;CONTINUE IF ALL LINES
1198  005264  001352                         BNE     1$                      ;NOT SET UP
1199  005266  012777  005326  004334         MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
1200  005274  012777  000240  004330         MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
1201  005302  012777  000100  004276         MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
1202  005310  012777  177777  004302         MOV     #-1,@DHBAR              ;SET ALL BAR BITS
1203  005316  005067  172454                 CLR     PS                      ;ENABLE INTERRUPTS
1204  005322  000167  000076                 JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE
1205
1206                                         ;RECEIVER INTERRUPT SERVICE ROUTINE
1207                                         ;CHECK FOR RECEIVER DONE AND NO ERRORS
1208                                         ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
1209                                         ;VERIFY THAT RECEIVED DATA IS CORRECT
1210                                         ;CHECK FOR END OF PASS
1211
1212  005326  105777  004254          6$:    TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
1213  005332  100026                         BPL     10$
1214  005334  032777  040000  004244         BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
1215  005342  001401                         BEQ     .+4
1216  005344  104002                         HLT     2                       ;SILO OVERRUN, ERROR
1217  005346  017701  004236          7$:    MOV     @DHNRC,R1               ;READ CHARACTER FROM SILO
1218  005352  010102                         MOV     R1,R2                   ;EXTRACT LINE NUMBER
1219  005354  000302                         SWAB    R2
1220  005356  042702  177760                 BIC     #177760,R2
1221  005362  010203                         MOV     R2,R3                   ;SAVE LINE NUMBER
1222  005364  006302                         ASL     R2                      ;USE LINE NUMBER AS OFFSET
1223  005366  026201  012312                 CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
1224  005372  001403                         BEQ     .+10
1225  005374  016205  012312                 MOV     RBUF(R2),R5             ;GET EXPECTED DATA
1226  005400  104003                         HLT     3                       ;DATA ERROR
1227  005402  105262  012312                 INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
1228  005406  000002                         RTI                             ;CONTINUE
1229  005410  104001                  10$:   HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
1230  005412  022626                  11$:   POP2SP                          ;RESTORE STACK
1231  005414  012777  004000  004164         MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
1232  005422  000411                         BR      13$                     ;RESTART TEST
```

```
                                                ;WAIT FOR ALL BAR BITS TO CLEAR
                                                ;WHEN ALL BAR BITS HAVE CLEARED,
                                                ;WAIT FOR SILO TO EMPTY
                                                ;WHEN SILO IS EMPTY, SCOPE ON TEST

005424  005777  004170          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
005430  001375                          BNE     .-4
005432  105777  004170                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
005436  001375                          BNE     .-4
005440  012767  000340  172330          MOV     #340,PS ;PREVENT INTERRUPTS
005446  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                                ;MULTI-LINE PARITY DATA TEST
                                                ;TRANSMIT A BINARY COUNT WITH EVEN PARITY
                                                ;PATTERN ON ALL LINES.
                                                ;SILO ALARM LEVEL SET TO 0
                                                ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                                ;IN INTERRUPT MODE.
                                                ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                                ;LINE SPEED: 2400 BAUD
                                                ;CHARACTER LENGTH: 7 + PARITY

005450  012767  000340  172320  T24:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
005456  012767  000010  004170          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
005464  012767  005716  004156          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
005472  012777  004000  004106          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
005500  012700  000020                  MOV     #16.,R0                 ;SET UP TO START 16. LINES
005504  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
005506  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
005512  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
005516  012777  011712  004070  1$:     MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
005524  012777  177600  004064          MOV     #-200,@DHBC             ;LOAD BYTE COUNT
005532  012777  027322  004052          MOV     #27322,@DHLPR           ;SET LINE SPEED AND
                                                        ;CHARACTER LENGTH 7 + EVEN PARITY
005540  105061  012312                  CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
005544  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
005550  005277  004032                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
005554  005202                          INC     R2                      ;UPDATE POINTER TO
005556  062701  000002                  ADD     #2,R1                   ;LOW AND HIGH BYTE OF
005562  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
005566  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
005570  001352                          BNE     1$                      ;NOT SET UP
005572  012777  005632  004030          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
005600  012777  000240  004024          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
005606  012777  000100  003772          MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
005614  012777  177777  003776          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
005622  005067  172150                  CLR     PS                      ;ENABLE INTERRUPTS
005626  000167  000076                  JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE

                                                ;RECEIVER INTERRUPT SERVICE ROUTINE
                                                ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                                ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                                ;VERIFY THAT RECEIVED DATA IS CORRECT
```

```
                                                ;CHECK FOR END OF PASS

005632  105777  003760          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
005636  100026                          BPL     10$
005640  032777  040000  003740          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, EPROR
005646  001401                          BEQ     .+4
005650  104002                          HLT     2                       ;SILO OVERRUN, ERROR
005652  017701  003732          7$:     MOV     @DHNRC,R1               ;PEAD CHARACTER FROM SILO
005656  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
005660  000302                          SWAB    R2
005662  042702  177760                  BIC     #177760,R2
005666  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
005670  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
005672  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
005676  001403                          BEQ     .+10
005700  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
005704  104003                          HLT     3                       ;DATA ERROR
005706  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
005712  000002                          RTI                             ;CONTINUE
005714  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
005716  022626                  11$:    POP2SP                          ;RESTORE STACK
005720  012777  004000  003660          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
005726  000411                          BR      13$                     ;RESTART TEST


                                                ;WAIT FOR ALL BAR BITS TO CLEAR
                                                ;WHEN ALL BAR BITS HAVE CLEARED,
                                                ;WAIT FOR SILO TO EMPTY
                                                ;WHEN SILO IS EMPTY, SCOPE ON TEST

005730  005777  003664          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
005734  001375                          BNE     .-4
005736  105777  003664                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
005742  001375                          BNE     .-4
005744  012767  000340  172024          MOV     #340,PS ;PREVENT INTERRUPTS
005752  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                                ;MULTI-LINE PARITY DATA TEST
                                                ;TRANSMIT A BINARY COUNT WITH ODD PARITY
                                                ;PATTERN ON ALL LINES.
                                                ;SILO ALARM LEVEL SET TO 0
                                                ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                                ;IN INTERRUPT MODE.
                                                ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                                ;LINE SPEED: 2400 BAUD
                                                ;CHARACTER LENGTH: 6 + PARITY

005754  012767  000340  172014  T25:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
005762  012767  000010  003664          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
005770  012767  006222  003652          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
005776  012777  004000  003602          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
006004  012700  000020                  MOV     #16.,R0                 ;SET UP TO START 16. LINES
006010  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
006012  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
006016  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
```

```
006022  012777  011712  003564  1$:     MOV     #TBUF,@DHBM             ;LOAD BUSS ADDRESS
006030  012777  177700  003560          MOV     #-100,@DHBC             ;LOAD BYTE COUNT
006036  012777  027361  003546          MOV     #27361,@DHLPR           ;SET LINE SPEED AND
                                                        ;CHARACTER LENGTH 6 + ODD PARITY
006044  105061  012312                  CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
006050  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
006054  005277  003526                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
006060  005202                          INC     R2                      ;UPDATE POINTER TO
006062  062701  000002                  ADD     #2,R1                   ;LOW AND HIGH BYTE OF
006066  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
006072  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
006074  001352                          BNE     1$                      ;NOT SET UP
006076  012777  006136  003524          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
006104  012777  000240  003520          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
006112  012777  000100  003466          MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
006120  012777  177777  003472          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
006126  005067  171644                  CLR     PS                      ;ENABLE INTERRUPTS
006132  000167  000076                  JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE

                                        ;RECEIVER INTERRUPT SERVICE ROUTINE
                                        ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                        ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                        ;VERIFY THAT RECEIVED DATA IS CORRECT
                                        ;CHECK FOR END OF PASS

006136  105777  003444          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
006142  100026                          BPL     10$
006144  032777  040000  003434          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
006152  001401                          BEQ     .+4
006154  104002                          HLT     2                       ;SILO OVERRUN, ERROR
006156  017701  003426          7$:     MOV     @DHNRC,R1               ;READ CHARACTER FROM SILO
006162  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
006164  000302                          SWAB    R2
006166  042702  177760                  BIC     #177760,R2
006172  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
006174  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
006176  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
006202  001403                          BEQ     .+10
006204  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
006210  104003                          HLT     3                       ;DATA ERROR
006212  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
006216  000002                          RTI                             ;CONTINUE
006220  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
006222  022626                  11$:    POP2SP                          ;RESTORE STACK
006224  012777  004000  003354          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
006232  000411                          BR      13$                     ;RESTART TEST


                                        ;WAIT FOR ALL BAR BITS TO CLEAR
                                        ;WHEN ALL BAR BITS HAVE CLEARED,
                                        ;WAIT FOR SILO TO EMPTY
                                        ;WHEN SILO IS EMPTY, SCOPE ON TEST

006234  005777  003360          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
006240  001375                          BNE     .-4
006242  105777  003360                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
```

```
006246  001375                          BNE     .-4
006250  012767  000340  171520          MOV     #340,PS ;PREVENT INTERRUPTS
006256  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                        ;MULTI-LINE PARITY DATA TEST
                                        ;TRANSMIT A BINARY COUNT WITH EVEN PARITY
                                        ;PATTERN ON ALL LINES.
                                        ;SILO ALARM LEVEL SET TO 0
                                        ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                        ;IN INTERRUPT MODE.
                                        ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                        ;LINE SPEED: 2400 BAUD
                                        ;CHARACTER LENGTH: 6 + PARITY

006260  012767  000340  171510  T26:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
006266  012767  000010  003360          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
006274  012767  006526  003346          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
006302  012777  004000  003276          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
006310  012700  000020                  MOV     #16.,R0                 ;SET UP TO START 16. LINES
006314  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
006316  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
006322  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
006326  012777  011712  003260  1$:     MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
006334  012777  177700  003254          MOV     #-100,@DHBC             ;LOAD BYTE COUNT
006342  012777  027321  003242          MOV     #27321,@DHLPR           ;SET LINE SPEED AND
                                                        ;CHARACTER LENGTH 6 + EVEN PARITY
006350  105061  012312                  CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
006354  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
006360  005277  003222                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
006364  005202                          INC     R2                      ;UPDATE POINTER TO
006366  062701  000002                  ADD     #2,R1                   ;LOW AND HIGH BYTE OF
006372  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
006376  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
006400  001352                          BNE     1$                      ;NOT SET UP
006402  012777  006442  003220          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
006410  012777  000240  003214          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
006416  012777  000100  003162          MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
006424  012777  177777  003166          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
006432  005067  171340                  CLR     PS                      ;ENABLE INTERRUPTS
006436  000167  000076                  JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE

                                        ;RECEIVER INTERRUPT SERVICE ROUTINE
                                        ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                        ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                        ;VERIFY THAT RECEIVED DATA IS CORRECT
                                        ;CHECK FOR END OF PASS

006442  105777  003140          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
006446  100026                          BPL     10$
006450  032777  040000  003130          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
006456  001401                          BEQ     .+4
006460  104002                          HLT     2                       ;SILO OVERRUN, ERROR
006462  017701  003122          7$:     MOV     @DHNRC,R1               ;READ CHARACTER FROM SILO
006466  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
006470  000302                          SWAB    R2
```

```
006472  042702  177760                  BIC     #177760,R2
006476  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
006500  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
006502  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
006506  001403                          BEQ     .+10
006510  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
006514  104003                          HLT     3                       ;DATA ERROR
006516  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
006522  000002                          RTI                             ;CONTINUE
006524  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
006526  022626                  11$:    POP2SP                          ;RESTORE STACK
006530  012777  004000  003050          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
006536  000411                          BR      13$                     ;RESTART TEST


                                        ;WAIT FOR ALL BAR BITS TO CLEAR
                                        ;WHEN ALL BAR BITS HAVE CLEARED,
                                        ;WAIT FOR SILO TO EMPTY
                                        ;WHEN SILO IS EMPTY, SCOPE ON TEST

006540  005777  003054          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
006544  001375                          BNE     .-4
006546  105777  003054                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
006552  001375                          BNE     .-4
006554  012767  000340  171214          MOV     #340,PS ;PREVENT INTERRUPTS
006562  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                        ;MULTI-LINE PARITY DATA TEST
                                        ;TRANSMIT A BINARY COUNT WITH ODD PARITY
                                        ;PATTERN ON ALL LINES.
                                        ;SILO ALARM LEVEL SET TO 0
                                        ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                        ;IN INTERRUPT MODE.
                                        ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                        ;LINE SPEED: 2400 BAUD
                                        ;CHARACTER LENGTH: 5 + PARITY

006564  012767  000340  171204  T27:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
006572  012767  000010  003054          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
006600  012767  007032  003042          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
006606  012777  004000  002772          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
006614  012700  000020                  MOV     #16.,R0                 ;SET UP TO START 16. LINES
006620  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
006622  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
006626  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
006632  012777  011712  002754  1$:     MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
006640  012777  177740  002750          MOV     #-40,@DHBC              ;LOAD BYTE COUNT
006646  012777  027360  002736          MOV     #27360,@DHLPR           ;SET LINE SPEED AND
                                                        ;CHARACTER LENGTH 5 + ODD PARITY
006654  105061  012312                  CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
006660  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
006664  005277  002716                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
006670  005202                          INC     R2                      ;UPDATE POINTER TO
006672  062701  000002                  ADD     #2,R1                   ;LOW AND HIGH BYTE OF
006676  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
```

```
006702  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
006704  001352                          BNE     1$                      ;NOT SET UP
006706  012777  006746  002714          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
006714  012777  000240  002710          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
006722  012777  000100  002656          MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
006730  012777  177777  002662          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
006736  005067  171034                  CLR     PS                      ;ENABLE INTERRUPTS
006742  000167  000076                  JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE

                                        ;RECEIVER INTERRUPT SERVICE ROUTINE
                                        ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                        ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                        ;VERIFY THAT RECEIVED DATA IS CORRECT
                                        ;CHECK FOR END OF PASS

006746  105777  002634          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
006752  100026                          BPL     10$
006754  032777  040000  002624          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
006762  001401                          BEQ     .+4
006764  104002                          HLT     2                       ;SILO OVERRUN, ERROR
006766  017701  002616          7$:     MOV     @DHNRC,R1               ;READ CHARACTER FROM SILO
006772  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
006774  000302                          SWAB    R2
006776  042702  177760                  BIC     #177760,R2
007002  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
007004  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
007006  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
007012  001403                          BEQ     .+10
007014  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
007020  104003                          HLT     3                       ;DATA ERROR
007022  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
007026  000002                          RTI                             ;CONTINUE
007030  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
007032  022626                  11$:    POP2SP                          ;RESTORE STACK
007034  012777  004000  002544          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
007042  000411                          BR      13$                     ;RESTART TEST


                                        ;WAIT FOR ALL BAR BITS TO CLEAR
                                        ;WHEN ALL BAR BITS HAVE CLEARED,
                                        ;WAIT FOR SILO TO EMPTY
                                        ;WHEN SILO IS EMPTY, SCOPE ON TEST

007044  005777  002550          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
007050  001375                          BNE     .-4
007052  105777  002550                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
007056  001375                          BNE     .-4
007060  012767  000340  170710          MOV     #340,PS ;PREVENT INTERRUPTS
007066  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP

                                        ;MULTI-LINE PARITY DATA TEST
                                        ;TRANSMIT A BINARY COUNT WITH EVEN PARITY
                                        ;PATTERN ON ALL LINES.
                                        ;SILO ALARM LEVEL SET TO 0
                                        ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                        ;IN INTERRUPT MODE.
```

```
                                                ;TRANSMITTER INTERRUPT WILL BE DISABLED
                                                ;LINE SPEED: 2400 BAUD
                                                ;CHARACTER LENGTH: 5 + PARITY

007070  012767  000340  170700  T30:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
007076  012767  000010  002550          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
007104  012767  007336  002536          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
007112  012777  004000  002466          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
007120  012700  000020                  MOV     #16.,R0                 ;SET UP TO START 16. LINES
007124  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
007126  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
007132  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
007136  012777  011712  002450  1$:     MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
007144  012777  177740  002444          MOV     #-40,@DHBC              ;LOAD BYTE COUNT
007152  012777  027320  002432          MOV     #27320,@DHLPR           ;SET LINE SPEED AND
                                                        ;CHARACTER LENGTH 5 + EVEN PARITY
007160  105061  012312                  CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
007164  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
007170  005277  002412                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
007174  005202                          INC     R2                      ;UPDATE POINTER TO
007176  062701  000002                  ADD     #2,R1                   ;LOW AND HIGH BYTE OF
007202  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
007206  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
007210  001352                          BNE     1$                      ;NOT SET UP
007212  012777  007252  002410          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
007220  012777  000240  002404          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
007226  012777  000100  002352          MOV     #100,@DHSCR             ;ENABLE RECEIVER INTERRUPT
007234  012777  177777  002356          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
007242  005067  170530                  CLR     PS                      ;ENABLE INTERRUPTS
007246  000167  000076                  JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE

                                                ;RECEIVER INTERRUPT SERVICE ROUTINE
                                                ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                                ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                                ;VERIFY THAT RECEIVED DATA IS CORRECT
                                                ;CHECK FOR END OF PASS

007252  105777  002330          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
007256  100026                          BPL     10$
007260  032777  040000  002320          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
007266  001401                          BEQ     .+4
007270  104002                          HLT     2                       ;SILO OVERRUN, ERROR
007272  017701  002312          7$:     MOV     @DHNRC,R1               ;READ CHARACTER FROM SILO
007276  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
007300  000302                          SWAB    R2
007302  042702  177760                  BIC     #177760,R2
007306  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
007310  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
007312  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
007316  001403                          BEQ     .+10
007320  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
007324  104003                          HLT     3                       ;DATA ERROR
007326  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
007332  000002                          RTI                             ;CONTINUE
007334  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
```

```
007336  022626                  11$:    POP2SP                          ;RESTORE STACK
007340  012777  004000  002240          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
007346  000411                          BR      13$                     ;RESTART TEST


                                        ;WAIT FOR ALL BAR BITS TO CLEAR
                                        ;WHEN ALL BAR BITS HAVE CLEARED,
                                        ;WAIT FOR SILO TO EMPTY
                                        ;WHEN SILO IS EMPTY, SCOPE ON TEST

007350  005777  002244          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
007354  001375                          BNE     .-4
007356  105777  002244                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
007362  001375                          BNE     .-4
007364  012767  000340  170404          MOV     #340,PS :PREVENT INTERRUPTS
007372  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                                ;MULTI LINE DATA TEST
                                                ;SILO ALARM LEVEL SET TO 0
                                                ;RECEIVER WILL BE SERVICED ON A PER CHARACTER BASIS
                                                ;IN INTERRUPT MODE
                                                ;TRANSMITTER INTERRUPTS WILL BE DISABLED
                                                ;TRANSMIT A BINARY COUNT PATTERN ON ALL LINES
                                                ;LINE SPEED IS 4800 BAUD FOR ALL LINES
                                                ;CHARACTER LENGTH IS 8 BITS FOR ALL LINES

007374  012767  000340  170374  T31:    MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
007402  012767  000010  002244          MOV     #10,ICOUNT              ;SET UP FOR 10 ITERATIONS
007410  012767  007642  002232          MOV     #11$,ESCAPE             ;SET UP TO ESCAPE TO NEXT TEST
007416  012777  004000  002162          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
007424  012700  000020                  MOV     #20,R0                  ;SET UP TO LOAD BYTE
007430  005001                          CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                        ;AND RECEIVED DATA STORAGE
007432  012702  000200                  MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
007436  012703  000001                  MOV     #1,R3                   ;OF EXPECTED DATA
007442  012777  011712  002144  1$:     MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
007450  012777  177400  002140          MOV     #-400,@DHBC             ;LOAD BYTE COUNT
007456  012777  031403  002126          MOV     #31403,@DHLPR           ;SET LINE SPEED AND
                                                                        ;CHARACTER LENGTH
007464  105061  012312                  CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
007470  110263  012312                  MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
007474  005277  002106                  INC     @DHSCR                  ;ADVANCE TO NEXT LINE
007500  005202                          INC     R2                      ;UPDATE POINTER TO
007502  062701  000002                  ADD     #2,R1                   ;LOW AND HIG BYTE OF
007506  062703  000002                  ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
007512  005300                          DEC     R0                      ;CONTINUE IF ALL LINES
007514  001352                          BNE     1$                      ;NOT SET UP
007516  012777  007556  002104          MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
007524  012777  000240  002100          MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
007532  012777  000100  002046          MOV     #100,@DHSCR             ;ENABLE INTERFACE
007540  012777  177777  002052          MOV     #-1,@DHBAR              ;SET ALL BAR BITS
007546  005067  170224                  CLR     PS                      ;ENABLE INTERRUPTS
007552  000167  000076                  JMP     12$                     ;GO TO TRANSMITTER WAITING ROUTINE

                                                ;RECEIVER INTERRUPT SERVICE ROUTINE
                                                ;CHECK FOR RECEIVER DONE AND NO ERRORS
                                                ;CHECK RECEIVED DATA FOR "VALID DATA" FLAG AND NO ERRORS
                                                ;VERIFY THAT RECEIVED DATA IS CORRECT
                                                ;CHECK FOR END OF PASS

007556  105777  002024          6$:     TSTB    @DHSCR                  ;IF CHARACTER AVAILABLE NOT SET, ERROR
007562  100026                          BPL     10$
007564  032777  040000  002014          BIT     #40000,@DHSCR           ;IF SILO OVERRUN SET, ERROR
007572  001401                          BEQ     .+4
007574  104002                          HLT     2                       ;SILO OVERRUN, ERROR
007576  017701  002006          7$:     MOV     @DHNRC,R1               ;READ CHARACTER FORM SILO
007602  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
007604  000302                          SWAB    R2
007606  042702  177760                  BIC     #177760,R2
007612  010203                          MOV     R2,R3                   ;SAVE LINE NUMBER
007614  006302                          ASL     R2                      ;USE LINE NUMBER AS OFFSET
007616  026201  012312                  CMP     RBUF(R2),R1             ;COMPARE EXPECTED AND RECEIVED DATA
```

```
007622  001403                          BEQ     .+10
007624  016205  012312                  MOV     RBUF(R2),,R5            ;GET EXPECTED DATA
007630  104003                          HLT     3                       ;DATA ERROR
007632  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED CHARACTER
007636  000002                          RTI                             ;CONTINUE
007640  104001                  10$:    HLT     1                       ;CHARACTER AVAILABLE NOT SET, ERROR
007642  022626                  11$:    POP2SP                          ;RESTORE STACK
007644  012777  004000  001734          MOV     #BIT11,@DHSCR           ;MASTER CLEAR INTERFACE
007652  000406                          BR      13$                     ;RESTART TEST


                                        ;WAIT FOR ALL BAR BITS TO CLEAR
                                        ;WHEN ALL BAR BITS HAVE CLEARED,
                                        ;WAIT FOR SILO TO EMPTY
                                        ;WHEN SILO IS EMPTY, SCOPE ON TEST

007654  005777  001740          12$:    TST     @DHBAR                  ;WAIT FOR ALL BAR BITS TO CLEAR
007660  001375                          BNE     .-4
007662  105777  001740                  TSTB    @DHSLR                  ;WAIT FOR SILO TO EMPTY
007666  001375                          BNE     .-4
007670  104400                  13$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP
```

```
                                                ;MULTI-LINE DATA TEST
                                                ;TRANSMIT A BINARY COUNT COUNT PATTERN ON ALLL LINES
                                                ;LINE SPEED IS 9600 BAUD
                                                ;SILO ALARM LEVEL IS SET TO 63 (NO RECEIVER DONE
                                                ;INTERRUPTS WILL OCCUR)
                                                ;CONTINUOUSLY ATTEMPT TO READ DATA FROM SILO
                                                ;IF "VALID DATA" FLAG IS DETECTED.
                                                ;THE DATA WILL BE CHECKED
007672  012767  000340  170076  T32:   MOV     #340,PS                 ;DISABLE ALL INTERRUPTS
007700  012767  000012  001746         MOV     #12,ICOUNT              ;SET UP FOR 12 ITERATIONS
007706  012767  010240  001734         MOV     #8$,ESCAPE              ;SET UP TO ESCAPE TO NEXT TEST
007714  012777  004000  001664         MOV #BIT11,@DHSCR               ;MASTER CLEAR INTERFACE
007722  012700  000020                 MOV     #20,R0                  ;SET UP TO LOAD BYTE
007726  005001                         CLR     R1                      ;COUNT AND BUS ADDRESS MEMORIES
                                                                       ;AND RECEIVED DATA STORAGE
007730  012702  000200                 MOV     #200,R2                 ;SET UP TO LOAD HIGH BYTE
007734  012703  000001                 MOV     #1,R3                                   ;OF EXPECTED DATA
007740  012777  011712  001646  1$:    MOV     #TBUF,@DHBA             ;LOAD BUSS ADDRESS
007746  012777  177400  001642         MOV     #-400,@DHBC             ;LOAD BYTE COUNT
007754  012777  033503  001630         MOV     #33503,@DHLPR           ;SET LINE SPEED AND
                                                                       ;CHARACTER LENGTH
007762  105061  012312                 CLRB    RBUF(R1)                ;CLEAR NEXT RECEIVED CHARACTER
007766  110263  012312                 MOVB    R2,RBUF(R3)             ;LOAD HIGH BYTE
007772  005277  001610                 INC     @DHSCR                  ;ADVANCE TO NEXT LINE
007776  005202                         INC     R2                      ;UPDATE POINTER TO
010000  062701  000002                 ADD     #2,R1                   ;LOW AND HIGH BYTE OF
010004  062703  000002                 ADD     #2,R3                   ;NEXT EXPECTED CHARACTER
010010  005300                         DEC     R0                      ;CONTINUE IF ALL LINES
010012  001352                         BNE     1$                      ;NOT SET UP
010014  012777  010104  001612         MOV     #2$,@DHTVEC             ;SET UP POINTER FOR
010022  012777  000240  001606         MOV     #240,@DHTLVL            ;TRANSMITTER INTERRUPT
010030  012777  010116  001572         MOV     #6$,@DHRVEC             ;SET UP POINTER FOR
010036  012777  000240  001566         MOV     #240,@DHRLVL            ;RECEIVER INTERRUPT
010044  012700  000012                 MOV     #12,R0                  ;SET UP TO TRANSMIT
010050  005067  001632                 CLR     ENDFLG                  ;10 (DEDIMAL) BLOCKS OF DATA
                                                                       ;ON EACH LINE
010054  012777  000077  001542         MOV     #77,@DHSSR
010062  012777  030000  001516         MOV     #30000,@DHSCR           ;ENABLE INTERFACE
010070  012777  177777  001522         MOV     #-1,@DHBAR              ;SET ALL BAR BITS
010076  005067  167674                 CLR     PS                      ;ENABLE INTERRUPTS
010102  000421                         BR      7$                      ;TRANSFER TO RECEIVER SERVICE

                                                ;TRANSMITTER INTERRUPT SERVICE ROUTINE
                                                ;CHECK FOR TRANSMIT DONE AND NO ERRORS

010104  005777  001476          2$:    TST     @DHSCR                  ;IS TRANSMITTER DONE SET
010110  100401                         BMI     .+4
010112  104004                         HLT     4                       ;TRANSMIT DONE NOT SET, ERROR
010114  000002                         RTI                             ;RETURN

                                                ;RECEIVER INTERRUPT SERVICE ROUTINE

010116  105777  001464          6$:    TSTB    @DHSCR                  ;IS CHARACTER AVAILABLE SET
010122  100001                         BPL     .+4
010124  104005                         HLT     5                       ;CHARACTER AVAILABLE SET, ERROR
```

```
010126  032777  040000  001452          BIT     #40000,@DHSCR           ;IS SILO OVERRUN SET
010134  001402                          BEQ     .+6
010136  104002                          HLT     2                       ;SILO OVERRUN, ERROR
010140  000401                          BR      3$
010142  104000                          HLT     0                       ;SPURIOUS INTERRUPT
                                                                        ;TEST WILL RESTART
010144  000002                  3$:     RTI

                                        ;RECEIVER DATA CHECK
                                        ;GET A CHARACTER FROM THE SILO
                                        ;IF IT IS VALID DATA, CHECK IT
                                        ;OTHERWISE, TRY AGAIN

010146  017701  001436          7$:     MOV     @DHNRC,R1               ;READ NEXT RECEIVED CHARACTER REGISTER
010152  005701                          TST     R1                      ;IF VALID DATA BIT NOT SET
010154  100374                          BPL     7$                      ;TRY AGAIN
010156  010102                          MOV     R1,R2                   ;EXTRACT LINE NUMBER
010160  000302                          SWAB    R2
010162  042702  177760                  BIC     #177760,R2
010166  010203                          MOV     R2,R3
010170  006302                          ASL     R2                      ;DOUBLE LINE NUMBER FOR OFFSET
010172  126201  012312                  CMPB    RBUF(R2),R1             ;COMPARE RECEIVED AND EXPECTED RESULTS
010176  001403                          BEQ     .+10
010200  016205  012312                  MOV     RBUF(R2),R5             ;GET EXPECTED DATA
010204  104003                          HLT     3                       ;DATA ERROR
010206  105262  012312                  INCB    RBUF(R2)                ;UPDATE EXPECTED DATA
010212  001355                          BNE     7$                      ;CONTINUE IF NOT 0
010214  005767  001466                  TST     ENDFLG                  ;IF ALL LINES HAVE FINISHED
010220  001007                          BNE     8$                      ;10 BLOCKS OF DATA, CLEAN UP
010222  005703                          TST     R3                      ;IF THE LINE IS 0
010224  001014                          BNE     9$
010226  005300                          DEC     R0                      ;UPDATE BLOCK COUNT
010230  001012                          BNE     9$                      ;IF LINE 0 HAS TRANSMITTED 10 BLOCKS
010232  012767  000001  001446          MOV     #1,ENDFLG               ;SET END FLAG
010240  005777  001354          8$:     TST     @DHBAR                  ;IF ALL LINES NOT DONE
010244  001340                          BNE     7$                      ;GET MORE CHARACTERS
010246  105777  001354                  TSTB    @DHSLR                  ;IF SILO IS NOT EMPTY
010252  001335                          BNE     7$                      ;GET REST OF DATA
010254  000417                          BR      10$                     ;WHEN EMPTY, SCOPE ON TEST
010256  042777  000017  001322  9$:     BIC     #17,@DHSCR              ;ADDRESS LINE THAT JUST FINISHED
010264  050377  001316                  BIS     R3,@DHSCR
010270  012777  011712  001316          MOV     #TBUF,@DHBA             ;SET UP BUS ADDRESS AND BYTE COUNT
010276  012777  177400  001312          MOV     #-400,@DHBC             ;FOR THAT LINE
010304  056277  010322  001306          BIS     BARBIT(R2),@DHBAR       ;SET BAR BIT FOR THAT LINE
010312  000715                          BR      7$                      ;CONTINUE
010314  104400                  10$:    SCOPE                           ;CHECK FOR ITERATIONS, LOOP
010316  000167  000040                  JMP     EOP                     ;GO TO END OF PASS ROUTINE
010322  000001                  BARBIT: 1
010324  000002                          2
010326  000004                          4
010330  000010                          10
010332  000020                          20
010334  000040                          40
010336  000100                          100
010340  000200                          200
010342  000400                          400
```

```
  1830  010344  001000                          1000
  1831  010346  002000                          2000
  1832  010350  004000                          4000
  1833  010352  010000                          10000
  1834  010354  020000                          20000
  1835  010356  040000                          40000
  1836  010360  100000                          100000
```

```
                                        ;END OF PASS
                                        ;TYPE NAME OF TEST
                                        ;UPDATE PASS COUNT
                                        ;CHECK FOR EXIT TO ACT-
                                        ;RESTART TEST

010362  104401                  EOP:    TYPE                            ;TYPE NAME OF TEST
010364  012765                          MEPASS
010366  005067  001312                  CLR     LAST                    ;CLEAR LAST ERROR PC
010372  005067  001242                  CLR     ERRFLG                  ;CLEAR ERROR FLAG
010376  005267  001240                  INC     PASCNT                  ;UPDATE PASS COUNT
010402  016767  001234  167160          MOV     PASCNT,LIGHTS           ;DISPLAY PASS COUNT
010410  013701  000042                  MOV     @#42,R1                 ;CHECK FOR ACT-11 OR DDP
010414  001405                          BEQ     RESTRT                  ;IF NOT, CONTINUE TESTING
010416  000005                          RESET
010420  004711                  LOGICAL:        JSR     PC,(R1)
010422  000240                          NOP
010424  000240                          NOP
010426  000240                          NOP
010430  000167  170546          RESTRT: JMP     BEGIN

                                        ;CHECK FOR LOOP ON CURRENT TEST
                                        ;CHECK FOR ITERATION SUPPRESSION

010434  032767  002000  167126  SCOPER: BIT     #SW10,SWR
010442  001030                          BNE     4$
010444  032767  040000  167116  1$:     BIT     #SW14,SWR
010452  001021                          BNE     3$
010454  032767  004000  167106          BIT     #SW11,SWR
010462  001006                          BNE     2$
010464  005267  001166                  INC     LPCNT
010470  026767  001162  001156          CMP     LPCNT,ICOUNT
010476  001007                          BNE     3$
010500  005067  001152          2$:     CLR     LPCNT
010504  005067  001130                  CLR     ERRFLG
010510  011667  001132                  MOV     (SP),RETURN
010514  000002                          RTI
010516  016716  001124          3$:     MOV     RETURN,(SP)
010522  000002                          RTI
010524  005767  001110          4$:     TST     ERRFLG
010530  001745                          BEQ     1$
010532  000762                          BR      2$

                                        ;CHECK FOR FREEZE ON CURRENT DATA

010534  032767  001000  167026  SCOP1R: BIT     #SW09,SWR
010542  001402                          BEQ     1$
010544  016716  001102                  MOV     FREEZ1,(SP)
010550  000002                  1$:     RTI

                                        ;ERROR HANDLER

010552  032767  020000  167010  ERRORS: BIT     #SW13,SWR
010560  001051                          BNE     HALTS
```

```
010562  021667  001116                  CMP     (SP),LAST
010566  001404                          BEQ     1$
010570  011667  001110                  MOV     (SP),LAST
010574  005067  001040                  CLR     ERRFLG
010600  104406                  1$:     SAVOSP
010602  011605                          MOV     (SP),R5
010604  162705  000002                  SUB     #2,R5
010610  011504                          MOV     (R5),R4
010612  006304                          ASL     R4
010614  006304                          ASL     R4
010616  042704  177001                  BIC     #177001,R4
010622  062704  013074                  ADD     #ERRTAB,R4
010626  012467  000034                  MOV     (R4)+,ERRMSG
010632  011467  000042                  MOV     (R4),DATABP
010636  005767  000776                  TST     ERRFLG
010642  001403                          BEQ     TYPMSG
010644  005767  000030                  TST     DATABP
010650  001007                          BNE     TYPDAT
010652  104402                  TYPMSG: OCTASC
010654  010746                          ERTABO
010656  012767  000001  000754          MOV     #1,ERRFLG
010664  104401                          TYPE
010666  000000                  ERRMSG: 0
010670  005767  000004          TYPDAT: TST     DATABP
010674  001402                          BEQ     RESREG
010676  104402                          OCTASC
010700  000000                  DATABP: 0
010702  104407                  RESREG: RESOS
010704  005767  166660          HALTS:  TST     SWR
010710  100005                          BPL     EXITER
010712  010046                          PUSHRO
010714  016600  000002                  MOV     2(SP),R0
010720  000000                          HALT
010722  012600                          POPRO
010724  005267  000714          EXITER: INC     ERRCNT
010730  032767  002000  166632          BIT     #SW10,SWR
010736  001402                          BEQ     1$
010740  016716  000704                  MOV     ESCAPE,(SP)
010744  000002                  1$:     RTI
010746  000001                  ERTABO: 1
010750     006     002                  .BYTE   6,2
010752  011676                          SAVPC
                                        ;TRAP DISPATCH SERVICE
                                        ;ARGUMENT OF TRAP IS EXTRACTED
                                        ;AND USED AS OFFSET TO OBTAIN POINTER
                                        ;TO SELECTED SUBROUTINE

010754  011646                  TRPSRV: MOV     (SP),-(SP)              ;GET PC OF RETURN
010756  162716  000002                  SUB     #2,(SP)                 ;=PC OF TRAP
010762  017616  000000                  MOV     @(SP),(SP)              ;GET TRP
010766  006316                  TRPOK:  ASL     (SP)                    ;MULTIPLY TRAP ARG BY 2
010770  042716  177001                  BIC     #177001,(SP)            ;CLEAR UNWANTED BITS
010774  062716  013014                  ADD     #TRPTAB,(SP)            ;POINTER TO SUBROUTINE ADDRESS
011000  017616  000000                  MOV     @(SP),(SP)              ;SUBROUTINE ADDRESS
011004  000136                          JMP     @(SP)+                  ;GO TO SUBROUTINE
```

```
                                                        ;TELETYPE OUTPUT ROUTINE

011006  017605  000000          TYPER:  MOV     @(SP),R5
011012  062716  000002                  ADD     #2,(SP)
011016  105777  000560          1$:     TSTB    @TPCSR
011022  100375                          BPL     1$
011024  105715                          TSTB    (R5)
011026  001001                          BNE     2$
011030  000002                          RTI
011032  112577  000546          2$:     MOVB    (R5)+,@TPDBR
011036  000767                          BR      1$

                                                        ;ASCII STRING INPUT ROUTINE

011040  017667  000000  000006  INSTRG: MOV     @(SP),MSG
011046  062716  000002                  ADD     #2,(SP)
011052  104401                  INSTR1: TYPE
011054  000000                  MSG:    0
011056  012704  013036                  MOV     #INBUF,R4
011062  012703  000007                  MOV     #7,R3
011066  105777  000504          1$:     TSTB    @TKCSR
011072  100375                          BPL     1$
011074  117714  000500                  MOVB    @TKDBR,(R4)
011100  142714  000200                  BICB    #200,(R4)
011104  122427  000015                  CMPB    (R4)+,#15
011110  001413                          BEQ     INSTR2
011112  117777  000462  000464          MOVB    @TKDBR,@TPDBR
011120  105777  000456          2$:     TSTB    @TPCSR
011124  100375                          BPL     2$
011126  005303                          DEC     R3
011130  001356                          BNE     1$
011132  104401                  INSTRE: TYPE
011134  012671                          MOM
011136  000745                          BR      INSTR1
011140  000002                  INSTR2: RTI

                                                        ;CONVERT ASCII STRING TO OCTAL

011142  011605                  PARAMS: MOV     (SP),R5
011144  012567  000146                  MOV     (R5)+,LOLIM
011150  012567  000144                  MOV     (R5)+,HILIM
011154  012567  000142                  MOV     (R5)+,DEVADR
011160  112567  000140                  MOVB    (R5)+,LOBITS
011164  112567  000135                  MOVB    (R5)+,ADRCNT
011170  010516                          MOV     R5,(SP)
011172  005005                  PARAM1: CLR     R5
011174  012704  013036                  MOV     #INBUF,R4
011200  122714  000015                  CMPB    #15,(R4)
011204  001420                          BEQ     PARERR
011206  121427  000060          1$:     CMPB    (R4),#60
011212  002415                          BLT     PARERR
011214  121427  000067                  CMPB    (R4),#67
011220  003012                          BGT     PARERR
011222  142714  000060                  BICB    #60,(R4)
011226  152405                          BISB    (R4)+,R5
011230  122714  000015                  CMPB    #15,(R4)
```

```
011234  001406                          BEQ     LIMITS
011236  006305                          ASL     R5
011240  006305                          ASL     R5
011242  006305                          ASL     R5
011244  000760                          BR      1$
011246  104404                  PARERR: INSTER
011250  000750                          BR      PARAM1

                                        ;TEST TO SEE IF NUMBER IS WITHIN LIMITS

011252  020567  000042          LIMITS: CMP     R5,HILIM
011256  101373                          BHI     PARERR
011260  020567  000032                  CMP     R5,LOLIM
011264  103770                          BLO     PARERR
011266  136705  000032                  BITB    LOBITS,R5
011272  001365                          BNE     PARERR

                                        ;STORE NUMBER AT SPECIFIED ADDRESS

011274  016704  000022                  MOV     DEVADR,R4
011300  010524                  1$:     MOV     R5,(R4)+
011302  062705  000002                  ADD     #2,R5
011306  105367  000013                  DECB    ADRCNT
011312  001372                          BNE     1$
011314  000002                          RTI
011316  000000                  LOLIM:  0
011320  000000                  HILIM:  0
011322  000000                  DEVADR: 0
011324  000000                  LOBITS: 0
        011325                  ADRCNT=LOBITS+1

                                        ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER

011326  104401                  OCTASN: TYPE
011330  012675                          MCRLF
011332  017601  000000                  MOV     @(SP),R1
011336  062716  000002                  ADD     #2,(SP)
011342  012167  000130                  MOV     (R1)+,WRDCNT
011346  112167  000126          1$:     MOVB    (R1)+,CHRCNT
011352  112167  000123                  MOVB    (R1)+,SPACNT
011356  013167  000120                  MOV     @(R1)+,BINWRD
011362  016704  000114          2$:     MOV     BINWRD,R4
011366  116705  000106                  MOVB    CHRCNT,R5
011372  012700  013050                  MOV     #TEMP,R0
011376  010403                  3$:     MOV     R4,R3
011400  042703  177770                  BIC     #177770,R3
011404  062703  000260                  ADD     #260,R3
011410  110320                          MOVB    R3,(R0)+
011412  006204                          ASR     R4
011414  006204                          ASR     R4
011416  006204                          ASR     R4
011420  005305                          DEC     R5
011422  001365                          BNE     3$
011424  012703  013062                  MOV     #MDATA,R3
011430  114023                  4$:     MOVB    -(R0),(R3)+
011432  105367  000042                  DECB    CHRCNT
```

```
011436  001374                          BNE     4$
011440  105767  000035                  TSTB    SPACNT
011444  001405                          BEQ     6$
011446  112723  000240          5$:     MOVB    #240,(R3)+
011452  105367  000023                  DECB    SPACNT
011456  001373                          BNE     5$
011460  105013                  6$:     CLRB    (R3)
011462  104401                          TYPE
011464  013062                          MDATA
011466  005367  000004                  DEC     WRDCNT
011472  001325                          BNE     1$
011474  000002                          RTI
011476  000000                  WRDCNT: 0
011500  000000                  CHRCNT: 0
        011501                  SPACNT=CHRCNT+1
011502  000000                  BINWRD: 0

                                        ;SAVE PC OF TEST THAT FAILED AND R0-R5

011504  016667  000004  000164  SVOSP:  MOV     4(SP),SAVPC

                                        ;SAVE R0-R5

011512  010567  000154          SVOS:   MOV     R5,SAVR5
011516  010467  000146                  MOV     R4,SAVR4
011522  010367  000140                  MOV     R3,SAVR3
011526  010267  000132                  MOV     R2,SAVR2
011532  010167  000124                  MOV     R1,SAVR1
011536  010067  000116                  MOV     R0,SAVR0
011542  000002                          RTI
                                        ;RESTORE R0-R5

011544  016700  000110          PSOS:   MOV     SAVR0,R0
011550  016701  000106                  MOV     SAVR1,R1
011554  016702  000104                  MOV     SAVR2,R2
011560  016703  000102                  MOV     SAVR3,R3
011564  016704  000100                  MOV     SAVR4,R4
011570  016705  000076                  MOV     SAVR5,R5
011574  000002                          RTI
                                        ;INDIRECT POINTERS

011576  177560                  TKCSR:  177560
011600  177562                  TKDBR:  177562
011602  177564                  TPCSR:  177564
011604  177566                  TPDBR:  177566
011606  000000                  DHSCR:  0
011610  000000                  DHNRC:  0
011612  000000                  DHLPR:  0
011614  000000                  DHBA:   0
011616  000000                  DHBC:   0
011620  000000                  DHBAR:  0
011622  000000                  DHBCR:  0
011624  000000                  DHSSR:  0
011626  000000                  DHSLR:  0
011630  000000                  DHRVEC: 0
011632  000000                  DHRLVL: 0
```

```
011634  000000                    DHTVEC: 0
011636  000000                    DHTLVL: 0
                                          ;PROGRAM VARIABLES

011640  000000                    ERRFLG: 0                ;ERROR FLAG
011642  000000                    PASCNT: 0                ;PASS COUNT
011644  000000                    ERRCNT: 0                ;ERROR COUNT
011646  000000                    RETURN: 0                ;SCOPE RETURN ADDRESS FOR TEST LOOPING
011650  000000                    ESCAPE: 0                ;ADDRESS FOR ERROR ESCAPE
011652  000000                    FREEZ1: 0                ;DATA LOOPING RETURN ADDRESS
011654  000000                    ICOUNT: 0                ;ITERATION COUNT FOR TEST IN PROGRESS
011656  000000                    LPCNT:  0                ;NUMBER OF ITERATIONS THIS TEST
011660  000000                    SAVR0:  0                ;R0 SAVE AREA
011662  000000                    SAVR1:  0                ;R1 SAVE AREA
011664  000000                    SAVR2:  0                ;R2 SAVE AREA
011666  000000                    SAVR3:  0                ;R3 SAVE ARE
011670  000000                    SAVR4:  0                ;R4 SAVE AREA
011672  000000                    SAVR5:  0                ;R5 SAVE AREA
011674  000000                    SAVSP:  0                ;STACK POINTER SAVE AREA
011676  000000                    SAVPC:  0                ;CALLING ROUTINE SAVE AREA
011700  000000                    INIFLG: 0                ;PROGRAM INITIALIZATION FLAG
011702  000000                    STFLG:  0                ;PROGRAM START FLAG
011704  000000                    LAST:   0                ;LAST ERROR PC
011706  000000                    ENDFLG: 0
011710  000000                    TDATA:  0
011712     000                    TBUF:   .BYTE   0
011713     001                    .BYTE   TDAT
011714     002                    .BYTE   TDAT
011715     003                    .BYTE   TDAT
011716     004                    .BYTE   TDAT
011717     005                    .BYTE   TDAT
011720     006                    .BYTE   TDAT
011721     007                    .BYTE   TDAT
011722     010                    .BYTE   TDAT
011723     011                    .BYTE   TDAT
011724     012                    .BYTE   TDAT
011725     013                    .BYTE   TDAT
011726     014                    .BYTE   TDAT
011727     015                    .BYTE   TDAT
011730     016                    .BYTE   TDAT
011731     017                    .BYTE   TDAT
011732     020                    .BYTE   TDAT
011733     021                    .BYTE   TDAT
011734     022                    .BYTE   TDAT
011735     023                    .BYTE   TDAT
011736     024                    .BYTE   TDAT
011737     025                    .BYTE   TDAT
011740     026                    .BYTE   TDAT
011741     027                    .BYTE   TDAT
011742     030                    .BYTE   TDAT
011743     031                    .BYTE   TDAT
011744     032                    .BYTE   TDAT
011745     033                    .BYTE   TDAT
011746     034                    .BYTE   TDAT
011747     035                    .BYTE   TDAT
011750     036                    .BYTE   TDAT
```

```
2173  011751    037              .BYTE   TDAT
2174  011752    040              .BYTE   TDAT
2175  011753    041              .BYTE   TDAT
2176  011754    042              .BYTE   TDAT
2177  011755    043              .BYTE   TDAT
2178  011756    044              .BYTE   TDAT
2179  011757    045              .BYTE   TDAT
2180  011760    046              .BYTE   TDAT
2181  011761    047              .BYTE   TDAT
2182  011762    050              .BYTE   TDAT
2183  011763    051              .BYTE   TDAT
2184  011764    052              .BYTE   TDAT
2185  011765    053              .BYTE   TDAT
2186  011766    054              .BYTE   TDAT
2187  011767    055              .BYTE   TDAT
2188  011770    056              .BYTE   TDAT
2189  011771    057              .BYTE   TDAT
2190  011772    060              .BYTE   TDAT
2191  011773    061              .BYTE   TDAT
2192  011774    062              .BYTE   TDAT
2193  011775    063              .BYTE   TDAT
2194  011776    064              .BYTE   TDAT
2195  011777    065              .BYTE   TDAT
2196  012000    066              .BYTE   TDAT
2197  012001    067              .BYTE   TDAT
2198  012002    070              .BYTE   TDAT
2199  012003    071              .BYTE   TDAT
2200  012004    072              .BYTE   TDAT
2201  012005    073              .BYTE   TDAT
2202  012006    074              .BYTE   TDAT
2203  012007    075              .BYTE   TDAT
2204  012010    076              .BYTE   TDAT
2205  012011    077              .BYTE   TDAT
2206  012012    100              .BYTE   TDAT
2207  012013    101              .BYTE   TDAT
2208  012014    102              .BYTE   TDAT
2209  012015    103              .BYTE   TDAT
2210  012016    104              .BYTE   TDAT
2211  012017    105              .BYTE   TDAT
2212  012020    106              .BYTE   TDAT
2213  012021    107              .BYTE   TDAT
2214  012022    110              .BYTE   TDAT
2215  012023    111              .BYTE   TDAT
2216  012024    112              .BYTE   TDAT
2217  012025    113              .BYTE   TDAT
2218  012026    114              .BYTE   TDAT
2219  012027    115              .BYTE   TDAT
2220  012030    116              .BYTE   TDAT
2221  012031    117              .BYTE   TDAT
2222  012032    120              .BYTE   TDAT
2223  012033    121              .BYTE   TDAT
2224  012034    122              .BYTE   TDAT
2225  012035    123              .BYTE   TDAT
2226  012036    124              .BYTE   TDAT
2227  012037    125              .BYTE   TDAT
2228  012040    126              .BYTE   TDAT
```

```
  2229  012041     127                    .BYTE   TDAT
  2230  012042     130                    .BYTE   TDAT
  2231  012043     131                    .BYTE   TDAT
  2232  012044     132                    .BYTE   TDAT
  2233  012045     133                    .BYTE   TDAT
  2234  012046     134                    .BYTE   TDAT
  2235  012047     135                    .BYTE   TDAT
  2236  012050     136                    .BYTE   TDAT
  2237  012051     137                    .BYTE   TDAT
  2238  012052     140                    .BYTE   TDAT
  2239  012053     141                    .BYTE   TDAT
  2240  012054     142                    .BYTE   TDAT
  2241  012055     143                    .BYTE   TDAT
  2242  012056     144                    .BYTE   TDAT
  2243  012057     145                    .BYTE   TDAT
  2244  012060     146                    .BYTE   TDAT
  2245  012061     147                    .BYTE   TDAT
  2246  012062     150                    .BYTE   TDAT
  2247  012063     151                    .BYTE   TDAT
  2248  012064     152                    .BYTE   TDAT
  2249  012065     153                    .BYTE   TDAT
  2250  012066     154                    .BYTE   TDAT
  2251  012067     155                    .BYTE   TDAT
  2252  012070     156                    .BYTE   TDAT
  2253  012071     157                    .BYTE   TDAT
  2254  012072     160                    .BYTE   TDAT
  2255  012073     161                    .BYTE   TDAT
  2256  012074     162                    .BYTE   TDAT
  2257  012075     163                    .BYTE   TDAT
  2258  012076     164                    .BYTE   TDAT
  2259  012077     165                    .BYTE   TDAT
  2260  012100     166                    .BYTE   TDAT
  2261  012101     167                    .BYTE   TDAT
  2262  012102     170                    .BYTE   TDAT
  2263  012103     171                    .BYTE   TDAT
  2264  012104     172                    .BYTE   TDAT
  2265  012105     173                    .BYTE   TDAT
  2266  012106     174                    .BYTE   TDAT
  2267  012107     175                    .BYTE   TDAT
  2268  012110     176                    .BYTE   TDAT
  2269  012111     177                    .BYTE   TDAT
  2270  012112     200                    .BYTE   TDAT
  2271  012113     201                    .BYTE   TDAT
  2272  012114     202                    .BYTE   TDAT
  2273  012115     203                    .BYTE   TDAT
  2274  012116     204                    .BYTE   TDAT
  2275  012117     205                    .BYTE   TDAT
  2276  012120     206                    .BYTE   TDAT
  2277  012121     207                    .BYTE   TDAT
  2278  012122     210                    .BYTE   TDAT
  2279  012123     211                    .BYTE   TDAT
  2280  012124     212                    .BYTE   TDAT
  2281  012125     213                    .BYTE   TDAT
  2282  012126     214                    .BYTE   TDAT
  2283  012127     215                    .BYTE   TDAT
  2284  012130     216                    .BYTE   TDAT
```

```
2285  012131    217          .BYTE   TDAT
2286  012132    220          .BYTE   TDAT
2287  012133    221          .BYTE   TDAT
2288  012134    222          .BYTE   TDAT
2289  012135    223          .BYTE   TDAT
2290  012136    224          .BYTE   TDAT
2291  012137    225          .BYTE   TDAT
2292  012140    226          .BYTE   TDAT
2293  012141    227          .BYTE   TDAT
2294  012142    230          .BYTE   TDAT
2295  012143    231          .BYTE   TDAT
2296  012144    232          .BYTE   TDAT
2297  012145    233          .BYTE   TDAT
2298  012146    234          .BYTE   TDAT
2299  012147    235          .BYTE   TDAT
2300  012150    236          .BYTE   TDAT
2301  012151    237          .BYTE   TDAT
2302  012152    240          .BYTE   TDAT
2303  012153    241          .BYTE   TDAT
2304  012154    242          .BYTE   TDAT
2305  012155    243          .BYTE   TDAT
2306  012156    244          .BYTE   TDAT
2307  012157    245          .BYTE   TDAT
2308  012160    246          .BYTE   TDAT
2309  012161    247          .BYTE   TDAT
2310  012162    250          .BYTE   TDAT
2311  012163    251          .BYTE   TDAT
2312  012164    252          .BYTE   TDAT
2313  012165    253          .BYTE   TDAT
2314  012166    254          .BYTE   TDAT
2315  012167    255          .BYTE   TDAT
2316  012170    256          .BYTE   TDAT
2317  012171    257          .BYTE   TDAT
2318  012172    260          .BYTE   TDAT
2319  012173    261          .BYTE   TDAT
2320  012174    262          .BYTE   TDAT
2321  012175    263          .BYTE   TDAT
2322  012176    264          .BYTE   TDAT
2323  012177    265          .BYTE   TDAT
2324  012200    266          .BYTE   TDAT
2325  012201    267          .BYTE   TDAT
2326  012202    270          .BYTE   TDAT
2327  012203    271          .BYTE   TDAT
2328  012204    272          .BYTE   TDAT
2329  012205    273          .BYTE   TDAT
2330  012206    274          .BYTE   TDAT
2331  012207    275          .BYTE   TDAT
2332  012210    276          .BYTE   TDAT
2333  012211    277          .BYTE   TDAT
2334  012212    300          .BYTE   TDAT
2335  012213    301          .BYTE   TDAT
2336  012214    302          .BYTE   TDAT
2337  012215    303          .BYTE   TDAT
2338  012216    304          .BYTE   TDAT
2339  012217    305          .BYTE   TDAT
2340  012220    306          .BYTE   TDAT
```

```
  2341  012221     307                    .BYTE   TDAT
  2342  012222     310                    .BYTE   TDAT
  2343  012223     311                    .BYTE   TDAT
  2344  012224     312                    .BYTE   TDAT
  2345  012225     313                    .BYTE   TDAT
  2346  012226     314                    .BYTE   TDAT
  2347  012227     315                    .BYTE   TDAT
  2348  012230     316                    .BYTE   TDAT
  2349  012231     317                    .BYTE   TDAT
  2350  012232     320                    .BYTE   TDAT
  2351  012233     321                    .BYTE   TDAT
  2352  012234     322                    .BYTE   TDAT
  2353  012235     323                    .BYTE   TDAT
  2354  012236     324                    .BYTE   TDAT
  2355  012237     325                    .BYTE   TDAT
  2356  012240     326                    .BYTE   TDAT
  2357  012241     327                    .BYTE   TDAT
  2358  012242     330                    .BYTE   TDAT
  2359  012243     331                    .BYTE   TDAT
  2360  012244     332                    .BYTE   TDAT
  2361  012245     333                    .BYTE   TDAT
  2362  012246     334                    .BYTE   TDAT
  2363  012247     335                    .BYTE   TDAT
  2364  012250     336                    .BYTE   TDAT
  2365  012251     337                    .BYTE   TDAT
  2366  012252     340                    .BYTE   TDAT
  2367  012253     341                    .BYTE   TDAT
  2368  012254     342                    .BYTE   TDAT
  2369  012255     343                    .BYTE   TDAT
  2370  012256     344                    .BYTE   TDAT
  2371  012257     345                    .BYTE   TDAT
  2372  012260     346                    .BYTE   TDAT
  2373  012261     347                    .BYTE   TDAT
  2374  012262     350                    .BYTE   TDAT
  2375  012263     351                    .BYTE   TDAT
  2376  012264     352                    .BYTE   TDAT
  2377  012265     353                    .BYTE   TDAT
  2378  012266     354                    .BYTE   TDAT
  2379  012267     355                    .BYTE   TDAT
  2380  012270     356                    .BYTE   TDAT
  2381  012271     357                    .BYTE   TDAT
  2382  012272     360                    .BYTE   TDAT
  2383  012273     361                    .BYTE   TDAT
  2384  012274     362                    .BYTE   TDAT
  2385  012275     363                    .BYTE   TDAT
  2386  012276     364                    .BYTE   TDAT
  2387  012277     365                    .BYTE   TDAT
  2388  012300     366                    .BYTE   TDAT
  2389  012301     367                    .BYTE   TDAT
  2390  012302     370                    .BYTE   TDAT
  2391  012303     371                    .BYTE   TDAT
  2392  012304     372                    .BYTE   TDAT
  2393  012305     373                    .BYTE   TDAT
  2394  012306     374                    .BYTE   TDAT
  2395  012307     375                    .BYTE   TDAT
  2396  012310     376                    .BYTE   TDAT
```

```
012311     377                    .BYTE   TDAT
                                  .EVEN
012312  000000                    RBUF:   0
        012354                    .=.+40
                                          ;ENTER HERE ON POWER FAILURE


012354  010046                    PFAIL:  MOV     R0,-(SP)                ;SAVE R0-R5 ON PROCESSOR STACK
012356  010146                            MOV     R1,-(SP)
012360  010246                            MOV     R2,-(SP)
012362  010346                            MOV     R3,-(SP)
012364  010446                            MOV     R4,-(SP)
012366  010546                            MOV     R5,-(SP)
012370  016746  165430                    MOV     24,-(SP)
012374  010667  177274                    MOV     SP,SAVSP                ;SAVE STACK POINTER
012400  012767  012412  165416            MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
012406  000000                            HALT                            ;HALT ON POWER DOWN NORMAL
012410  000777                            BR      .

                                          ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED

012412  016706  177256            RESTAR: MOV     SAVSP,SP                ;RESTORE STACK POINTER
012416  012605                            MOV     (SP)+,R5                ;RESTORE R0-R5
012420  012604                            MOV     (SP)+,R4
012422  012603                            MOV     (SP)+,R3
012424  012602                            MOV     (SP)+,R2
012426  012601                            MOV     (SP)+,R1
012430  012600                            MOV     (SP)+,R0
012432  012767  012354  165364            MOV     #PFAIL,24               ;SET UP FOR POWER FAILURE
012440  012767  000340  165330            MOV     #340,PS
012446  012706  013564                    MOV     #STACK,SP
012452  005067  000372                    CLR     TEMP
012456  005267  000366                    INC     TEMP
012462  001375                            BNE     .-4
012464  104402                            OCTASC
012466  012510                            PFTAB
012470  104401                            TYPE
012472  012700                            MPFAIL
012474  005067  177140                    CLR     ERRFLG
012500  005067  177200                    CLR     LAST
012504  000177  177136                    JMP     @RETURN
012510  000001                    PFTAB:  1
012512  000006  000002                    6,2
012516  000207                            RETURN
012520  005015  042012  030510    MTITLE: .ASCIZ  <15><12><12>/DH11 SINGLE LINE PARITY CHECK & MULTI-LINE DATA TEST /<15><
012526  020061  044523  043516
012534  042514  046040  047111
012542  020105  040520  044522
012550  054524  041440  042510
012556  045503  023040  046440
012564  046125  044524  046055
012572  047111  020105  040504
012600  040524  052040  051505
012606  020124  005015     000
012613     015  053012  041505    MVECTO: .ASCIZ  <15><12>/VECTOR ADDRESS-/
012620  047524  020122  042101
```

```
012626  051104  051505  026523
012634     000
012635     015  041412  047117  MREGAD: .ASCIZ  <15><12>/CONTROL REGISTER ADDRESS-/
012642  051124  046117  051040
012650  043505  051511  042524
012656  020122  042101  051104
012664  051505  026523     000
012671     040  037440     000  MQM:    .ASCIZ  /  ?/
012675     015  000012          MCRLF:  .ASCIZ   <15><12>
012700  020040  047520  042527  MPFAIL: .ASCIZ  /  POWER FAILURE, PROGRAM RESTART AT TEST IN PROGRESS/
012706  020122  040506  046111
012714  051125  026105  050040
012722  047522  051107  046501
012730  051040  051505  040524
012736  052122  040440  020124
012744  042524  052123  044440
012752  020116  051120  043517
012760  042522  051523     000
012765     015  042012  042132  MEPASS: .ASCIZ  <15><12>/DZDHG/
012772  043510     000
012775     015  051012     000  MR:     .ASCIZ  <15><12>/R/
013001     015  052012  051505  MTSTPC: .ASCIZ  <15><12>/TEST PC-/
013006  020124  041520  000055
                                .EVEN

                                        ;TABLE OF POINTERS FOR TRAP DECODING

013014  010434                  TRPTAB: SCOPER
013016  011006                          TYPER
013020  011326                          OCTASN
013022  011040                          INSTRG
013024  011132                          INSTRE
013026  011142                          PARAMS
013030  011504                          SVOSP
013032  011544                          RSOS
013034  010534                          SCOP1R

                                        ;BUFFERS FOR INPUT-OUTPUT

013036  000000                  INBUF:  0
        013050                  .=.+10
013050  000000                  TEMP:   0
        013062                  .=.+10
013062  000000                  MDATA:  0
        013074                  .=.+10

                                        ;TABLE OF POINTERS TO ERROR MESSAGES AND DATA

013074                          ERRTAB:
013074  013124                          EM0
013076  000000                          0
013100  013151                          EM1
013102  000000                          0
013104  013205                          EM2
013106  000000                          0
013110  013226                          EM3
```

```
013112  013346                          DT1
013114  013267                          EM4
013116  000000                          0
013120  013315                          EM5
013122  000000                          0
013124  047125  054105  042520  EM0:    .ASCIZ  /UNEXPECTED INTERRUPT/
013132  052103  042105  044440
013140  052116  051105  052522
013146  052120     000
013151     103  040510  040522  EM1:    .ASCIZ  /CHARACTER AVAILABLE NOT SET/
013156  052103  051105  040440
013164  040526  046111  041101
013172  042514  047040  052117
013200  051440  052105     000
013205     123  046111  020117  EM2:    .ASCIZ  /SILO OVERRUN SET/
013212  053117  051105  052522
013220  020116  042523  000124
013226  040504  040524  042440  EM3:    .ASCII  /DATA ERROR/
013234  051122  051117
013240  005015  054105  020120          .ASCIZ  <15><12>/EXP     REC     LINE/
013246  020040  020040  042522
013254  020103  020040  020040
013262  044514  042516     000
013267     124  040522  051516  EM4:    .ASCIZ  /TRANSMIT DONE NOT SET/
013274  044515  020124  047504
013302  042516  047040  052117
013310  051440  052105     000
013315     103  040510  040522  EM5:    .ASCIZ  /CHARACTER AVAILABLE SET/
013322  052103  051105  040440
013330  040526  046111  041101
013336  042514  051440  052105
013344     000
        013346                  .EVEN

                                        ;DATA TABLES FOR ERRORS

013346  000003                  DT1:    3
013350     006     002          .BYTE   6,2
013352  011672                          SAVR5
013354     006     002          .BYTE   6,2
013356  011662                          SAVR1
013360     002     000          .BYTE   2,0
013362  011666                          SAVR3
013364  000000                  ENDCOD: 0
        000001                  .END
```

```
ADRCNT= 011325          1992*   2027*   2034*
BARBIT  010322          1817    1821*
BEGIN   001202           395     424     430*   1858
BINWRD  011502          2045*   2046    2076*
BITX  = 000000           449*    484*    519*    554*    589*    624*    659*    694*    729*    764*    799*    834*    869*
                         904*    939*    974*   1009*
BIT00 = 000001            85*
BIT01 = 000002            84*
BIT02 = 000004            83*
BIT03 = 000010            82*
BIT04 = 000020            81*
BIT05 = 000040            80*
BIT06 = 000100            79*
BIT07 = 000200            78*
BIT08 = 000400            77*
BIT09 = 001000            76*
BIT10 = 002000            75*
BIT11 = 004000            74*     463     498     533     568     603     638     673     708     743     778     813     848
                          883     918     953     988    1023    1073    1102    1152    1181    1231    1260    1310    1339
                         1389    1418    1468    1497    1547    1576    1626    1654    1704    1730
BIT12 = 010000            73*
BIT13 = 020000            72*
BIT14 = 040000            71*
BIT15 = 100000            70*
CHRCNT  011500          2043*   2047    2060*   2074*   2075
DATABP  010700          1906*   1909    1916    1919*
DEVADR  011322          1990*   2024    2032*
DHBA    011614           474*    509*    544*    579*    614*    649*    684*    719*    754*    789*    824*    859*    894*
                         929*    964*    999*   1029*   1108*   1187*   1266*   1345*   1424*   1503*   1582*   1660*   1736*
                        1815*   2109*
DHBAR   011620           476*    511*    546*    581*    616*    651*    686*    721*    756*    791*    826*    861*    896*
                         931*    966*   1001*   1044*   1082    1123*   1161    1202*   1240    1281*   1319    1360*   1398
                        1439*   1477    1518*   1556    1597*   1635    1675*   1713    1757*   1808    1817*   2111*
DHBC    011616           473*    508*    543*    578*    613*    648*    683*    718*    753*    788*    823*    858*    893*
                         928*    963*    998*   1030*   1109*   1188*   1267*   1346*   1425*   1504*   1583*   1661*   1737*
                        1816*   2110*
DHBCR   011622           475*    510*    545*    580*    615*    650*    685*    720*    755*    790*    825*    860*    895*
                         930*    965*   1000*   2112*
DHLPR   011612           471*    506*    541*    576*    611*    646*    681*    716*    751*    786*    821*    856*    891*
                         926*    961*    996*   1031*   1110*   1189*   1268*   1347*   1426*   1505*   1584*   1662*   1738*
                        2108*
DHNRC   011610           479     514     549     584     619     654     689     724     759     794     829     864     899
                         934     969    1004    1059    1138    1217    1296    1375    1454    1533    1612    1690    1787
                        2107*
DHRLVL  011632          1042*   1121*   1200*   1279*   1358*   1437*   1516*   1595*   1673*   1751*   2116*
DHRVEC  011630           410    1041*   1120*   1199*   1278*   1357*   1436*   1515*   1594*   1672*   1750*   2115*
DHSCR   011606           418     463*    470*    477     498*    505*    512     533*    540*    547     568*    575*    582
                         603*    610*    617     638*    645*    652     673*    680*    687     708*    715*    722     743*
                         750*    757     778*    785*    792     813*    820*    827     848*    855*    862     883*    890*
                         897     918*    925*    932     953*    960*    967     988*    995*   1002    1023*   1035*   1043*
                        1054    1056    1073*   1102*   1114*   1122*   1133    1135    1152*   1181*   1193*   1201*   1212
                        1214    1231*   1260*   1272*   1280*   1291    1293    1310*   1339*   1351*   1359*   1370    1372
                        1389*   1418*   1430*   1438*   1449    1451    1468*   1497*   1509*   1517*   1528    1530    1547*
                        1576*   1588*   1596*   1607    1609    1626*   1654*   1666*   1674*   1685    1687    1704*   1730*
                        1742*   1756*   1764    1771    1774    1813*   1814*   2106*
DHSLR   011626           421*    422*    1084    1163    1242    1321    1400    1479    1558    1637    1715    1810    2114*
```

```
DHSSR   011624          421    1755*  2113#
DHTLVL  011636         1749*   2118#
DHTVEC  011634         1748*   2117#
DT1     013346         2509    2545#
EM0     013124         2502    2514#
EM1     013151         2504    2518#
EM2     013205         2506    2523#
EM3     013226         2508    2526#
EM4     013267         2510    2532#
EM5     013315         2512    2536#
ENDCOD  013364           57    2552#
ENDFLG  011706         1753*   1801   1807*  2140#
EOP     010362         1820    1845#
ERRCNT  011644          386*   1927*  2123#
ERRFLG  011640          387*    388*  1848*  1873*  1878   1896*  1907   1913*  2121#  2435*
ERRMSG  010666         1905*   1915#
ERRORS  010552          350    1891#
ERRTAB  013074         1904    2501#
ERTABO  010746         1912    1932#
ESCAPE  011650          461*    496*   531*   566*   601*   636*   671*   706*   741*   776*   811*   846*   881*
                        916*    951*   986*  1022*  1101*  1180*  1259*  1338*  1417*  1496*  1575*  1653*  1729*
                       1930    2125#
EXITER  010724         1922    1927#
FREEZ1  011652          462*    497*   532*   567*   602*   637*   672*   707*   742*   777*   812*   847*   882*
                        917*    952*   987*  1886   2126#
HALTS   010704         1892    1921#
HILIM   011320         1989*   2015   2031#
ICOUNT  011654          460*    495*   530*   565*   600*   635*   670*   705*   740*   775*   810*   845*   880*
                        915*    950*   985*  1021*  1100*  1179*  1258*  1337*  1416*  1495*  1574*  1652*  1728*
                       1870    2127#
INBJF   013036         1967    1995   2492#
INIFLG  011700          390     423    425*  2137#
INSTER= 104404          367#   2010
INSTR = 104403          366#    405    413    434
INSTRE  011132         1980#   2484
INSTRG  011040         1963#   2483
INSTR1  011052         1965#   1982
INSTR2  011140         1974    1983#
LAST    011704         1847*   1893   1895*  2139#  2436*
LIGHTS= 177570           55#   1850*
LIMITS  011252         2005    2015#
LINE  = 000020          449#    484#   519#   554#   589#   624#   659#   694#   729#   764#   799#   834#   869#
                        904#    939#   974#  1009#
LOBITS  011324         1991*   2019   2033#  2034
LOGICA  010420         1854#
LOLIM   011316         1988*   2017   2030#
LPCNT   011656         1869*   1870   1872*  2128#
MCRLF   012675         2039    2461#
MDATA   013062         2058    2069   2496#
MEPASS  012765         1846    2471#
MPFAIL  012700         2434    2462#
MQM     012671         1981    2460#
MR      012775          447    2473#
MREGAD  012635          414    2455#
MSG     011054         1963*   1966#
MTITLE  012520          389    2441#
```

```
MTSTPC  013001          435    2474#
MVECTO  012613          436    2451#
N     = 000001            1#
OCTASC= 104402          365#   1911    1918    2431
OCTASN  011326         2038#   2482
PARAM = 104405          368#    407     415     436
PARAMS  011142         1987#   2485
PARAMI  011172         1994#   2011
PARERR  011246         1997    1999    2001    2010#   2016    2018    2020
PASCNT  011642          385*   1849*   1850    2122#
PC    =%000007           50#   1854*
PFAIL   012354          348     383    2404#   2425
PFTAB   012510         2432    2438#
POPRO = 012600           64#   1926
POP1SP= 005726           62#
POP2SP= 022626           66#   1072    1151    1230    1309    1388    1467    1546    1625    1703
PS    = 177776           56#    381*    430*    459*    494*    529*    564*    599*    634*    669*    704*    739*    774*
                        809*    844*    879*    914*    949*    984*   1020*   1045*   1086*   1099*   1124*   1165*   1178*
                       1203*   1244*   1257*   1282*   1323*   1336*   1361*   1402*   1415*   1440*   1481*   1494*   1519*
                       1560*   1573*   1598*   1639*   1651*   1676*   1727*   1758*   2426*
PUSHRO= 010046           63#   1923
PUSH1S= 005746           61#
PUSH2S= 024646           65#
RBUF    012312         1033*   1034*   1065    1067    1069*   1112*   1113*   1144    1146    1148*   1191*   1192*   1223
                       1225    1227*   1270*   1271*   1302    1304    1306*   1349*   1350*   1381    1383    1385*   1428*
                       1429*   1460    1462    1464*   1507*   1508*   1539    1541    1543*   1586*   1587*   1618    1620
                       1622*   1664*   1665*   1696    1698    1700*   1740*   1741*   1795    1797    1799*   2399#
RESREG  010702         1917    1920#
RESTAR  012412         2412    2418#
RESTRT  010430         1852    1659#
RESOS = 104407          370#   1920
RETURN  011646          443*    448    1874*   1876    2124#   2437
RSOS    011544         2093#   2487
R0    =%000000           43#   1024*   1039*   1103*   1118*   1182*   1197*   1261*   1276*   1340*   1355*   1419*   1434*
                       1498*   1513*   1577*   1592*   1655*   1670*   1731*   1746*   1752*   1805*   1924*   2048*   2052*
                       2059    2089    2093*   2404    2424*
R1    =%000001           44#    396*    399*    401*    403     479*    480     514*    515     549*    550     584*    585
                        619*    620     654*    655     689*    690     724*    725     759*    760     794*    795     829*
                        830     864*    865     899*    900     934*    935     969*    970    1004*   1005    1025*   1033*
                       1037*   1059*   1060    1065    1104*   1112*   1116*   1138*   1139    1144    1183*   1191*   1195*
                       1217*   1218    1223    1262*   1270*   1274*   1296*   1297    1302    1341*   1349*   1353*   1375*
                       1376    1381    1420*   1428*   1432*   1454*   1455    1460    1499*   1507*   1511*   1533*   1534
                       1539    1578*   1586*   1590*   1612*   1613    1618    1656*   1664*   1668*   1690*   1691    1696
                       1732*   1740*   1744*   1787*   1788    1790    1795    1851*   1854    2040*   2042    2043    2044
                       2045    2088    2094*   2405    2423*
R2    =%000002           45#    397*    399     400*    402*   1027*   1034    1036*   1060*   1061*   1062*   1063    1064*
                       1065    1067    1069*   1106*   1113    1115*   1139*   1140*   1141*   1142    1143*   1144    1146
                       1148*   1185*   1192    1194*   1218*   1219*   1220*   1221    1222*   1223    1225    1227*   1264*
                       1271    1273*   1297*   1298*   1299*   1300    1301*   1302    1304    1306*   1343*   1350    1352*
                       1376*   1377*   1378*   1379    1380*   1381    1383    1385*   1422*   1429    1431*   1455*   1456*
                       1457*   1458    1459*   1460    1462    1464*   1501*   1508    1510*   1534*   1535*   1536*   1537
                       1538*   1539    1541    1543*   1580*   1587    1589*   1613*   1614*   1615*   1616    1617*   1618
                       1620    1622*   1658*   1665    1667*   1691*   1692*   1693*   1694    1695*   1696    1698    1700*
                       1734*   1741    1743*   1790*   1791*   1792*   1793    1794*   1795    1797    1799*   1817    2087
                       2095*   2406    2422*
R3    =%000003           46#    398*    401     402     464*    499*    534*    569*    604*    639*    674*    709*    744*
```

```
                          779*   814*   849*   884*   919*   954*   989*  1028*  1034*  1038*  1063*  1107*  1113*
                         1117*  1142*  1186*  1192*  1196*  1221*  1265*  1271*  1275*  1300*  1344*  1350*  1354*
                         1379*  1423*  1429*  1433*  1458*  1502*  1508*  1512*  1537*  1581*  1587*  1591*  1616*
                         1659*  1665*  1669*  1694*  1735*  1741*  1745*  1793*  1803   1814   1968*  1978*  2049*
                         2050*  2051*  2052   2058*  2059*  2064*  2067*  2086   2096*  2407   2421*
R4      =%000004           47*  1900*  1901*  1902*  1903*  1904*  1905   1906   1967*  1971*  1972*  1973   1995*
                         1996   1998   2000   2002*  2003   2004   2024*  2025*  2046*  2049   2053*  2054*  2055*
                         2085   2097*  2408   2420*
R5      =%000005           48*   465*   480    500*   515    535*   550    570*   585    605*   620    640*   655
                          675*   690    710*   725    745*   760    780*   795    815*   830    850*   865    885*
                          900    920*   935    955*   970    990*  1005   1067*  1146*  1225*  1304*  1383*  1462*
                         1541*  1620*  1698*  1797*  1898*  1899*  1900   1951*  1955   1958   1997*  1988   1989
                         1990   1991   1992   1993   1994*  2003*  2006*  2007*  2008*  2015   2017   2019   2025
                         2026*  2047*  2056*  2084   2098*  2409   2419*
SAVPC   011676           1934   2080*  2136*
SAVR0   011660           2089*  2093   2129*
SAVR1   011662           2088*  2094   2130*  2549
SAVR2   011664           2087*  2095   2131*
SAVR3   011666           2086*  2096   2132*  2551
SAVR4   011670           2085*  2097   2133*
SAVR5   011672           2084*  2098   2134*  2547
SAVSP   011674           2135*  2411*  2418
SAVDSP= 104406            369*  1897
SCOPE = 104400            363*   483    518    553    588    623    658    693    728    763    798    833    868
                          903    938    973   1008   1087   1166   1245   1324   1403   1482   1561   1640   1717
                         1819
SCOPER  010434           1863*  2480
SCOPE1= 104410            371*
SCOP1R  010534           1884*  2488
SP      =%000006           49*   382*   431*  1874   1876*  1886*  1893   1895   1898   1924   1930*  1940*  1941*
                         1942*  1943*  1944*  1945*  1946*  1947   1951   1952*  1963   1964*  1987   1993*  2040
                         2041*  2080   2404*  2405*  2406*  2407*  2408*  2409*  2410*  2411   2418*  2419   2420
                         2421   2422   2423   2424   2427*
SPACNT= 011501           2044*  2062   2065*  2075*
STACK = 013564             57*   382    431   2427
START   001000            355    381*
STFLG   011702            384*   444    446*  2138*
SVOS    011512           2084*
SVOSP   011504           2080*  2486
SWR     = 177570           54*   394    432   1863   1865   1867   1884   1891   1921   1928
SW00    = 000001           37*   394
SW01    = 000002           36*   432
SW02    = 000004           35*
SW03    = 000010           34*
SW04    = 000020           33*
SW05    = 000040           32*
SW06    = 000100           31*
SW08    = 000400           30*
SW09    = 001000           29*  1884
SW10    = 002000           28*  1863   1928
SW11    = 004000           27*  1867
SW12    = 010000           26*
SW13    = 020000           25*  1891
SW14    = 040000           24*  1865
SW15    = 100000           23*
TBUF    011712           1029   1108   1187   1266   1345   1424   1503   1582   1660   1736   1815   2142*
```

```
TDAT   = 000400          2142#   2143    2144#   2145#   2146#   2147#   2148#   2149#   2150#   2151#   2152#   2153#   2154#
                         2155#   2156#   2157#   2158#   2159#   2160#   2161#   2162#   2163#   2164#   2165#   2166#   2167#
                         2168#   2169#   2170#   2171#   2172#   2173#   2174#   2175#   2176#   2177#   2178#   2179#   2180#
                         2181#   2182#   2183#   2184#   2185#   2186#   2187#   2188#   2189#   2190#   2191#   2192#   2193#
                         2194#   2195#   2196#   2197#   2198#   2199#   2200#   2201#   2202#   2203#   2204#   2205#   2206#
                         2207#   2208#   2209#   2210#   2211#   2212#   2213#   2214#   2215#   2216#   2217#   2218#   2219#
                         2220#   2221#   2222#   2223#   2224#   2225#   2226#   2227#   2228#   2229#   2230#   2231#   2232#
                         2233#   2234#   2235#   2236#   2237#   2238#   2239#   2240#   2241#   2242#   2243#   2244#   2245#
                         2246#   2247#   2248#   2249#   2250#   2251#   2252#   2253#   2254#   2255#   2256#   2257#   2258#
                         2259#   2260#   2261#   2262#   2263#   2264#   2265#   2266#   2267#   2268#   2269#   2270#   2271#
                         2272#   2273#   2274#   2275#   2276#   2277#   2278#   2279#   2280#   2281#   2282#   2283#   2284#
                         2285#   2286#   2287#   2288#   2289#   2290#   2291#   2292#   2293#   2294#   2295#   2296#   2297#
                         2298#   2299#   2300#   2301#   2302#   2303#   2304#   2305#   2306#   2307#   2308#   2309#   2310#
                         2311#   2312#   2313#   2314#   2315#   2316#   2317#   2318#   2319#   2320#   2321#   2322#   2323#
                         2324#   2325#   2326#   2327#   2328#   2329#   2330#   2331#   2332#   2333#   2334#   2335#   2336#
                         2337#   2338#   2339#   2340#   2341#   2342#   2343#   2344#   2345#   2346#   2347#   2348#   2349#
                         2350#   2351#   2352#   2353#   2354#   2355#   2356#   2357#   2358#   2359#   2360#   2361#   2362#
                         2363#   2364#   2365#   2366#   2367#   2368#   2369#   2370#   2371#   2372#   2373#   2374#   2375#
                         2376#   2377#   2378#   2379#   2380#   2381#   2382#   2383#   2384#   2395#   2386#   2387#   2388#
                         2389#   2390#   2391#   2392#   2393#   2394#   2395#   2396#   2397#   2398#
TDATA    011710           469*    474     504*    509     539*    544     574*    579     609*    614     644*    649     679*
                          684    714*    719     749*    754     784*    789     819*    824     854*    859     889*    894
                          924*    929    959*    964     994*    999     2141#
TEMP     013050          2048    2428*   2429*   2494#
TKCSR    011576          1969    2102#
TKDBR    011600          1971    1975    2103#
TPCSR    011602          1953    1976    2104#
TPDBR    011604          1958*   1975*   2105#
TRPOK    010766          1943#
TRPSRV   010754           352    1940#
TRPTAB   013014          1945    2480#
TYPDAT   010670          1910    1916#
TYPE   = 104401           364#    389     447     1845    1914    1965    1980    2038    2068    2433
TYPER    011006          1951#   2481
TYPMSG   010652          1908    1911#
T1       001274           443     459#
T10      002552           704#
T11      002714           739#
T12      003056           774#
T13      003220           809#
T14      003362           844#
T15      003524           879#
T16      003666           914#
T17      004030           949#
T2       001436           494#
T20      004172           984#
T21      004334          1020#
T22      004640          1099#
T23      005144          1178#
T24      005450          1257#
T25      005754          1336#
T26      006260          1415#
T27      006564          1494#
T3       001600           529#
T30      007070          1573#
T31      007374          1651#
```

```
T32      007672          1727#
T4       001742           564#
T5       002104           599#
T6       002246           634#
T7       002410           669#
VEC1     001060           391     394#
VEC2     001070           393     396#
WRDCNT   011476          2042*   2070*   2073#
X      = 000000             1#
XBIT   = 000001           449#
XLINE  = 000000           449#
XN     = 000033             1#     459     463#     494     498#     529     533#     564     568#     599     603#     634     638#
                           669     673#     704     709#     739     743#     774     778#     809     813#     844     848#     879
                           883#     914     918#     949     953#     984     988#    1020    1023#    1099    1102#    1178    1181#
                          1257    1260#    1336    1339#    1415    1418#    1494    1497#    1573    1576#    1651    1654#    1727
                          1730#
Y      = 000011             1#     363     364#     365#     366#     367#     368#     369#     370#     371#     372#
.      = 013366            87#      88      90      92      94      96      98     100     102     104     106     108     110
                           112     114     116     118     120     122     124     126     128     130     132     134     136
                           138     140     142     144     146     148     150     152     154     156     158     160     162
                           164     166     168     170     172     174     176     178     180     182     184     186     188
                           190     192     194     196     198     200     202     204     206     208     210     212     214
                           216     218     220     222     224     226     228     230     232     234     236     238     240
                           242     244     246     248     250     252     254     256     258     260     262     264     266
                           268     270     272     274     276     278     280     282     284     286     288     290     292
                           294     296     298     300     302     304     306     308     310     312     314     316     318
                           320     322     324     326     328     330     332     334     336     338     340     342     347#
                           354#    372#     481     516     551     586     621     656     691     726     761     796     831
                           866     901     936     971    1006    1057    1066    1083    1085    1136    1145    1162    1164
                          1215    1224    1241    1243    1294    1303    1320    1322    1373    1382    1399    1401    1452
                          1461    1478    1480    1531    1540    1557    1559    1610    1619    1636    1638    1688    1697
                          1714    1716    1765    1772    1775    1796    2400#    2414    2430    2493#    2495#    2497#    2541#
```

```
CODEM1    372#    421
HLT        67#    482     517     552     587     622     657     692     727     762     797     832     867     902     937
          972    1007    1058    1068    1071    1137    1147    1150    1216    1226    1229    1295    1305    1308    1374
         1384    1397    1453    1463    1466    1532    1542    1545    1611    1621    1624    1689    1699    1702    1766
         1773    1776    1778    1798
POPOVE    449#    484     519     554     589     624     659     694     729     764     799     834     869     904     939
          974
SUPERC    449#   1009    1088    1167    1246    1325    1404    1483    1562
TRPDEF      1#    363     364     365     366     367     368     369     370     371
TS          1#    459     494     529     564     599     634     669     704     739     774     809     844     879     914
          949     984    1020    1099    1178    1257    1336    1415    1494    1573    1651    1727
.BUFFE      1#   2489
.CATCH      1#     86
.ENDCO      1#   2552
.EOP        1#   1838
.ERROR      1#   1888
.ERRTA      1#   2498
.HEADE      1#
.INSTR      1#   1960
.MSG        1#   2441
.OCTAS      1#   2035
.PARAM      1#   1984
.PFAIL      1#   2401
.POINT      1#   2100
.RESRE      1#   2091
.SAVRE      1#   2077
.SCOPE      1#   1859
.SCOPI      1#   1881
.SETVE      1#    344
.START      1#    372
.SYMBO      1#     20
.TRPDE      1#    358
.TRPSR      1#   1935
.TRPTA      1#   2477
.TYPER      1#   1948
.VARIA      1#   2119
```

ADD	401	402	1037	1038	1116	1117	1195	1196	1274	1275	1353	1354	1432	1433	1511
	1512	1590	1591	1668	1669	1744	1745	1904	1945	1952	1964	2026	2041	2052	
ASL	1064	1143	1222	1301	1380	1459	1538	1617	1695	1794	1901	1902	1943	2006	2007
	2008														
ASR	2053	2054	2055												
BEQ	395	433	481	516	551	586	621	656	691	726	761	796	831	866	901
	936	971	1006	1057	1066	1136	1145	1215	1224	1294	1303	1373	1382	1452	1461
	1531	1540	1610	1619	1688	1697	1775	1796	1852	1879	1885	1894	1908	1917	1929
	1974	1997	2005	2063											
BGT	2001														
BHI	2016														
BIC	1062	1141	1220	1299	1378	1457	1536	1615	1693	1792	1813	1903	1944	2050	
BICB	1972	2002													
BIS	1814	1817													
BISB	2003														
BIT	394	432	1056	1135	1214	1293	1372	1451	1530	1609	1687	1774	1863	1865	1867
	1884	1891	1928												
BITB	2019														
BLO	2018														
BLT	1999														
BMI	1765														
BNE	391	404	424	445	1040	1083	1085	1119	1162	1164	1198	1241	1243	1277	1320
	1322	1356	1399	1401	1435	1478	1480	1514	1557	1559	1593	1636	1638	1671	1714
	1716	1747	1800	1802	1804	1806	1809	1811	1864	1866	1868	1871	1892	1910	1956
	1979	2020	2028	2057	2061	2066	2071	2430							
BPL	478	513	548	583	618	653	688	723	758	793	828	863	898	933	968
	1003	1055	1134	1213	1292	1371	1450	1529	1608	1686	1772	1789	1922	1954	1970
	1977														
BR	393	442	1074	1153	1232	1311	1390	1469	1548	1627	1705	1759	1777	1812	1818
	1680	1959	1982	2009	2011	2414									
CLR	384	385	386	387	388	400	1025	1045	1104	1124	1183	1203	1262	1282	1341
	1361	1420	1440	1499	1519	1578	1598	1656	1676	1732	1753	1758	1847	1848	1872
	1873	1896	1994	2428	2435	2436									
CLRB	1033	1112	1191	1270	1349	1428	1507	1586	1664	1740	2067				
CMP	403	480	515	550	585	620	655	690	725	760	795	830	865	900	935
	970	1005	1065	1144	1223	1302	1381	1460	1539	1618	1696	1870	1893	2015	2017
CMPB	1795	1973	1996	1998	2000	2004									
COM	425	446													
DEC	1039	1118	1197	1276	1355	1434	1513	1592	1670	1746	1805	1978	2056	2070	
DECB	2027	2060	2065												
EMT	67														
HALT	89	91	93	95	97	99	101	103	105	107	109	111	113	115	117
	119	121	123	125	127	129	131	133	135	137	139	141	143	145	147
	149	151	153	155	157	159	161	163	165	167	169	171	173	175	177
	179	181	183	185	187	189	191	193	195	197	199	201	203	205	207
	209	211	213	215	217	219	221	223	225	227	229	231	233	235	237
	239	241	243	245	247	249	251	253	255	257	259	261	263	265	267
	269	271	273	275	277	279	281	283	285	287	289	291	293	295	297
	299	301	303	305	307	309	311	313	315	317	319	321	323	325	327
	329	331	333	335	337	339	341	343	1925	2413					
INC	422	1035	1036	1114	1115	1193	1194	1272	1273	1351	1352	1430	1431	1509	1510
	1588	1589	1666	1667	1742	1743	1849	1869	1927	2429					
INCB	1069	1148	1227	1306	1385	1464	1543	1622	1700	1799					
JMP	355	448	1046	1125	1204	1283	1362	1441	1520	1599	1677	1820	1858	1947	2437
JSR	1854														
MOV	381	382	383	396	397	398	399	421	430	431	443	459	460	461	462

	463	464	465	469	470	471	473	474	475	476	479	494	495	496	497
	498	499	500	504	505	506	508	509	510	511	514	529	530	531	532
	533	534	535	539	540	541	543	544	545	546	549	564	565	566	567
	568	569	570	574	575	576	578	579	580	581	584	599	600	601	602
	603	604	605	609	610	611	613	614	615	616	619	634	635	636	637
	638	639	640	644	645	646	648	649	650	651	654	669	670	671	672
	673	674	675	679	680	681	683	684	685	686	689	704	705	706	707
	708	709	710	714	715	716	718	719	720	721	724	739	740	741	742
	743	744	745	749	750	751	753	754	755	756	759	774	775	776	777
	778	779	780	784	785	786	788	789	790	791	794	809	810	811	812
	813	814	815	819	820	821	823	824	825	826	829	844	845	846	847
	848	849	850	854	855	856	858	859	860	861	864	879	880	881	882
	883	884	885	889	890	891	893	894	895	896	899	914	915	916	917
	918	919	920	924	925	926	928	929	930	931	934	949	950	951	952
	953	954	955	959	960	961	963	964	965	966	969	984	985	986	987
	988	989	990	994	995	996	998	999	1000	1001	1004	1020	1021	1022	1023
	1024	1027	1028	1029	1030	1031	1041	1042	1043	1044	1059	1060	1063	1067	1073
	1086	1099	1100	1101	1102	1103	1106	1107	1108	1109	1110	1120	1121	1122	1123
	1138	1139	1142	1146	1152	1165	1178	1179	1180	1181	1182	1185	1186	1187	1188
	1189	1199	1200	1201	1202	1217	1218	1221	1225	1231	1244	1257	1258	1259	1260
	1261	1264	1265	1266	1267	1268	1278	1279	1280	1281	1296	1297	1300	1304	1310
	1323	1336	1337	1338	1339	1340	1343	1344	1345	1346	1347	1357	1358	1359	1360
	1375	1376	1379	1383	1389	1402	1415	1416	1417	1418	1419	1422	1423	1424	1425
	1426	1436	1437	1438	1439	1454	1455	1458	1462	1468	1481	1494	1495	1496	1497
	1498	1501	1502	1503	1504	1505	1515	1516	1517	1518	1533	1534	1537	1541	1547
	1560	1573	1574	1575	1576	1577	1580	1581	1582	1583	1584	1594	1595	1596	1597
	1612	1613	1616	1620	1626	1639	1651	1652	1653	1654	1655	1658	1659	1660	1661
	1662	1672	1673	1674	1675	1690	1691	1694	1698	1704	1727	1728	1729	1730	1731
	1734	1735	1736	1737	1738	1748	1749	1750	1751	1752	1755	1756	1757	1787	1790
	1793	1797	1807	1815	1816	1850	1851	1874	1876	1886	1895	1898	1900	1905	1906
	1913	1924	1930	1940	1942	1946	1951	1963	1967	1968	1987	1988	1989	1990	1993
	1995	2024	2025	2040	2042	2045	2046	2048	2049	2058	2080	2084	2085	2086	2087
	2088	2089	2093	2094	2095	2096	2097	2098	2404	2405	2406	2407	2408	2409	2410
	2411	2412	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427			
MOVB	1034	1113	1192	1271	1350	1429	1508	1587	1665	1741	1958	1971	1975	1991	1992
	2043	2044	2047	2052	2059	2064									
NOP	1855	1856	1857												
RESET	1853														
RETURN	439	2440													
RTI	1070	1149	1228	1307	1386	1465	1544	1623	1701	1767	1780	1875	1877	1887	1931
	1957	1983	2029	2072	2090	2099									
SUB	1899	1941													
SWAB	1061	1140	1219	1298	1377	1456	1535	1614	1692	1791					
TRAP	363	364	365	366	367	368	369	370	371						
TST	390	423	444	1082	1161	1240	1319	1398	1477	1556	1635	1713	1764	1788	1801
	1803	1808	1878	1907	1909	1916	1921								
TSTB	477	512	547	582	617	652	687	722	757	792	827	862	897	932	967
	1002	1054	1084	1133	1163	1212	1242	1291	1321	1370	1400	1449	1479	1528	1558
	1607	1637	1685	1715	1771	1810	1953	1955	1969	1976	2062				
.ASCII	2526														
.ASCIZ	2441	2451	2455	2460	2461	2462	2471	2473	2474	2514	2518	2523	2528	2532	2536
.BYTE	411	412	419	420	440	441	1933	2142	2143	2144	2145	2146	2147	2148	2149
	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164
	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179
	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194
	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209

```
          2210   2211   2212   2213   2214   2215   2216   2217   2218   2219   2220   2221   2222   2223   2224
          2225   2226   2227   2228   2229   2230   2231   2232   2233   2234   2235   2236   2237   2238   2239
          2240   2241   2242   2243   2244   2245   2246   2247   2248   2249   2250   2251   2252   2253   2254
          2255   2256   2257   2258   2259   2260   2261   2262   2263   2264   2265   2266   2267   2268   2269
          2270   2271   2272   2273   2274   2275   2276   2277   2278   2279   2280   2281   2282   2283   2284
          2285   2286   2287   2288   2289   2290   2291   2292   2293   2294   2295   2296   2297   2298   2299
          2300   2301   2302   2303   2304   2305   2306   2307   2308   2309   2310   2311   2312   2313   2314
          2315   2316   2317   2318   2319   2320   2321   2322   2323   2324   2325   2326   2327   2328   2329
          2330   2331   2332   2333   2334   2335   2336   2337   2338   2339   2340   2341   2342   2343   2344
          2345   2346   2347   2348   2349   2350   2351   2352   2353   2354   2355   2356   2357   2358   2359
          2360   2361   2362   2363   2364   2365   2366   2367   2368   2369   2370   2371   2372   2373   2374
          2375   2376   2377   2378   2379   2380   2381   2382   2383   2384   2385   2386   2387   2388   2389
          2390   2391   2392   2393   2394   2395   2396   2397   2546   2548   2550
.ENABL      20
.END      2553
.ENDC      393    394    421    423    463    498    533    568    603    638    673    708    743    778    813
           848    883    918    953    988   1023   1031   1102   1110   1181   1188   1189   1260   1267   1268
          1339   1346   1347   1418   1425   1426   1497   1504   1505   1576   1583   1584   1654   1730
.EQUIV      67
.EVEN     2398   2476   2541
.IF        391    393    421    462    497    532    567    602    637    672    707    742    777    812    847
           882    917    952    987   1023   1030   1031   1102   1109   1110   1181   1188   1189   1260   1267
          1268   1339   1346   1347   1418   1425   1426   1497   1504   1576   1583   1654   1730
.IFF       393    394
.IIF       380
.IRP      2106   2140
.LIST        1     20    364    365    366    367    368    369    370    371    372    449    463    484    498
           519    533    554    568    589    603    624    638    659    673    694    708    729    743    764
           778    799    813    834    848    869    883    904    918    939    953    974    988   1009   1023
          1102   1181   1260   1339   1418   1497   1576   1654   1730   2142   2144   2145   2146   2147   2148
          2149   2150   2151   2152   2153   2154   2155   2156   2157   2158   2159   2160   2161   2162   2163
          2164   2165   2166   2167   2168   2169   2170   2171   2172   2173   2174   2175   2176   2177   2178
          2179   2180   2181   2182   2183   2184   2185   2186   2187   2188   2189   2190   2191   2192   2193
          2194   2195   2196   2197   2198   2199   2200   2201   2202   2203   2204   2205   2206   2207   2208
          2209   2210   2211   2212   2213   2214   2215   2216   2217   2218   2219   2220   2221   2222   2223
          2224   2225   2226   2227   2228   2229   2230   2231   2232   2233   2234   2235   2236   2237   2238
          2239   2240   2241   2242   2243   2244   2245   2246   2247   2248   2249   2250   2251   2252   2253
          2254   2255   2256   2257   2258   2259   2260   2261   2262   2263   2264   2265   2266   2267   2268
          2269   2270   2271   2272   2273   2274   2275   2276   2277   2278   2279   2280   2281   2282   2283
          2284   2285   2286   2287   2288   2289   2290   2291   2292   2293   2294   2295   2296   2297   2298
          2299   2300   2301   2302   2303   2304   2305   2306   2307   2308   2309   2310   2311   2312   2313
          2314   2315   2316   2317   2318   2319   2320   2321   2322   2323   2324   2325   2326   2327   2328
          2329   2330   2331   2332   2333   2334   2335   2336   2337   2338   2339   2340   2341   2342   2343
          2344   2345   2346   2347   2348   2349   2350   2351   2352   2353   2354   2355   2356   2357   2358
          2359   2360   2361   2362   2363   2364   2365   2366   2367   2368   2369   2370   2371   2372   2373
          2374   2375   2376   2377   2378   2379   2380   2381   2382   2383   2384   2385   2386   2387   2388
          2389   2390   2391   2392   2393   2394   2395   2396   2397   2398
.MACRO       1    372    449
.NLIST       1     20    364    365    366    367    368    369    370    371    372    449    463    484    498
           519    533    554    568    589    603    624    638    659    673    694    708    729    743    764
           778    799    813    834    848    869    883    904    918    939    953    974    988   1009   1023
          1102   1181   1260   1339   1418   1497   1576   1654   1730   2142   2144   2145   2146   2147   2148
          2149   2150   2151   2152   2153   2154   2155   2156   2157   2158   2159   2160   2161   2162   2163
          2164   2165   2166   2167   2168   2169   2170   2171   2172   2173   2174   2175   2176   2177   2178
          2179   2180   2181   2182   2183   2184   2185   2186   2187   2188   2189   2190   2191   2192   2193
          2194   2195   2196   2197   2198   2199   2200   2201   2202   2203   2204   2205   2206   2207   2208
```

```
          2209    2210    2211    2212    2213    2214    2215    2216    2217    2218    2219    2220    2221    2222    2223
          2224    2225    2226    2227    2228    2229    2230    2231    2232    2233    2234    2235    2236    2237    2238
          2239    2240    2241    2242    2243    2244    2245    2246    2247    2248    2249    2250    2251    2252    2253
          2254    2255    2256    2257    2258    2259    2260    2261    2262    2263    2264    2265    2266    2267    2268
          2269    2270    2271    2272    2273    2274    2275    2276    2277    2278    2279    2280    2281    2282    2283
          2284    2285    2286    2287    2288    2289    2290    2291    2292    2293    2294    2295    2296    2297    2298
          2299    2300    2301    2302    2303    2304    2305    2306    2307    2308    2309    2310    2311    2312    2313
          2314    2315    2316    2317    2318    2319    2320    2321    2322    2323    2324    2325    2326    2327    2328
          2329    2330    2331    2332    2333    2334    2335    2336    2337    2338    2339    2340    2341    2342    2343
          2344    2345    2346    2347    2348    2349    2350    2351    2352    2353    2354    2355    2356    2357    2358
          2359    2360    2361    2362    2363    2364    2365    2366    2367    2368    2369    2370    2371    2372    2373
          2374    2375    2376    2377    2378    2379    2380    2381    2382    2383    2384    2385    2386    2387    2388
          2389    2390    2391    2392    2393    2394    2395    2396    2397    2398
.PAGE       39      86     344     372
.REPT       88     449    2143
.TITLE      20


ERRORS DETECTED:  0
DEFAULT GLOBALS GENERATED:  0

*,DZDHGB.SEQ←UTIL1,DZDHGB
RUN-TIME: 13 22 4 SECONDS
RUN-TIME RATIO: 166/40=4.1
CORE USED:  11K  (21 PAGES)
```

K05

Spooler runtime 10 Seconds, 44 KCS, 264 disk reads, 4 disk writes, 61 pages

[redacted]

[redacted]

[redacted]

[redacted]