

LPC11

INTERFACE DIAGNOSTIC TEST
MD-11-DZLPB-B

EP-DZLPB-B-DL
COPYRIGHT © 1973
FICHE 1 OF 1

JUN 1978
digital
MADE IN USA

Microfiche frame 1	Microfiche frame 2	Microfiche frame 3
Microfiche frame 4	Microfiche frame 5	Microfiche frame 6
Microfiche frame 7	Microfiche frame 8	Microfiche frame 9
Microfiche frame 10	Microfiche frame 11	Microfiche frame 12
Microfiche frame 13	Microfiche frame 14	Microfiche frame 15
Microfiche frame 16	Microfiche frame 17	Microfiche frame 18
Microfiche frame 19	Microfiche frame 20	Microfiche frame 21
Microfiche frame 22	Microfiche frame 23	Microfiche frame 24
Microfiche frame 25	Microfiche frame 26	Microfiche frame 27
Microfiche frame 28	Microfiche frame 29	Microfiche frame 30
Microfiche frame 31	Microfiche frame 32	Microfiche frame 33
Microfiche frame 34	Microfiche frame 35	Microfiche frame 36
Microfiche frame 37	Microfiche frame 38	Microfiche frame 39
Microfiche frame 40	Microfiche frame 41	Microfiche frame 42
Microfiche frame 43	Microfiche frame 44	Microfiche frame 45
Microfiche frame 46	Microfiche frame 47	Microfiche frame 48



IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZLFB-B-0
PRODUCT NAME: LPC11 INTERFACE
DIAGNOSTIC TEST
DATE CREATED: FEBRUARY 25, 1973
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: R.B. BARNES

COPYRIGHT © 1973
DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT

THIS DIAGNOSTIC WILL EXERCISE ALL LOGIC FUNCTIONS AND DATA CAPABILITIES OF THE LPC11 INTERFACE. THE PROGRAM SHOULD BE STARTED AT LOCATION 200 AND WILL TYPE OUT THE PROGRAM NAME AND REQUEST INPUT OF VECTOR ADDRESS, REGISTER ADDRESS, AND INTERFACE MODE SETTING.

THE PROGRAM CONSISTS OF FOUR (4) TEST GROUPS: LOGIC TESTS, DATA TESTS, MAINTENANCE MODE TESTS, AND LINE COUNT MODE. THE LOGIC AND DATA TESTS ARE PERFORMED SEQUENTIALY; (EXCEPT LOGIC TEST 22 WHICH IS ENTERED VIA SWITCH TEN). THE MAINTENANCE MODE AND LINE COUNT MODE ARE ENTERED THROUGH THE CONSOLE SWITCHES.

THE PROGRAM IS DESIGNED TO PROVIDE THE OPERATOR WITH AS MUCH FLEXABILITY AS POSSIBLE THROUGH THE USE OF THE CONSOLE SWITCH REGISTER. USE OF THE SWITCHES PROVIDES FOR CONTROL OF ERROR PRINT, STOP ON ERROR, ITERATION OF DATA PATTERNS, REPEAT LOOP, ERROR STATUS BIT TESTING, MAINTENANCE MODE ENTRY, MAINTENANCE MODE INTERRUPT RECOGNITION, AND LINE COUNT MODE ENTRY.

2. REQUIREMENTS (EQUIPMENT)

- A. PDP-11/25,15,20,45
- B. TELETYPE
- C. LPC11 INTERFACE
- D. PDP-11/45 AND KL11-L LINE CLOCK
ARE REQUIRED FOR THE LINE COUNT MODE.

3. LOADING PROCEDURE

- A. USE STANDARD PROCEDURE FOR LOADING BINARY TAPES

4. STARTING PROCEDURE

- A. LOAD AND START 200 TO ENTER NEW VECTOR AND REGISTER ADDRESSES; SET CONSOLE SWITCHES BEFORE TYPING 'G'.
- B. LOAD AND START 204 TO BYPASS ENTRY OF NEW ADDRESSES; SET CONSOLE SWITCHES BEFORE TYPING 'G'.
- C. DUE TO FLOATABLE VECTOR AND REGISTER USAGE, A START AT 200 MUST BE DONE ON THE INITIAL RUN. THEREAFTER 204 MAY BE USED.
- D. THE PROGRAM WILL TYPE "END OF TEST" UPON COMPLETION OF A SINGLE PASS AND MAY BE RESTARTED BY PRESSING THE CONTINUE SWITCH.
- E. A POWER FAIL RESTART IS PROVIDED AND WILL TYPE A POWER FAIL MESSAGE BEFORE RESTARTING AT LOC 200(0).

5. CONSOLE SWITCH SETTINGS

- A. THE CONSOLE SWITCHES MAY BE SET TO ANY CONFIGURATION AND ARE APPLICABLE TO ALL TESTS.
- B. ANY SWITCH MAY BE CHANGED DYNAMICALLY.
- C. IF NO SWITCHES ARE SET THE PROGRAM WILL RUN IN THE "NORMAL" MANNER. THE SWITCHES SHOULD BE SET FOR ANY VARIATION DESIRED.

SWITCHES:

- SW 15: 1=NO ERROR PRINTS
2=PRINT ALL ERRORS

- SW 14: 1=STOP ON ERROR
2=CONTINUE ON ERROR

- SW 13: 1=LOOP MODE
0=SINGLE PASS

- SW 12: 1=INHIBIT DATA ITERATIONS
0=DO NOT INHIBIT ITERATIONS

- SW 11: 1=GO TO MAINTENANCE MODE
2=DO NOT ENTER MAINTENANCE MODE

- SW 10: 1=DO ERROR BIT TEST
2=DO NOT DO ERROR BIT TEST

- SW 9: 1=USE TESTER INTERRUPT FOR MAINTENANCE MODE
2=USE CONTROLLER INTERRUPT

- SW 8: 1=ENTER LINE COUNT MODE (PDP-11/45 ONLY)
2=DO NOT ENTER LINE COUNT MODE

6. ERROR PRINTOUTS

- A. THERE ARE THREE (3) TYPES OF ERROR PRINTOUTS:
 - 1. ILLEGAL VECTOR OR REGISTER ENTRY
 - 2. LOGIC TEST ERROR
 - 3. DATA TEST ERROR

- B. ANY ODD VECTOR OR REGISTER ADDRESS, OR OUT OF BOUNDS ENTRY (0-500 FOR VECTORS; 760000-777600 FOR REGISTERS) WILL RESULT IN AN ERROR PRINT AND REENTRY REQUEST, REGARDLESS OF CONSOLE SWITCH 15 SETTING.

- C. ANY LOGIC ERROR WILL RESULT IN A PRINT OUT OF THE TEST NUMBER AND ENGLISH LANGUAGE DESCRIPTION OF THE ERROR.

- D. ANY DATA ERROR WILL RESULT IN A PRINT OUT OF THE EXPECTED DATA AND THE RECEIVED DATA IN BIT FORMAT.

- E. CONSOLE SWITCH 15 CONDITIONS ERROR PRINTOUTS. BY SETTING SWITCH 15 TO A ONE, ALL ERROR MESSAGES ARE INHIBITED EXCEPT FOR ENTRY OF AN ILLEGAL VECTOR OR REGISTER ADDRESS.
- F. CONSOLE SWITCH 14 CONDITIONS STOP/CONTINUE ON ERROR IF A LOGIC TEST ERROR HALT IS ENCOUNTERED, THE PROGRAM MAY BE RESTARTED BY PRESSING THE CONTINUE SWITCH.
IF A DATA TEST ERROR HALT IS ENCOUNTERED, THE PROGRAM MAY BE CONTINUED BY PRESSING THE CONTINUE SWITCH.

7. TEST DESCRIPTIONS

- A. GROUP 11 LOGIC TESTS
THIS GROUP OF 22(8) TESTS WILL CHECK ALL USABLE BITS OF BOTH THE TESTER AND CONTROLLER STATUS REGISTERS FOR SET AND RESET. THE ERROR BIT (19) IN THE CONTROLLER STATUS REGISTERS ONLY SETS WHEN THE INTERFACE IS OFF LINE, THEREFORE REQUIRING MANUAL INTERVENTION BY AN OPERATOR. CONSOLE SWITCH 18 CONDITIONS ENTRY TO THIS TEST. WHEN ENTERED A MESSAGE WILL BE TYPED REQUESTING THE INTERFACE BE PLACED 'OFF LINE', WHEN COMPLETED ANOTHER MESSAGE WILL BE TYPED REQUESTING A RETURN TO "ON LINE," THE ERROR BIT WILL BE CHECKED FOR BOTH SET AND RESET.

- TEST11 ASSURE THAT ALL BITS OF THE CONTROLLER STATUS REGISTER EXCEPT READY CAN BE CLEARED, BITS 0-9 ARE NOT CHECKED AS THEY MAY BE USED FOR INTERFACE TYPE IDENTIFICATION AT A LATER DATE.
- TEST21 ASSURE THAT ALL BITS OF THE TESTER STATUS REGISTER EXCEPT READY CAN BE CLEARED.
- TEST31 ASSURE THAT THE CONTROLLER READY BIT WAS CLEARED BY INIT.
- TEST41 ASSURE THAT THE TESTER READY BIT WAS CLEARED BY INIT.
- TEST51 ASSURE THAT THE CONTROLLER READY BIT REMAINS RESET WHEN A CHARACTER IS LOADED INTO THE CONTROLLER BUFFER.
- TEST61 ASSURE THAT THE TESTER READY BIT DOES SET UPON COMPLETION OF DATA TRANSFER FROM CONTROLLER BUFFER TO TESTER BUFFER.
- TEST71 ASSURE THAT THE GO BIT OF THE TESTER STATUS REGISTER RESETS IMMEDIATELY AFTER BEING SET.
- TEST101 ASSURE THAT THE TESTER READY BIT IS NOT RESET BY THE SETTING OF THE GO BIT.

- TEST11: ASSURE THAT THE CONTROLLER READY BIT SETS AFTER THE GO BIT IS SET TO COMPLETE THE CYCLE.
- TEST12: ASSURE THAT THE TESTER READY BIT IS NOT RESET AT THE END OF THE CYCLE.
- TEST13: ASSURE THAT THE CONTROLLER READY BIT IS NOT RESET AT AT THE END OF THE CYCLE.
- TEST14: ASSURE THAT THE TESTER INTERRUPT IS GENERATED AND RECOGNIZED AFTER COMPLETION OF A DATA TRANSFER CYCLE.
- TEST15: ASSURE THAT THE CONTROLLER INTERRUPT IS GENERATED AND RECOGNIZED AT THE END OF A CYCLE.
- TEST16: ASSURE THAT THE TESTER INTERRUPT WILL NOT BE RECOGNIZED WHEN THE INTERRUPT ENABLE BIT IS NOT SET.
- TEST17: ASSURE THAT THE TESTER INTERRUPT WILL NOT BE RECOGNIZED AT A PRIORITY LEVEL ABOVE FOUR (4).
- TEST20: ASSURE THAT THE CONTROLLER INTERRUPT IS NOT RECOGNIZED WHEN THE INTERRUPT ENABLE BIT IS NOT SET.
- TEST21: ASSURE THAT THE CONTROLLER INTERRUPT IS NOT RECOGNIZED AT A PRIORITY LEVEL ABOVE FOUR (4).
- TEST22: THIS TEST, ENTERED VIA CONSOLE SWITCH TEN (10), WILL CHECK THE ERROR STATUS BIT, BIT FIFTEEN (15), FOR THE PROPER STATE IN EACH MODE OF THE INTERFACE. OPERATOR INTERVENTION WILL BE REQUESTED ON THE TELETYPE TO SET THE MODE SWITCHES ON THE INTERFACE TO EACH OF THE FOUR STATES:
1. ON LINE = NORMAL MODE: BIT 15 = 0
 2. OFF LINE = NORMAL MODE: BIT 15 = 1
 3. OFF LINE = TEST MODE: BIT 15 = 1
 4. ON LINE = TEST MODE: BIT 15 = 1

B. GROUP 21 DATA TESTS

THIS GROUP OF 2 TESTS WILL PERFORM DATA CHECKS BY WRAPPING DATA AROUND FROM THE CONTROLLER BUFFER TO THE TESTER BUFFER AND COMPARING FOR ERRORS.

- TEST 1: WRAP = PATTERNS OF ALL ZEROS, ALL ONES, A ZERO WALKING FROM RIGHT TO LEFT IN A FIELD OF ONES, AND A ONE WALKING FROM RIGHT TO LEFT IN A FIELD OF ZEROS. THE DATA CAN BE OBSERVED IN THE INDICATOR LIGHTS ON THE INTERFACE PANEL.
- TEST 2: WRAP PATTERNS OF ALL POSSIBLE COMBINATIONS OF THE 8 BIT CHARACTER (2⁸ = 256).

THE PATTERNS IN TEST 1 ARE EACH REPEATED 512 TIMES AND EACH SET OF ALL BIT COMBINATIONS IS REPEATED 6 TIMES IN TEST 2. THESE ITERATIONS MAY BE INHIBITED VIA CONSOLE SWITCH 12.

- C. GROUP 3: MAINTENANCE MODE
THIS GROUP, ENTERED VIA CONSOLE SWITCH 11, WILL RUN CONTINUOUSLY AND SHOULD BE USED FOR DEBUGGING AND DELAY ADJUSTMENTS. A FULL CYCLE OF DATA WRAP AND INTERRUPT AT END OF CYCLE IS PERFORMED. THE INTERRUPT RECOGNIZED, TESTER OR CONTROLLER, IS VARIABLE DYNAMICALLY VIA CONSOLE SWITCH 9. THE INTERRUPT ENABLE INDICATORS ON THE INTERFACE PANEL WILL SHOW WHICH IS BEING USED. DATA IS TAKEN FROM CONSOLE SWITCHES 8-7 AND MAY BE VARIED DYNAMICALLY. THE DATA IS COMPLEMENTED ON EVERY OTHER PASS TO PROVIDE TRANSITIONS IN THE LOGIC.
- D. GROUP 4: LINE COUNT MODE
THIS GROUP, ENTERED VIA CONSOLE SWITCH EIGHT (8), WILL COUNT THE NUMBER OF CHARACTERS TRANSMITTED IN ONE SECOND AND DIVIDE THIS NUMBER BY 37 IN ORDER TO PROVIDE A PRINTOUT OF THE NUMBER OF LINES PER SECOND. THE TEST WILL ONLY RUN ON A PDP-11/45 WITH A KL11-L LINE CLOCK. AS LONG AS SWITCH 8 IS SET TO A ONE, THE TEST WILL CONTINUE TO RUN WITH A PRINTOUT APPROXIMATELY EVERY SECOND. CONSOLE SWITCH 13 (CONTINUOUS OR SINGLE PASS) WILL HAVE FULL USE DURING THIS TEST; HOWEVER, NO OTHER SWITCHES ARE USED.

8. LISTING

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38

```
.TITLE LPC11  
.ABS  
  
!LPC11 INTERFACE TEST  
!MAINDEC-11-DZLPH-B-D  
!FEBRUARY 25, 1973  
!DIGITAL EQUIPMENT CORP. MAYNARD MASS.  
!PROGRAMMER R. B. BARNES  
  
!*****  
!CONSOLE SWITCH SETTINGS  
!  
!SW 15: 1=NO ERROR PRINT  
!      0=PRINT ALL ERRORS  
!  
!SW 14: 1=STOP ON ERROR  
!      0=CONTINUE  
!  
!SW 13: 1=CONTINUE MODE  
!      0=1 PASS ONLY  
!  
!SW 12: 1=INHIBIT ITERATIONS  
!      0=NORMAL TEST  
!  
!SW 11: 1=MAINTENANCE MODE  
!      0=NORMAL MODE  
!  
!SW 10: 1=DO ERROR BIT TEST  
!      0=NO ERROR BIT TEST  
!  
!SW 9: 1=USE TESTER INTERRUPT IN MAINTENANCE MODE  
!      0=USE CONTROLLER INTERRUPT  
!  
!SW 8: 1=ENTER LINE COUNT MODE (PDP-11/45 ONLY)  
!      0=DO NOT ENTER LINE COUNT MODE  
!*****
```


40			
41			
42			REGISTER EQUIVS
43			
44		000000	R0=00
45		000001	R1=01
46		000002	R2=02
47		000003	R3=03
48		000004	R4=04
49		000005	R5=05
50		000006	SP=06
51		000007	PC=07
52			
53			TRAP CATCHERS 0-1000
54			
55		000000	.=0
56			.NEPT 200
57			.+2
58			HALT
59			.ENDP
60			
61			POWER FAIL VECTOR
62			
63		000024	.=24
64	000024	005352	PwRFAL
65	000026	000340	340
66			
67			LINE CLOCK VECTOR
68			
69		000100	.=100
70	000100	004224	TINT
71	000102	000340	340
72			
73			

75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95

```
.....  
;PROGRAM START LOCATIONS:  
;  
;200 = INPUT VECTOR AND REGISTER ADDRESSES AT START  
;204 = USE PREVIOUSLY SET ADDRESSES  
;  
;VECTORS AND REGISTER ADDRESSES MUST BE  
;INPUT ON THE INITIAL RUN.  
;ON ALL OTHER RUNS, THE PREVIOUSLY SET  
;ADDRESSES MAY BE USED.  
.....  
      .=200  
89 000200 000167 001002  JMP      PBEG          ;GO TO START AND SET VECTOR / REGISTERS  
      .=204  
92 000204 000167 000770  JMP      PSKIP         ;GO TO START BUT DO NOT SET ADDRESSES
```

```

97
98          001000          .=1000
99
100          ;FLAGS AND COUNTERS
101
102 001000 000000      PVEC: 0          ;VECTOR POINTER SAVE
103 001002 000000      PREG: 0          ;REGISTER POINTER SAVE
104 001004 000000      PTN: 0           ;TEST NUMBER STORAGE
105 001006 000000      PSPS: 0          ;STACK POINTER SAVE LOC
106 001010 000000      PITA: 0          ;INTERRUPT RETURN STORAGE
107 001012 000000      PGOF: 0          ;GO FLAG
108 001014 000000      PTIM1: 0         ;TIMER
109 001016 000000      PTIM2: 0         ;GROSS TIMER
110 001020 000000      PIC: 0           ;ITERATION CNTR
111 001022 000000      PCHAR: 0         ;TEST CHAR TEMP STORAGE
112 001024 000000      TUB: 0           ;TTY OUTPUT BUFFER STORAGE
113 001026 000000      TIB: 0           ;TTY INPUT BUFFER STORAGE
114 001030 000000      ROTA: 0          ;ROTATION FACTOR
115 001032 000000      PINTR: 0         ;INTERRUPT RETURN POINTER
116 001034 000000      PET: 0           ;ERNOH BIT TEMP STORAGE
117
118          ;ADDRESS CONSTANTS
119
120 001036 177560      TKS: 177560      ;TTY STATUS
121 001040 177562      TKB: 177562      ;TTY BUFFER
122 001042 177564      TPS: 177564      ;TTP STATUS
123 001044 177566      TPB: 177566      ;TTP BUFFER
124 001046 177546      LKS: 177546      ;LINE CLOCK REGISTER
125 001050 177570      SWR: 177570      ;CONSOLE SWITCH REGISTER
126 001052 177776      PSW: 177776      ;PROGRAM STATUS WORD
127 001054 000000      PCB: 0           ;PACESETTER STATUS
128 001056 000000      PCB: 0           ;PACESETTER BUFFER
129 001060 000000      PTB: 0           ;TEST STATUS
130 001062 000000      PTB: 0           ;TEST BUFFER

```

```

132
133          001200          ,=1200
134
135          ;*****
136          ;PROGRAM START
137          ;
138          ;THIS IS THE HOUSE KEEPING ROUTINE
139          ;WHICH CLEANS ALL COUNTERS AND FLAGS
140          ;TYPES THE PROGRAM NAME, SETS UP
141          ;REGISTERS AND VECTOR ADDRESSES, ASSUMES
142          ;THAT THE LPC11 INTERFACE IS IN THE PROPER
143          ;MODE, AND CHECKS FOR MAINTENANCE MODE.
144          ;*****
145
146          ;CLEAR COUNTERS AND FLAGS
147
148          PSKIP:  MOV     #1,R2          ;SET SKIP FLAG
149          BR      PSTART              ;GO TO START
150          PBEG:   CLR     R2          ;RESET SKIP FLAG
151          PSTART: MOV     #500,#P      ;SET STACK POINTER
152          MOV     #340,#PSW          ;SET PSW
153          RESET
154          CLR     R0                  ;RESET AT START OR RESTART
155          DEC     R0
156          BNE    ,=-2                ;DELAY FOR RESET
157          MOV     #14,R1             ;R1=NUMBER OF LOCATIONS TO CLEAR
158          MOV     #PET,#R0           ;R0=START ADDR
159          CLR     #R0
160          PCLR:   CLR     -(R0)        ;CLEAR ALL COUNTERS
161          DEC     R1
162          BNE    PCLR                ;CONTINUE UNTIL DONE
163          TST    R2                  ;SEE IF SHOULD GET ADDR
164          BNE    PSMC                ;IF NOT; BR
165
166          ;PRINT PROGRAM NAME
167
168          MOV     #MSG1,R4
169          JSR    PC,TTOUT             ;TYPE PROGRAM NAME
170          MOV     #MSG2,R4
171
172          ;GET NEW VECTOR ADDRESS
173
174          JSR    PC,TTOUT             ;TYPE VECTOR REQUEST
175          MOV     #PVEC,R4            ;R4=VECTOR STORAGE ADDR
176          MOV     #3,R3              ;R3=INPUT CHAR NUMBER
177          MOV     #3,ROTA            ;ROTA=ROTATION FACTOR
178          MOV     #177770,R5         ;R5=MASK
179          JSR    PC,READ              ;GO READ IN VECTOR
180          BIT    #1,PVEC             ;TEST FOR ODD ADDR
181          BNE    PVE                  ;IF ODD; BR
182          CMP    #500,PVEC           ;TEST FOR TOO HIGH
183          BHI    PRG                  ;IF OK; BR
  
```

```

185 001344 012704 005624      PVE:  MOV      @MSG3,R4
186 001350 004767 003430      JSR      PC,TTOUT      ;TYPE ILLEGAL VECIOM
187 001354 000750      BH      PVI      ;TRY AGAIN
188
189      ;GEI NEW REGISTER ADDRESS
190
191 001356 012704 005660      PRG:  MOV      @MSG4,R4
192 001362 004767 003416      JSR      PC,TTOUT      ;TYPE REGISTER ADDR REQUEST
193 001366 012704 001002      PRG1:  MOV      @PREG,R4
194 001372 012703 000006      MOV      @6,R3
195 001376 012767 000003 177424  MOV      @3,ROTA
196 001404 012705 177770      MOV      @177770,R5
197 001410 004767 003466      JSR      PC,HEAD      ;GO READ STARTING REGISTER ADDR
198 001414 032767 000001 177360  BIT      @1,PREG      ;TEST FOR ODD ADDR
199 001322 001010      BNE     PREG      ;IF ODD: BR
200 001424 022767 160000 177350  CMP      @160000,PREG  ;TEST FOR TOO LOW
201 001432 101004      BHI     PREG      ;IF TOO LOW: BR
202 001434 022767 177600 177340  CMP      @177600,PREG  ;TEST FOR TOO HIGH
203 001442 101005      BHI     PSET      ;IF NOT TO HIGH: BR
204 001444 012704 005722      PREG:  MOV      @MSG5,R4
205 001450 004767 003330      JSR      PC,TTOUT      ;TYPE ILLEGAL REGISTER ADDR
206 001454 000744      BR      PRG1      ;TRY AGAIN
207 001456 016700 177320      PSET:  MOV      @PREG,R0      ;SET P. S. STATUS ADDR IN R0
208 001462 012701 001054      MOV      @PCS,R1      ;R1=PACESETTER STATUS ADDR
209      .REPT 4
210      MOV      R0,(R1)+      ;SET REGISTERS
211      ADD      @2,R0
212      .ENDR
213
214      ;REQUEST ON LINE/TEST MODE
215
216 001516 012704 005760      MOV      @MSG6,R4
217 001522 004767 003256      JSR      PC,TTOUT      ;REQUEST ON LINE/TEST MODE
218 001526 012704 001012      MOV      @PGOF,R4
219 001532 012703 000001      MOV      @1,R3
220 001536 012767 000003 177264  MOV      @3,ROTA
221 001544 012705 177770      MOV      @177770,R5      ;SET MASK
222 001550 004767 003326      JSR      PC,READ      ;AWAIT GO COMMAND
223
224      ;MAINTENANCE MODE CHECK
225
226 001554 032777 004000 177266  PSMC:  BIT      @4000,@SWR      ;LOOK FOR MAINTENANCE MODE BIT
227 001562 001402      BEQ     PLCC      ;IF NOT: BR
228 001564 000167 002074      JMP     PSCOP      ;ELSE GO TO MAINTENANCE MODE
229
230      ;LINE COUNT MODE ENTRY CHECK
231
232 001570 032777 000400 177252  PLCC:  BIT      @400,@SWR      ;SEE IF SHOULD ENTER LINE COUNT MODE
233 001576 001402      BEQ     PT1      ;IF NOT: BR
234 001600 000167 002306      JMP     PLC      ;ELSE GO TO LINE COUNT MODE
235
236

```

238
 239
 240
 241
 242
 243
 244
 245
 246
 247
 248
 249
 250
 251
 252
 253
 254
 255
 256
 257
 258
 259
 260
 261
 262
 263
 264
 265
 266
 267
 268
 269
 270
 271
 272
 273
 274
 275
 276
 277
 278
 279
 280
 281
 282
 283
 284
 285
 286
 287
 288
 289
 290
 291

```

;*****
;GROUP 1: LOGIC TESTS
;
  
```

```

;THIS GROUP CONSISTS OF 22(0) TESTS TO
;ASSUME THAT ALL FUNCTIONS OF EACH USABLE
;BIT IN BOTH THE TESTER AND THE CONTROLLEN
;STATUS REGISTERS WILL SET AND RESET FROM
;PROPER CONDITIONS AND IN THE PROPER SEQUENCE.
;THE INTERRUPT SYSTEM OF BOTH THE TESTER
;AND CONTROLLEN IS ALSO TESTED TO ASSURE
;THAT EACH WILL TRAP TO THE PROPER VECTOR
;AND PRIORITY LEVEL.
;*****
  
```

LOGIC TEST 01: CLEAR CONT STATUS

```

PT1: CLR      0PCS      ;CLEAR CONT STATUS
      CLR      0PTS      ;CLEAR TESTER STATUS
      MOV      030460,PTN ;SAVE TEST NO.
      JSP     PC,PRSTV   ;RESET VECTORS
      BIT      077500,0PCS ;LOOK FOR CLEAR
      BEQ     PT1A1      ;IF OK: BR
      MOV      0MSG13,R3  ;SET ERROR MSG ADDR
      JMP     PLE        ;GO TYPE ERROR
  
```

LOGIC TEST 02: CLEAR TESTER STATUS

```

PT1A1: MOV     031060,PTN ;SAVE TEST NO.
        BIT     077577,0PTS ;LOOK FOR CLEAR
        BEQ     PT1A1A     ;IF OK: BR
        MOV     0MSG12,R3  ;SET ERROR MSG ADDR
        JMP     PLE        ;GO TYPE ERROR MSG
  
```

LOGIC TEST 03: CONT READY RESET FROM INIT

```

PT1A1A: MOV    031460,PTN ;SAVE TEST NO.
         BIT    0200,0PCS ;LOOK FOR CONT READY RESET
         BEQ    PT1A1B     ;IF OK: BR
         MOV    0MSG41,R3  ;SET ERROR MSG ADDR
         JMP    PLE        ;GO TO ERROR ROUTINE
  
```

LOGIC TEST 04: TESTER READY RESET FROM INIT

```

PT1A1B: MOV    032060,PTN ;SAVE TEST NO.
         BIT    0200,0PTS ;LOOK FOR TESTER READY RESET
         BEQ    PT1A2      ;IF OK: BR
         MOV    0MSG42,R3  ;SET ERROR MSG ADDR
         JMP    PLE        ;GO TO ERROR ROUTINE
  
```

LOGIC TEST 05: CONT READY RESET FROM CHAR LOAD

```

PT1A2: MOV    032460,PTN ;SAVE TEST NO.
        MOV    00,0PCB    ;CLEAR BUFFER
  
```

```

292 001764 032777 000200 177062 BIT #200,@PCS ;LOOK FOR READY
293 001772 001404 BEQ PT1A3 ;IF OK: BR
294 001774 012703 006337 MOV #MSG14,R3 ;SET ERROR MSG ADDR
295 002000 000167 002312 JMP PLE ;GO TYPE ERROR MSG
296
297 ;LOGIC TEST 06: TESTER READY SET AFTER CHAR LOAD
298
299 002004 012767 033060 176772 PT1A3: MOV #33060,PTN ;SAVE TEST NO.
300 002012 005067 176776 CLR PTIM1 ;SET TIMER
301 002016 005367 176772 PT1A3A: DEC PTIM1 ;DELAY
302 002022 001375 BNE PT1A3A
303 002024 032777 000200 177026 BIT #200,@PTS ;LOOK FOR TESTER READY
304 002032 001004 BNE PT1A4 ;IF OK: BR
305 002034 012703 006411 MOV #MSG15,R3 ;SET ERROR MSG ADDR
306 002040 000167 002252 JMP PLE ;GO TYPE ERROR MSG
307
308 ;LOGIC TEST 07: ASSURE GO BIT RESET
309
310 002044 012767 034060 176732 PT1A4: MOV #34060,PTN ;SAVE TEST NO.
311 002052 005277 177002 INC @PTS ;SET GO
312 002056 012767 000100 176730 MOV #100,PTIM1 ;SET TIME
313 002064 005367 176724 PT1A: DEC PTIM1 ;DELAY
314 002070 001375 BNE PT1A
315 002072 032777 000001 176760 BIT #1,@PTS ;LOOK FOR GO RESET
316 002100 001404 BEQ PT1B1 ;IF OK: BR
317 002102 012703 006460 MOV #MSG16,R3 ;SET ERROR MSG ADDR
318 002106 000167 002204 JMP PLE ;GO TYPE ERROR MSG
319
320 ;LOGIC TEST 10: TESTER READY NOT RESET BY GO
321
322 002112 012767 030061 176664 PT1B1: MOV #30061,PTN ;SAVE TEST NO.
323 002120 032777 000200 176732 BIT #200,@PTS ;LOOK FOR TESTER READY NOT RESET
324 002126 001004 BNE PT1B2 ;IF OK: BR
325 002130 012703 006510 MOV #MSG17,R3 ;SET ERROR MSG ADDR
326 002134 000167 002156 JMP PLE ;GO TYPE ERROR MSG
327
328 ;LOGIC TEST 11: CONT READY SET FROM GO
329
330 002140 012767 030461 176636 PT1B2: MOV #30461,PTN ;SAVE TEST NO.
331 002146 005067 176642 CLR PTIM1 ;SET TIMER
332 002152 005367 176636 PT1B2A: DEC PTIM1 ;DELAY
333 002156 001375 BNE PT1B2A
334 002160 032777 000200 176666 BIT #200,@PCS ;LOOK FOR CONT READY SET
335 002166 001004 BNE PT1B3 ;IF OK: BR
336 002170 012703 006546 MOV #MSG20,R3 ;SET ERROR MSG ADDR
337 002174 000167 002116 JMP PLE ;GO TYPE ERROR MSG
338
339 ;LOGIC TEST 12: TESTER READY AT END OF CYCLE
340
341 002200 012767 032601 176576 PT1B3: MOV #32601,PTN ;SAVE TEST NO.
342 002206 012777 000000 176642 MOV #0,@PCH ;LOAD CHAR
343 002214 005067 176574 CLR PTIM1 ;SET TIME
344 002220 005367 176570 PT1B: DEC PTIM1 ;DELAY
345 002224 001375 BNE PT1B

```

```

346 002226 032777 000200 176624      BIT      #200,@PTS      ;LOOK FOR TESTER READY
347 002234 001004                BNE      PT1C        ;IF OK: BR
348 002236 012703 006611          MOV      @MSG21,R3   ;SET ERROR MSG ADDR
349 002242 000167 002050          JMP      PLE         ;GO TYPE ERROR MSG
350
351                                ;LOGIC TEST 13: CONT READY RESET AT END OF CYCLE
352
353 002246 012767 031461 176530 PT1C:  MOV      #31461,PTN   ;SAVE TEST NO.
354 002254 005367 176534                DEC      PTIM1
355 002260 001372                RNE      PT1C        ;DELAY
356 002262 032777 000200 176564      BIT      #200,@PCS   ;LOOK FOR CONT READY
357 002270 001404                BEO      PT1E        ;IF OK: BR
358 002272 012703 006660          MOV      @MSG22,R3   ;SET ERROR MSG ADDR
359 002276 000167 002014          JMP      PLE         ;GO TYPE ERROR MSG
360
361                                ;LOGIC TEST 14: TESTER INTERRUPT
362
363 002302 012767 031461 176474 PT1E:  MOV      #31461,PTN   ;SAVE TEST NO.
364 002310 005077 176544                CLR      @PTS        ;CLEAR TESTER
365 002314 005077 176534                CLR      @PCS        ;CLEAR CONTROLLER
366 002320 016700 176454          MOV      PVEC,R0     ;R0 = STARTING VECTOR
367 002324 022020                CMP      (R0)+,(R0)+ ;POINT TO TESTER VECTOR
368 002326 012720 005236          MOV      @PINT,(R0)+ ;SET VECTOR ADDR
369 002332 012720 000340          MOV      #340,(R0)+ ;SET PSW
370 002336 052777 000100 176514      BIS      #100,@PTS   ;SET TESTER INTERRUPT ENABLE
371 002344 005067 176444                CLR      PTIM1       ;SET TIME
372 002350 012767 000010 176440      MOV      #10,PTIM2   ;SET TIME
373 002356 012767 002422 176446      MOV      @PT1FA,PINTR ;SET RETURN
374 002364 005077 176462                CLR      @PSW        ;SET PSW TO LOWEST PRIORITY
375 002370 012777 000000 176460      MOV      #0,@PCB     ;LOAD CHAR
376 002376 005367 176412                PT1F:  DEC      PTIM1       ;DELAY
377 002402 001375                BNE      PT1F
378 002404 005367 176406                DEC      PTIM2       ;DELAY
379 002410 001372                BNE      PT1F
380 002412 012703 006727          MOV      @MSG24,R3   ;SET ERROR MSG ADDR
381 002416 000167 001674          JMP      PLE         ;GO TYPE ERROR MSG
382
383                                ;LOGIC TEST 15: CONT INTERRUPT
384
385 002422 012767 032461 176354 PT1FA: MOV      #32461,PTN   ;SAVE TEST NO.
386 002430 005077 176424                CLR      @PTS        ;CLEAR TESTER INT ENABLE
387 002434 005067 176354                CLR      PTIM1
388 002440 005367 176350                PT1FB: DEC      PTIM1
389 002444 001375                BNE      PT1FB       ;AWAIT COMPLETE FROM TESTER INTERRUPT
390 002446 005367 176342                PT1FC: DEC      PTIM1
391 002452 001375                BNE      PT1FC
392 002454 004767 002646          JSR      PC,PRSTV    ;RESET VECTORS
393 002460 016700 176314          MOV      PVEC,R0     ;R0 = STARTING VECTOR
394 002464 012720 005236          MOV      @PINT,(R0)+ ;SET VECTOR ADDR
395 002470 012720 000340          MOV      #340,(R0)+ ;SET PSW
396 002474 052777 000100 176352      BIS      #100,@PCS   ;SET CONT INTERRUPT ENABLE
397 002502 012767 000010 176306      MOV      #10,PTIM2   ;SET TIME
398 002510 012767 002552 176314      MOV      @PT1H,PINTR ;SET RETURN
399 002516 005077 176330                CLR      @PSW        ;SET PSW TO LOWEST PRIORITY

```



```

400 002522 005277 176332          INC      @PTS          ;SET GO
401 002526 005367 176262          PT1G:  DEC      PTIM1
402 002532 001375                   BNE      PT1G          ;DELAY
403 002534 005367 176256          DEC      PTIM2
404 002540 001372                   BNE      PT1G          ;DELAY
405 002542 012703 006770          MOV      @MSG25,R3    ;SET ERROR MSG ADDR
406 002546 000167 001544          JMP      PLE          ;GO TYPE ERROR MSG
407
408                                ;LOGIC TEST 16: TESTER INT ENABLE OFF
409
410 002552 012767 033061 176224  PT1H:  MOV      @33061,PTN    ;SAVE TEST NO.
411 002560 005077 176270          CLR      @PCB
412 002564 005077 176270          CLR      @PTS
413 002570 005005                   CLR      R5           ;CLEAR FLAG
414 002572 012767 002704 176210  MOV      @PT1H,PITA    ;SET POINTER
415 002600 012704 002674          MOV      @PT1L,R4     ;SET RETURN ADDRESS
416 002604 005003                   CLR      R3
417 002606 016700 176166          MOV      PVEC,R0      ;R0=STARTING VECTOR ADDRESS
418 002612 022020                   CMP      (R0)+,(R0)+  ;POINT TO TESTER VECTOR
419 002614 012720 005224          PT1HA: MOV      @PINT6,(R0)+ ;SET VECTOR TO SERVICE ROUTINE
420 002620 012720 000340          MOV      @340,(R0)+  ;SET PSW TO PRIORITY 7
421 002624 010467 176202          PT1HB: MOV      R4,PINTR  ;SET INTERRUPT RETURN
422 002630 010377 176216          MOV      R3,@PSW     ;SET PSW
423 002634 005067 176154          CLR      PTIM1       ;CLEAR TIMER
424 002640 012777 000000 176210  MOV      @0,@PCB      ;LOAD CHAR
425 002646 005367 176142          PT1J:  DEC      PTIM1
426 002652 001375                   BNE      PT1J          ;DELAY
427 002654 005367 176134          PT1K:  DEC      PTIM1
428 002660 001375                   BNE      PT1K          ;DELAY
429 002662 012777 000340 176162  MOV      @340,@PSW    ;RESET PSW TO PRIORITY 7
430 002670 000177 176114          JMP      @PITA        ;CONTINUE TO NEXT TEST
431 002674 012703 007246          PT1L:  MOV      @MSG34,R3 ;TYPE ERROR
432 002700 000167 001412          JMP      PLE
433
434                                ;LOGIC TEST 17: TESTER INT AT HIGHER LEVEL
435
436 002704 012767 033461 176072  PT1M:  MOV      @33461,PTN    ;SAVE TEST NO.
437 002712 005077 176144          CLR      @PTS
438 002716 005077 176136          CLR      @PCB
439 002722 005077 176126          CLR      @PTS
440 002726 005705                   TST     R5           ;SEE IF TESTER COMPLETE
441 002730 001029                   BNE     PT1P          ;IF SO:R
442 002732 005205                   INC     R5           ;SET FLAG
443 002734 005367 176054          PT1M1: DEC      PTIM1
444 002740 001375                   BNE     PT1M1        ;DELAY
445 002742 005077 176110          CLR      @PCB
446 002746 052777 000100 176104  BIS     @100,@PTS     ;SET TESTER INTERRUPT ENABLE
447 002754 012777 000200 176070  MOV     @200,@PSW     ;PSW SETTING = PRIORITY 4
448 002762 012767 002774 176042  MOV     @PT1H,PINTR   ;LOAD RETURN ADDRESS
449 002770 000167 176552          JMP     PT1J          ;DELAY FOR INTERRUPT
450 002774 012703 007323          PT1N:  MOV     @MSG35,R3
451 003000 000167 001312          JMP     PLE          ;GO TO ERROR ROUTINE
452

```

```

454
455
456
457 003004 012767 030062 175772 PT1P: MOV      #30062,PTN      ;SAVE TEST NO.
458 003012 005077 176044          CLR      @PTB
459 003016 005077 176036          CLR      @PTS
460 003022 005077 176026          CLR      @PCS
461 003026 005005          CLR      R5          ;CLEAR FLAG
462 003030 016700 175744          MOV      PVEC,R0     ;R0=VECTOR ADDRESS
463 003034 005003          CLR      R3          ;R3=PSW
464 003036 012704 003104          MOV      @PT1R,R4    ;R4=INTERUPT RETURN ADDR
465 003042 012767 003114 175740          MOV      @PT1S,P1TA  ;SET CONTINUE ADDR
466 003050 012720 005224          MOV      @PINTS,(R0)+ ;SET VECTOR ADDR
467 003054 012720 000340          MOV      #340,(R0)+ ;SET PSW
468 003060 010467 175746          MOV      R4,PINTR    ;SET RETURN ADDR
469 003064 010377 175762          MOV      R3,@PSW     ;SET PSW
470 003070 005067 175720          CLR      PTIM1       ;SET TIMER
471 003074 005277 175760          INC      @PTS        ;SET GO
472 003100 000167 177542          JMP      PT1J        ;TEST CONT INTERRUPT
473 003104 012703 007376          PT1R: MOV      @MSG36,R3
474 003110 000167 001202          JMP      PLE         ;GO TO ERROR ROUTINE
475
476
477
478 003114 012767 030462 175662 PT1S: MOV      #30462,PTN      ;SAVE TEST NO.
479 003122 005077 175734          CLR      @PTB
480 003126 005077 175726          CLR      @PTS
481 003132 005077 175716          CLR      @PCS
482 003136 005705          TST      R5          ;SEE IF DONE BOTH TESTS
483 003140 001016          BNE      PEBT        ;IF SO:BR
484 003142 005205          INC      R5          ;SET FLAG
485 003144 052777 000100 175702          BIS      #100,@PCS   ;SET CONTROLLER INTERRUPT ENABLE
486 003152 012703 000200          MOV      #200,R3    ;SET PSW
487 003156 012704 003166          MOV      @PT1T,R4    ;SET INTERRUPT RETURN ADDR
488 003162 000167 177672          JMP      PT1Q        ;TEST CONTROLLER
489 003166 012703 007457          PT1T: MOV      @MSG37,R3
490 003172 000167 001120          JMP      PLE         ;GO TO ERROR ROUTINE
491
492
493
494 003176 012767 031062 175600 PEBT: MOV      #31062,PTN      ;SAVE TEST NUMBER
495 003204 004767 002116          JSR      PC,PRSTV    ;RESET VECTORS.
496 003210 032777 002000 175632          BIT      #2000,@BWR  ;SEE IF ERROR BIT TEST
497 003216 001002          BNE      PEBT0      ;IF SO: BR
498 003220 000167 000150          JMP      PDAT        ;ELSE GO TO DATA TEST
499 003224 005067 175604          PEBT0: CLR      PET      ;TEST FOR RESET
500 003230 012704 007720          MOV      @MSG44,R4   ;REQUEST ON LINE - NORMAL MODE
501 003234 004767 000052          JSR      PC,PEBT1
502 003240 012767 100000 175566          MOV      #100000,PET ;TEST FOR SET
503 003246 012704 010061          MOV      @MSG45,R4   ;REQUEST OFF LINE - NORMAL MODE
504 003252 004767 000034          JSR      PC,PEBT1
505 003256 012704 006131          MOV      @MSG11,R4   ;REQUEST OFF LINE - TEST MODE
506 003262 004767 000024          JSR      PC,PEBT1
507 003266 012704 005760          MOV      @MSG6,R4    ;REQUEST ON LINE - TEST MODE

```

506	003272	004767	000014		JSR	PC,PEBT1	
509	003276	012704	010174		MOV	8MSG46,R4	
510	003302	004767	001476		JSR	PC,ITOUT	ITYPE ERROR BIT OK
511	003306	000167	000062		JMP	PDAT	IGO TO DATA TEST
512	003312	004767	001466	PEBT1:	JSR	PC,TTOUT	ITYPE MSG
513	003316	016704	175470		MOV	PGOF,R4	
514	003322	012703	000001		MOV	81,R3	
515	003326	012767	000003	175474	MOV	83,HOTA	
516	003334	012705	177770		MOV	8177770,R5	
517	003340	004767	001536		JSR	PC,READ	IWAIT GO REPLY
518	003344	017700	175504	PEBT2:	MOV	8PCS,R0	ISAVE CONTROLLER STATUS
519	003350	042700	077777		BIC	877777,R0	IMASK ERROR BIT
520	003354	020067	175454		CMP	R0,PEI	ISEE IF ERROR BIT IN PROPER STATE
521	003360	001404			BEQ	PEBT3	IIF SOI BR
522	003362	012703	007035		MOV	8MSG26,R3	
523	003366	000167	000724		JMP	PLE	IGO TYPE ERROR MSG
524	003372	000207		PEBT3:	RTS	PC	
525							

```

527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550 003374 012701 005466          PDAT1: MOV      #PPATS,R1          ;R1=START OF DATA TABLE
551 003400 012767 001000 175412 PDAT0: MOV      #1000,PIC
552 003406 005077 175442          CLR      #PCB
553 003412 005077 175442          CLR      #PTS
554 003416 005077 175440          CLR      #PTB
555 003422 005067 175374          CLR      PCHAR
556 003426 022701 005510          CMP      #PPATE,R1          ;LOOK FOR END OF DATA TABLE
557 003432 001437                    BEQ      PDAT4              ;IF SO:BR
558 003434 112167 175362          MOVSB   (R1)+,PCHAR        ;SET CHAR
559 003440 116777 175356 175410 PDAT1: MOVSB   PCHAR,#PCB      ;SET BUFFER
560 003446 005067 175342          CLR      PTIM1            ;SET TIMER
561 003452 032777 000200 175400 PDAT2: BIT      #200,#PTS      ;AWAIT TESTER READY
562 003460 001005                    BNE     PDAT2A             ;IF HAVE: BR
563 003462 005367 175326          DEC     PTIM1             ;DELAY
564 003466 001371                    BNE     PDAT2              ;TYPE ERROR
565 003470 000167 000100          JMP     PDAT7A            ;R0=INPUT CHAR
566 003474 117700 175362          PDAT2A: MOVSB   #PTB,#0      ;COMPARE INPUT/OUTPUT
567 003500 120067 175316          CMPB   R0,PCHAR          ;IF OK: BR
568 003504 001402                    BEQ     PDAT3              ;GO TYPE ERROR
569 003506 004767 000672          JSR    PC,PDE             ;LOOK FOR INHIBIT ITERATIONS
570 003512 032777 010000 175330 PDAT3: BIT      #10000,#S#R   ;IF SO: BR
571 003520 001327                    BNE     PDAT0              ;DECREMENT ITERATION COUNTER
572 003522 005367 175272          DEC     PIC               ;IF NOT DONE: BR
573 003526 001344                    BNE     PDAT1              ;ELSE GET NEXT PATTERN
574 003530 000723                    BR      PDAT0
575
576
577
578 003532 012767 000006 175260 PDAT4: MOV      #6,PIC        ;SET ITERATION CNTR
579 003540 005067 175256          PDAT5: CLR      PCHAR
580 003544 016777 175252 175304 PDAT6: MOV      PCHAR,#PCB    ;SET BUFFER

```

581	003552	005067	175236		CLR	PTIM1	%CLEAR TIMEN
582	003556	032777	000200	175274	PDAT7:	#200,%PTS	%LOOK FOR READY BIT SET
583	003564	001011			RNE	PDAT7B	%IF OK: BR
584	003566	005367	175222		DEC	PTIM1	
585	003572	001371			BNE	PDAT7	%AWAIT READY
586	003574	012704	007205		PDAT7A:	MSG33,R4	
587	003600	004767	001200		JSR	PC,%TOUT	%TYPE NO READY IN DATA TEST
588	003604	000167	000746		JMP	PEND	%GO TO END ROUTINE
589	003610	117700	175246		PDAT7B:	%PTB,R0	%R0=INPUT CHAR
590	003614	120067	175202		MOVB	R0,PCHAR	%TEST CHAR
591	003620	001402			CMQB	PDAT10	%IF OK: BR
592	003622	004767	000556		BEQ	PC,PDE	%GO TYPE ERROR
593	003626	005267	175170		JSR	PCHAR	%SET NEW CHAR
594	003632	032767	000400	175162	PDAT10:	#400,PCHAR	%SEE IF DONE ALL
595	003640	001741			BIT	PDAT6	%IF NOT: BR
596	003642	032777	010000	175200	BEQ	#10000,%SNR	%LOOK FOR INHIBIT ITERATIONS
597	003650	001003			BIT	PDAT11	%IF SO: BR
598	003652	005367	175142		RNE	PIC	%LOOK FOR COMPLETE ITERATIONS
599	003656	001330			DEC	PDAT5	%IF NOT: BR
600	003660	000167	000672		BNE	PEND	%GO TO END ROUTINE
601					PDAT11:	JMP	

```

603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625 003664 017702 175160      PSCOP1: MOV      08WR,R2      ;R2 = DATA FROM SWITCHES
626 003670 010203              MOV      R2,R3
627 003672 127702 175152      PSCOP01: CMPB     08WR,R2      ;SEE IF NEW DATA
628 003676 001372              BNE     PSCOP              ;IF SO: BR
629 003700 012777 000340 175144  MOV      0340,0PSW        ;RESET PSW TO PRIORTIY 7
630 003706 005077 175146      CLR     0PTS              ;CLEAR STATUS REGISTERS
631 003712 005077 175136      CLR     0PCS
632 003716 004767 001404      JSR     PC,PRSTV          ;RESET VECTORS
633 003722 032777 001000 175120  BIT     01000,08WR        ;SEE IF TESTER INTERRUPT
634 003730 001012              BNE     PSCOP1            ;IF SO: BR
635 003732 016700 175042      MOV     PVEC,R0          ;R0 = STARTING VECTOR
636 003736 012720 005224      MOV     0PINTS,(R0)+      ;SET VECTOR FOR MAINTENANCE MODE TRAP
637 003742 012720 000340      MOV     0340,(R0)+
638 003746 052777 000100 175100  BIS     0100,0PCS        ;SET CONTROLLER INTERRUPT ENABLE
639 003754 000412              BR      PSCOP2
640 003756 052777 000100 175074  PSCOP11: BIS     0100,0PTS      ;SET TESTER INTERRUPT ENABLE
641 003764 016700 175010      MOV     PVEC,R0          ;R0 = STARTING VECTOR
642 003770 022020              CMP     (R0)+,(R0)+      ;POINT TO TESTER VECTOR
643 003772 012720 005224      MOV     0PINTS,(R0)+      ;SET VECTOR ADDR TO MAINTENANCE MODE TRAP
644 003776 012720 000340      MOV     0340,(R0)+
645 004002 012767 004052 175022  PSCOP21: MOV     0PSCOP4,PINTR    ;SET RETURN ADDR
646 004010 012767 000010 175000  MOV     010,PTIM2        ;SET DELAY
647 004016 005067 174772      CLR     PTIM1            ;SET DELAY
648 004022 110377 175030      MOVB   R3,0PCB          ;LOAD CHAR
649 004026 005277 175026      INC     0PTS              ;SET GO BIT
650 004032 005077 175014      CLR     0PSW            ;SET TO PRIORITY 0
651 004036 005367 174752      PSCOP31: DEC     PTIM1
652 004042 001375              BNE     PSCOP3
653 004044 005367 174746      DEC     PTIM2
654 004050 001372              BNE     PSCOP3
655 004052 005103      PSCOP41: COM     R3
656 004054 032777 001000 174766  BIT     01000,08WR        ;SEE IF ON TESTER INTERRUPT

```

657	004062	001703		BEG	PSCOPO		IF NOT: BR
658	004064	005067	174724	CLR	PTIM1		
659	004070	032777	000200	174762	PSCOPS:	BIT	200,PTS
660	004076	001275		BNE	PSCOPO		LOOK FOR TESTER READY
661	004100	005367	174710	DEC	PTIM1		IF 801 BH
662	004104	001371		BNE	PSCOPS		DELAY FOR COMPLETE FROM TESTER
663	004106	000167	177560	JMP	PSCOPO		CONTINUE

```

665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682 004112 005077 174742          PLC:  CLR      OPTS          ;CLEAR TESTER STATUS
683 004116 005077 174732          CLR      OPC5          ;CLEAR CONTROLLER STATUS
684 004122 005077 174720          CLR      OLKS          ;CLEAR LINE CLOCK
685 004126 005003                  CLR      R3            ;CLEAR CHARACTER COUNTER
686 004130 005277 174724          INC      OPTS          ;SET GO
687 004134 016701 000152          MOV      LINE,R1       ;SET DIVISON
688 004140 052777 000100 174712  BIS      @100,OPTS     ;SET TESTER INTERRUPT ENABLE
689 004146 016700 174626          MOV      PVEC,R0
690 004152 022020                  CMP      (R0)+,(R0)+   ;POINT TO TESTER VECTOR
691 004154 012720 004232          MOV      @DINT,(R0)+  ;SET VECTOR
692 004160 012710 000340          MOV      @340,(R0)    ;SET TRAP PSW
693 004164 016700 000124          MOV      TIME,R0      ;SET 1 SECOND TIMER
694 004170 005077 174656          CLR      OPSW          ;CLEAR PSW
695 004174 032777 000200 174644  PLC1:  BIT      @200,OLKS
696 004202 001774                  BEQ      PLC1          ;AWAIT CLOCK CYCLE START
697 004204 052777 000100 174634  BIS      @100,OLKS     ;SET CLOCK INTERRUPT ENABLE
698 004212 010077 174640          PLC2:  MOV      R0,OPCB  ;TRANSMIT CHARACTER
699 004216 000001                  WAIT
700 004220 000167 177766          JMP      PLC2          ;AWAIT INTERRUPT
701
702
703
704 004224 005300                  ;CLOCK TIMER INTERRUPT HANDLER
705 004226 001405          TINT:  DEC      R0            ;SEE IF DONE 1 SECOND
706 004230 000002          BEQ      PLCP          ;IF SO: BR
707
708
709
710 004232 005203                  ;DATA INTERRUPT HANDLER
711 004234 005277 174620          DINT:  INC      R3            ;BUMP CHARACTER COUNTER
712 004240 000002          INC      OPTS          ;SET GO
713
714
715
716 004242 005077 174600          RTI
717 004246 005077 174606          ;LINE COUNT PRINT ROUTINE
718 004252 005077 174576          PLCP:  CLR      OLKS          ;CLEAR CLOCK
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```


719	004256	071103				DIV	R3,R1		!DIVIDE CHARACTER AMOUNT BY LINE FACTOR
720	004260	012704	007673			MOV	@MSG43,R4		
721	004264	004767	000914			JSR	PC,TTOUT		!PRINT HEADER
722	004270	004767	000310			JSR	PC,DECPNT		!PRINT LINE AMOUNT IN DECIMAL
723	004274	032777	020000	174546		BIT	@20000,@SWR		!SEE IF CONTINUOUS MODE
724	004302	001001				BNE	PLCG		!IF SO: BR
725	004304	000000				HALT			
726	004306	000167	175256		PLCG:	JMP	PLCC		!CONTINUE
727									
728	004312	000045			LINE:		45		!DIVISION FOR 37 CHARACTER PER LINE
729	004314	000072			TIME:		72		!TIMER FOR 1 SECOND AT 60 CYCLES
730									
731									!*****
732									!LOGIC TEST ERROR ROUTINE
733									!
734									!THIS ROUTINE IS USED TO DISPLAY THE
735									!TEST NUMBER AND ERROR CONDITION FOUND
736									!DURING GROUP 1.
737									!ERROR PRINTING IS CONDITIONED BY
738									!CONSOLE SWITCH 15, CONSOLE SWITCH 14
739									!CONDITIONS STOP OR CONTINUE ON ERROR.
740									!DUE TO TIMING AND SEQUENCE CONSIDERATIONS
741									!ANY LOGIC ERROR WILL RESULT IN A RESTART
742									!OF THE LOGIC TEST.
743									!*****
744									
745	004316	032777	100000	174524	PLE:	BIT	@100000,@SWR		!SEE IF NO PRINT
746	004324	001012				BNE	PCONTL		!IF SO: BR
747	004326	012704	005512			MOV	@MSG0,R4		!R4 = TEST NUMBER MSG ADDR
748	004332	016764	174446	000020		MOV	PTN,+20(R4)		!SET TEST NUMBER
749	004340	004767	000440			JSR	PC,TTOUT		!GO TYPE TEST NUMBER
750	004344	010304				MOV	R3,R4		!R4 = ERROR MAG ADDR
751	004346	004767	000432			JSR	PC,TTOUT		!PRINT ERROR
752	004352	032777	040000	174470	PCONTL:	BIT	@40000,@SWR		!SEE IF HALT ON ERROR
753	004360	001401				BEO	PGOL		!IF NOT: BR
754	004362	000000				HALT			
755	004364	032777	002000	174456	PGOL:	BIT	@2000,@SWR		!SEE IF IN ERROR BIT TEST
756	004372	001774				BEO	PGOL		!IF NOT: BR
757	004374	000167	176744			JMP	PEBT2		!RETRY ERROR BIT TEST
758	004400	000167	174574		PGO:	JMP	PSKIP		!RESTART
759									

```

761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777 004404 032777 100000 174436 PDE1 BIT      0100000,0SWR  ;SEE IF NO PRINT
778 004412 001053          BNE      PCONTD      ;IF NO PRINT: BR
779 004414 012704 007123      MOV      0MSG27,R4    ;SET ERROR MSG ADDR
780 004420 004767 000360      JSR      PC,TTOUT
781 004424 012704 007140      MOV      0MSG30,R4
782 004430 004767 000350      JSR      PC,TTOUT      ;TYPE EXPECTED DATA HEADER
783 004434 005003          CLR      R3
784 004436 016704 174360      MOV      PCHAR,R4     ;R4 = EXPT CHAR
785 004442 012705 000010      PDE1:  MOV      010,R5     ;R5 = NUMBER OF BITS TO TYPE
786 004446 105777 174370      PDE2:  TSTB      0TPB
787 004452 100375          BPL      PDE2         ;AWAIT TTY READY
788 004454 132704 000200      BITB      0200,R4     ;SEE IF ONE OR ZERO
789 004460 001404          BEQ      PDE3         ;IF ZERO: BR
790 004462 012777 000061 174354      MOV      0061,0TPB   ;TYPE A ONE
791 004470 000403          BR       PDE4
792 004472 012777 000060 174344      PDE3:  MOV      0060,0TPB ;TYPE A ZERO
793 004500 006104      PDE4:  ROL      R4       ;GET NEXT BIT
794 004502 005305          DEC      R5          ;SEE IF DONE ALL
795 004504 001360          BNE      PDE2         ;IF NOT: BR
796 004506 005703          TST      R3          ;LOOK FOR RECIEVED DONE FLAG
797 004510 001010          BNE      PDE5         ;IF NOT: BR
798 004512 012704 007161      MOV      0MSG31,R4
799 004516 004767 000262      JSR      PC,TTOUT
800 004522 005203          INC      R3
801 004524 010004          MOV      R0,R4
802 004526 000167 177710      JMP      PDE1         ;SET RECIEVED CHAR IN R4
803 004532 012704 007202      PDE5:  MOV      0MSG32,R4 ;DO RECIEVED
804 004536 004767 000242      JSR      PC,TTOUT
805 004542 032777 040000 174300      PCONTD: BIT      040000,0SWR ;TYPE CR/LF
806 004550 001401          BEQ      PGOD         ;SEE IF HALT ON ERROR
807 004552 000000          HALT
808 004554 000207      PGOD:  RTS      PC      ;IF NOT: BR
809

```

```
011  
012  
013  
014  
015  
016  
017  
018  
019  
020  
021  
022 004556 032777 020000 174264 PEND: BIT 020000,0SWR ;LOOK FOR CONTUOUS MODE  
023 004564 001005 BNE PLOOP ;IF 00: 0K  
024 004566 012704 006067 MOV 0MSG7,R4  
025 004572 004767 000206 JSR PC,ITUUT ;PRINT END OF TEST  
026 004576 000000 HALT  
027 004600 000167 174374 PLOOP: JMP P&KIP ;LOOP  
028
```

```
;*****  
;PROGRAM END ROUTINE  
;  
;THIS ROUTINE IS USED TO PRINT OUT  
;"END OF TEST" AND THEN HALT AT  
;THE END OF A SINGLE PASS OR TO  
;RESTART WITH NO PRINT OUT DEPENDING  
;ON CONSOLE SWITCH 13.  
;*****
```

```

b30
b31
b32
b33
b34
b35
b36
b37 004604 012767 177773 000152  DECPRT: MOV      #=5,DIGCNT
b38 004612 012767 004772 000150      MOV      @DECPNT+2,DECPNT
b39 004620 005067 000142      CLR      ZERO
b40 004624 012767 177777 000130  TYPT1:  MOV      #=1,DIGIT
b41 004632 005267 000124      TYPT2:  INC      DIGIT
b42 004636 167703 000126      SUB      @DECPNT,R3
b43 004642 100373      BPL     TYPT2
b44 004644 067703 000120      ADD      @DECPNT,R3
b45 004650 004767 000020      JSR     PC,DECOUT
b46 004654 005267 000104      INC     DIGCNT
b47 004660 001001      BNE     TYPT3
b48 004662 000207      RIS     PC
b49 004664 062767 000002 000076  TYPT3:  ADD      #2,DECPNT
b50 004672 000754      BR      TYPT1
b51 004674 005767 000062      DECOUT: TST     DIGIT
b52 004700 001010      BNE     DEC1
b53 004702 022767 177777 000054      CMP     #=1,DIGCNT
b54 004710 001404      BEQ     DEC1
b55 004712 016767 000050 000042      MOV     ZERO,DIGIT
b56 004720 000406      BR      DEC2
b57 004722 012767 000260 000036  DEC1:  MOV     #260,ZERO
b58 004730 052767 000260 000024      BIS     #260,DIGIT
b59 004736 005767 000020      DEC2:  TST     DIGIT
b60 004742 001406      BEQ     DEC3
b61 004744 105777 174072      TST@   @TP5
b62 004750 100372      BPL     DEC2
b63 004752 016777 000004 174064      MOV     DIGIT,@TPB
b64 004760 000207      DEC3:  RIS     PC
b65
b66 004762 000000      DIGIT:  0
b67 004764 000000      DIGCNT: 0
b68 004766 000000      ZERO:  0
b69 004770 004772      DECPNT: .+2
b70 004772 023420      10000.
b71 004774 001750      1000.
b72 004776 000144      100.
b73 005000 000012      10.
b74 005002 000001      1.
b75
  
```

```

877
878
879
880
881
882
883
884 005004 112467 174014      TTOUT:  MOVB    (R4)+,TOB      ;TOB = OUTPUT CHAN
885 005010 122767 000043 174006      CMPB    043,TOB          ;LOOK FOR TERMINATOR
886 005016 001430                      BEQ     TEX              ;EXIT
887 005020 122767 000045 173776      CMPB    045,TOB          ;LOOK FOR CR/LF
888 005026 001403                      BEQ     TCRLF            ;IF SO: BR
889 005030 004767 000030                      JSR     PC,TOG           ;GO TYPE CHAR
890 005034 000763                      BR      TTOUT            ;CONTINUE
891 005036 112767 000015 173760      TCRLF:  MOVB    015,TOB          ;TOB = CR
892 005044 004767 000014                      JSR     PC,TOG           ;GO TYPE CR
893 005050 112767 000012 173746      MOVB    012,TOB          ;TOB = LF
894 005056 004767 000002                      JSR     PC,TOG           ;GO TYPE LF
895 005062 000750                      BR      TTOUT            ;CONTINUE
896 005064 105777 173752      TOG:    TSTB    0TPB          ;LOOK FOR DONE
897 005070 100375                      BPL     TOG              ;AWAIT FLAG
898 005072 116777 173726 173744      MOVB    TOB,0TPB         ;ECHO CHAN
899 005100 000207      TEX:    RTS     PC              ;EXIT
900
901
902
903
904
905
906
907
908
909 005102 005014                      READ:   CLR     0R4          ;CLEAR FOR INPUT
910 005104 004767 000042      READ1:  JSR     PC,TTIN         ;GO READ KEYBOARD AND ECHO
911 005110 022767 000215 173710      CMP     0215,TIB         ;LOOK FOR CR
912 005116 001414                      BEQ     REX              ;IF SO: BR
913 005120 016701 173704      MOV     ROTA,R1          ;SET ROTATION FACTOR
914 005124 000241      ROT:    CLC              ;
915 005126 006114      ROL     0R4              ;ROTATE INPUT
916 005130 005301      DEC     R1              ;DECREMENT POSITION FACTOR
917 005132 001374      BNE     ROT              ;DO ALL ROTATIONS
918 005134 040567 173666      BIC     R5,TIB           ;STRIP ASCII
919 005140 056714 173662      BIS     TIB,0R4          ;SET NEW DIGIT
920 005144 005303      DEC     R3              ;DECREMENT CHAR CNTR
921 005146 001356      BNE     READ1           ;CONTINUE
922 005150 000207      REX:    RTS     PC              ;EXIT
923
924
925
926 005152 005077 173660      TTIN:   CLR     0TKS         ;
927 005156 005077 173656      CLR     0TKB         ;
928 005162 005067 173640      CLR     TIB           ;
929 005166 005277 173644      INC     0TKS         ;SET READER GO BIT
930 005172 105777 173640      TTIN1:  TSTB    0TKS         ;AWAIT DONE

```

LPC11b.SPC

931	005176	100375				BPL	TTIN1	
932	005200	017767	173634	173620		MOV	@TKB, TIB	!TIB = INPUT CHAR
933	005206	105777	173630		TTIN2:	TSTB	@TPS	!AWAIT PUNCH READY
934	005212	100375				BPL	TTIN2	
935	005214	116777	173606	173622		MOVB	TIB, @TPB	!SET ECHO
936	005222	000207				RIS	PC	!EXIT
937								

```

939
940
941
942
943
944
945
946
947
948 005224 032626
949 005226 004767 000074
950 005232 000177 173574
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965 005236 022626
966 005240 005077 173614
967 005244 005077 173604
968 005250 024040
969 005252 012720 005316
970 005256 012720 000340
971 005262 005077 173564
972 005266 005067 173522
973 005272 005367 173516
974 005276 001375
975 005300 012777 000340 173544
976 005306 004767 000014
977 005312 000177 173514
978 005316 012704 006105
979 005322 000167 176770
980

;*****
;MAINTENANCE MODE INTERRUPT ROUTINE
;
;THIS ROUTINE IS USED TO RESET THE
;STACK POINTER AND TO RESET VECTORS
;TO HALTS THEN RETURN.
;*****

PINTS: BIT      (SP)+,(SP)+      ;RESET STACK POINTER
        JSR      PC,PRSTV      ;RESET VECTORS
        JMP      @PINTR        ;RETURN

;*****
;LOGIC TEST INTERRUPT ROUTINE
;
;THIS ROUTINE IS USED TO RESET THE
;THE STACK POINTER AND TO ASSURE THAT
;THE INTERRUPT DOES INDEED CLEAR.
;IF THE INTERRUPT SHOULD NOT CLEAR
;IT WILL BE FLAGGED AS AN ERROR,
;IF THE INTERRUPT IS CLEARED, THEN
;THE VECTORS WILL BE RESET AND A
;RETURN EXECUTED,
;*****

PINTI: CMP      (SP)+,(SP)+      ;RESET STACK POINTER
        CLR      @PTS          ;RESET INTERRUPT ENABLE IN TESTER
        CLR      @PCS          ;RESET CONT INTERRUPT ENABLE
        CMP      -(R0),-(R0)    ;POINT TO PROPER VECTOR
        MOV      @PINTE,(R0)+   ;SET TO ERROR VECTOR
        MOV      @340,(R0)+
        CLR      @PSW          ;SET PSW TO LOWEST PRIORITY
        CLR      PTIM1         ;SET TIME
PINT1: DEC      PTIM1
        BNE      PINT1         ;AWAIT ANY ERROR INTERRUPTS
        MOV      @340,@PSW      ;RESET PSW
        JSR      PC,PRSTV      ;RESET VECTORS
        JMP      @PINTR        ;RETURN
PINTE: MOV      @MSG10,R4
        JMP      PLE           ;GO TO ERROR ROUTINE
  
```

```

982
983
984
985
986
987
988
989
990
991 005326 016700 173446 PRSTVI MOV PVEC,R0 ;RO = CONT VECTOR
992 005332 010001 MOV R0,R1
993 005334 005721 TST (R1)+ ;POINT TO CONT HALT LOC
994 005336 010120 MOV R1,(R0)+ ;SET CONT VECTOR ADDR
995 005340 005020 CLR (R0)+ ;SET CONT HALT
996 005342 022121 CMP (R1)+,(R1)+ ;POINT TO TESTER HALT
997 005344 010120 MOV R1,(R0)+ ;SET TESTER VECTOR ADDR
998 005346 005020 CLR (R0)+ ;SET TESTER HALT
999 005350 000207 RTS PC ;RETURN

1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010 005352 010046 PWRFALI MOV R0,-(SP)
1011 005354 010146 MOV R1,-(SP)
1012 005356 010246 MOV R2,-(SP)
1013 005360 010346 MOV R3,-(SP)
1014 005362 010446 MOV R4,-(SP)
1015 005364 010546 MOV R5,-(SP)
1016 005366 016746 172432 MOV 24,-(SP)
1017 005372 010667 173410 MOV SP,PSPS ;SAVE STACK POINTER
1018 005376 012767 005406 172420 MOV @PWRUP,24
1019 005404 000000 HALT

1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030 005406 012777 000340 173436 PWRUPI MOV @340,@PSW
1031 005414 016706 173366 MOV PSPS,SP
1032 005420 012667 172400 MOV (SP)+,24
1033 005424 012605 MOV (SP)+,R5
1034 005426 012604 MOV (SP)+,R4
1035 005430 012603 MOV (SP)+,R3

```


1036	005432	012602		MOV	(SP)+,R2	
1037	005434	012601		MOV	(SP)+,P1	
1038	005436	012600		MOV	(SP)+,R0	
1039	005440	005005		CLR	R5	
1040	005442	005305		DEC	R5	
1041	005444	001376		BNE	.-2	
1042	005446	012704	007536	MOV	BMSG40,R4	ITYPE POWER FAIL MESSAGE
1043	005452	004767	177326	JSR	PC,TTOUT	
1044	005456	005305		DEC	R5	
1045	005460	001376		BNE	.-2	
1046	005462	000167	173520	JMP	PBEG	
1047						

```

1049
1050
1051
1052
1053
1054
1055 005466      000      377      376  PPATIS: .EVEN
      005471      375      373      367      .BYTE  0,377,376,375,373,367,357,337,277,177
      005474      357      337      277
      005477      177
1056 005500      001      002      004      .BYTE  1,2,4,10,20,40,100,200
      005503      010      020      040
      005506      100      200
1057 005510      377          PPATE:  .BYTE  377
1058          005512          .EVEN
1059
1060
1061
1062
1063
1064
1065 005512  046045  043517  041511  MSG0:  .ASCII  /%LOGIC TEST NO. 00%/
      005520  052040  051505  020124
      005526  047516  020056  030060
      005534      043
1066
1067 005535      045  050114  030503  MSG1:  .ASCII  /%LPC11 INTERFACE TEST%/
      005542  020061  047111  042524
      005550  043122  041501  020105
      005556  042524  052123  021445
1068
1069 005564  042445  052116  051105  MSG2:  .ASCII  /%ENTER STARTING VECTOR ADDRESS%/
      005572  051440  040524  052122
      005600  047111  020107  042526
      005606  052103  051117  040440
      005614  042104  042522  051523
      005622  021445
1070
1071 005624  044445  046114  043505  MSG3:  .ASCII  /%ILLEGAL VECTOR: TRY AGAIN%/
      005632  046101  053040  041505
      005640  047524  035122  052040
      005646  054522  040440  040507
      005654  047111  021445
1072
1073 005660  042445  052116  051105  MSG4:  .ASCII  /%ENTER STARTING REGISTER ADDRESS%/
      005666  051440  040524  052122
      005674  047111  020107  042522
      005702  044507  052123  051105
      005710  040440  042104  042522
      005716  051523  021445
1074
1075 005722  044445  046114  043505  MSG5:  .ASCII  /%ILLEGAL REGISTER: TRY AGAIN%/
      005730  046101  051040  043505
      005736  051511  042524  035122
  
```

LPC11+.SRC

	005744	052040	054522	040440		
	005752	040507	047111	021445		
1076						
1077	005760	051445	052105	044440	MSG6:	.ASCII /%SET INTERFACE TO 'ON LINE' AND 'TEST MODE'%/
	005766	052116	051105	040506		
	005774	042503	052040	020117		
	006002	047447	020116	044514		
	006010	042516	020047	047101		
	006016	020104	052047	051505		
	006024	020124	047515	042504		
	006032	022447				
1076	006034	054524	042520	023440		.ASCII /TYPE 'G' WHEN READY TO GO%/
	006042	023507	053440	042510		
	006050	020116	042522	042101		
	006056	020131	047524	043440		
	006064	022517	043			
1079						
1080	006067	045	047105	020104	MSG7:	.ASCII /%END OF TEST%/
	006074	043117	052040	051505		
	006102	022524	043			
1081						
1082	006105	045	047111	042524	MSG10:	.ASCII /%INTERRUPT HANG UP%/
	006112	051122	050125	020124		
	006120	040510	043516	052440		
	006126	021520	045			
1083						
1084	006131	045	042523	020124	MSG11:	.ASCII /%SET INTERFACE TO 'OFF LINE' AND 'TEST MODE'%/
	006136	047111	042524	043122		
	006144	041501	020105	047524		
	006152	023440	043117	020106		
	006160	044514	042516	020047		
	006166	047101	020104	052047		
	006174	051505	020124	047515		
	006202	042504	022447			
1085	006206	054524	042520	023440		.ASCII /TYPE 'G' WHEN READY TO GO%/
	006214	023507	053440	042510		
	006222	020116	042522	042101		
	006230	020131	047524	043440		
	006236	022517	043			
1086						
1087	006241	045	040503	047116	MSG12:	.ASCII /%CANNOT CLEAR TESTER STATUS%/
	006246	052117	041440	042514		
	006254	051101	052040	051505		
	006262	042524	020122	052123		
	006270	052101	051525	021445		
1088						
1089	006276	041445	047101	047516	MSG13:	.ASCII /%CANNOT CLEAR CONTROLLER STATUS%/
	006304	020124	046103	040505		
	006312	020122	047503	052116		
	006320	047522	046114	051105		
	006326	051440	040524	052524		
	006334	022523	043			
1090						
1091	006337	045	047503	052116	MSG14:	.ASCII /%CONTROLLER READY NOT RESET BY CHAR LOAD%/

	006344	047522	046114	051105		
	006352	051040	040505	054504		
	006360	047040	052117	051040		
	006366	051505	052105	041040		
	006374	020131	044103	051101		
	006402	046040	040517	022504		
	006410	043				
1092						
1093	006411	045	042524	052123	MSG15:	.ASCII /%TESTER READY NOT SET AFTER CHAR LOAD%/
	006416	051105	051040	040505		
	006424	054504	047040	052117		
	006432	051440	052105	040440		
	006440	052106	051105	041440		
	006446	040510	020122	047514		
	006454	042101	021445			
1094						
1095	006460	043445	020117	044502	MSG16:	.ASCII /%GO BIT WILL NOT RESET%/
	006466	020124	044527	046114		
	006474	047040	052117	051040		
	006502	051505	052105	021445		
1096						
1097	006510	052045	051505	042524	MSG17:	.ASCII /%TESTER READY RESETS FROM GO%/
	006516	020122	042522	042101		
	006524	020131	042522	042523		
	006532	051524	043040	047522		
	006540	020115	047507	021445		
1098						
1099	006546	041445	047117	051124	MSG20:	.ASCII /%CONTROLLER READY NOT SET FROM GO%/
	006554	046117	047514	020122		
	006562	042522	042101	020131		
	006570	047516	020124	042523		
	006576	020124	051106	046517		
	006604	043440	022517	043		
1100						
1101	006611	045	042524	052123	MSG21:	.ASCII /%TESTER READY NOT SET AT END OF CYCLE%/
	006616	051105	051040	040505		
	006624	054504	047040	052117		
	006632	051440	052105	040440		
	006640	020124	047105	020104		
	006646	043117	041440	041531		
	006654	042514	021445			
1102						
1103	006660	041445	047117	051124	MSG22:	.ASCII /%CONTROLLER READY SET AT END OF CYCLE%/
	006666	046117	042514	020122		
	006674	042522	042101	020131		
	006702	042523	020124	052101		
	006710	042440	042116	047440		
	006716	020106	054503	046103		
	006724	022505	043			
1104						
1105	006727	045	047516	044440	MSG24:	.ASCII /%NO INTERRUPT FROM TESTER READY%/
	006734	052116	051105	052522		
	006742	052120	043040	047522		
	006750	020115	042524	052123		

LPC11B.SPC

	006756	051105	051040	040505	
	006764	054504	021445		
1106					
1107	006770	047045	020117	047111	MSG25: .ASCII /%NO INTERRUPT FROM CONTROLLER READY%/
	006776	042524	051122	050125	
	007004	020124	051106	046517	
	007012	041440	047117	051124	
	007020	046117	042514	020122	
	007026	042522	042101	022531	
	007034	043			
1108					
1109	007035	045	051105	047522	MSG26: .ASCII /%ERROR BIT WILL NOT GO TO PROPER STATE%/
	007042	020122	044502	020124	
	007050	044527	046114	047040	
	007056	052117	043440	020117	
	007064	047524	050040	047522	
	007072	042520	020122	052123	
	007100	052101	022505		
1110	007104	047506	020122	044124	.ASCII /%FOR THIS MODE%/
	007112	051511	046440	042117	
	007120	022505	043		
1111					
1112	007123	045	040504	040524	MSG27: .ASCII /%DATA ERROR%/
	007130	042440	051122	051117	
	007136	021445			
1113					
1114	007140	042445	050130	041505	MSG30: .ASCII /%EXPECTED DATA: %/
	007146	042524	020104	040504	
	007154	040524	020072	043	
1115					
1116	007161	045	042522	044503	MSG31: .ASCII /%RECEIVED DATA: %/
	007166	053105	042105	042040	
	007174	052101	035101	021440	
1117					
1118	007202	022445	043		MSG32: .ASCII /%%/
1119					
1120					
1121	007205	045	047516	051040	MSG33: .ASCII /%NO READY FLAG DURING DATA TEST%/
	007212	040505	054504	043040	
	007220	040514	020107	052504	
	007226	044522	043516	042040	
	007234	052101	020101	042524	
	007242	052123	021445		
1122					
1123	007246	052045	051505	042524	MSG34: .ASCII /%TESTER INTERRUPTS WITH NO INTERRUPT ENABLE%/
	007254	020122	047111	042524	
	007262	051122	050125	051524	
	007270	053440	052111	020110	
	007276	047516	044440	052116	
	007304	051105	052522	052120	
	007312	042440	040516	046102	
	007320	022505	043		
1124					
1125	007323	045	042524	052123	MSG35: .ASCII /%TESTER INTERRUPTS AT HIGHER THAN LEVEL 4%/

	007330	051105	044440	052116	
	007336	051105	052522	052120	
	007344	020123	052101	044040	
	007352	043511	042510	020122	
	007360	044124	047101	046040	
	007366	053105	046105	032040	
	007374	021445			
1126					
1127	007376	041445	047117	051124	MSG36: .ASCII /%CONTROLLER INTERRUPTS WITH NO INTERRUPT ENABLE%/
	007404	046117	042514	020122	
	007412	047111	042524	051122	
	007420	050125	051524	053440	
	007426	052111	020110	047516	
	007434	044440	052116	051105	
	007442	052522	052120	042440	
	007450	040516	046102	022505	
	007456	043			
1128					
1129	007457	045	047503	052116	MSG37: .ASCII /%CONTROLLER INTERRUPTS AT HIGHER THAN LEVEL 4%/
	007464	047522	046114	051105	
	007472	044440	052116	051105	
	007500	052522	052120	020123	
	007506	052101	044040	043511	
	007514	042510	020122	044124	
	007522	047101	046040	053105	
	007530	046105	032040	021445	
1130					
1131	007536	051045	051505	040524	MSG40: .ASCII /%RESTART FROM POWER FAILURE%/
	007544	052122	043040	047522	
	007552	020115	047520	042527	
	007560	020122	040506	046111	
	007566	051125	022505	043	
1132					
1133	007573	045	047503	052116	MSG41: .ASCII /%CONT READY NOT RESET BY INIT%/
	007600	051040	040505	054504	
	007606	047040	052117	051040	
	007614	051505	052105	041040	
	007622	020131	047111	052111	
	007630	021445			
1134					
1135	007632	052045	051505	042524	MSG42: .ASCII /%TESTER READY NOT RESET BY INIT%/
	007640	020122	042522	042101	
	007646	020131	047516	020124	
	007654	042522	042523	020124	
	007662	054502	044440	044516	
	007670	022524	043		
1136					
1137	007673	045	044514	042516	MSG43: .ASCII /%LINES PER SECOND: %%/
	007700	020123	042520	020122	
	007706	042523	047503	042116	
	007714	020072	021445		
1138					
1139	007720	042445	051122	051117	MSG44: .ASCII /%ERROR STATUS BIT TEST %/
	007726	051440	040524	052524	

	007734	020123	044502	020124	
	007742	042524	052123	022440	
1140	007750	042523	020124	047111	.ASCII /SET INTERFACE TO 'ON LINE' AND 'NORMAL MODE'&/
	007756	042524	043122	041501	
	007764	020105	047524	023440	
	007772	047117	046040	047111	
	010000	023505	040440	042116	
	010006	023440	047516	046522	
	010014	046101	046440	042117	
	010022	023505	045		
1141	010025	124	050131	020105	.ASCII /TYPE 'G' WHEN READY TO GO &/
	010032	043447	020047	044127	
	010040	047105	051040	040505	
	010046	054504	052040	020117	
	010054	047507	022440	043	
1142					
1143	010061	045	042523	020124	MSG45: .ASCII /&SET INTERFACE TO 'OFF LINE' AND 'NORMAL MODE'&/
	010066	047111	042524	043122	
	010074	041501	020105	047524	
	010102	023440	043117	020106	
	010110	044514	042516	020047	
	010116	047101	020104	047047	
	010124	051117	040515	020114	
	010132	047515	042504	022447	
1144	010140	054524	042520	023440	.ASCII /TYPE 'G' WHEN READY TO GO &/
	010146	023507	053440	042510	
	010154	020116	042522	042101	
	010162	020131	047524	043440	
	010170	020117	021445		
1145					
1146	010174	042445	051122	051117	MSG46: .ASCII /&ERROR BIT OK&/
	010202	041040	052111	047440	
	010210	022513	043		
1147	000001				.END

DECGUT	004674	845	851e																	
DECPNT	004770	838e	842	844	849e	869e														
DECPRT	004604	722	837e																	
DEC1	004722	852	854	857e																
DEC2	004736	856	859e	862																
DEC3	004760	860	864e																	
DIGCNT	004764	837e	846e	853	867e															
DIGIT	004762	840e	841e	851	855e	858e	859	863	866e											
DINT	004232	691	710e																	
LINE	004312	687	728e																	
LKS	001046	124e	684e	695	697e	716e														
MSG0	005512	747	1065e																	
MSG1	005535	168	1067e																	
MSG10	006105	978	1082e																	
MSG11	006131	505	1084e																	
MSG12	006241	269	1087e																	
MSG13	006276	261	1089e																	
MSG14	006337	294	1091e																	
MSG15	006411	305	1093e																	
MSG16	006460	317	1095e																	
MSG17	006510	325	1097e																	
MSG2	005564	170	1069e																	
MSG20	006546	336	1099e																	
MSG21	006611	348	1101e																	
MSG22	006660	358	1103e																	
MSG24	006727	380	1105e																	
MSG25	006770	405	1107e																	
MSG26	007035	522	1109e																	
MSG27	007123	779	1112e																	
MSG3	005624	185	1071e																	
MSG30	007140	781	1114e																	
MSG31	007161	798	1116e																	
MSG32	007202	803	1118e																	
MSG33	007205	586	1121e																	
MSG34	007246	431	1123e																	
MSG35	007323	450	1125e																	
MSG36	007376	473	1127e																	
MSG37	007457	489	1129e																	
MSG4	005660	191	1073e																	
MSG40	007536	1042	1131e																	
MSG41	007573	277	1133e																	
MSG42	007632	285	1135e																	
MSG43	007673	720	1137e																	
MSG44	007720	500	1139e																	
MSG45	010061	503	1143e																	
MSG46	010174	509	1146e																	
MSG5	005722	204	1075e																	
MSG6	005760	216	507	1077e																
MSG7	006067	824	1080e																	
PBEG	001206	89	150e	1046																
PC	MSG00007	51e	169e	174e	179e	186e	192e	197e	205e	217e	222e	258e	392e	495e						
		501e	504e	506e	508e	510e	512e	517e	524e	569e	587e	592e	632e	721e						
		722e	749e	751e	780e	782e	799e	804e	808e	825e	845e	848e	864e	889e						
		892e	894e	899e	910e	922e	936e	949e	976e	999e	1043e									

LPC11F.SPC CROSS REFERENCE TABLE

PCR	001056	128*	291*	342*	375*	424*	445*	559*	580*	648*	698*			
PCHAN	001022	111*	555*	558*	559	567	579*	560	590	593*	594	784		
PCLF	001244	160*	162											
PCONTD	004542	778	805*											
PCONTL	004352	746	752*											
PCS	001054	127*	208	254*	259	275	292	334	356	365*	396*	411*	439*	460*
		481*	485*	516	552*	631*	638*	663*	718*	967*				
PDA1	003374	498	511	550*										
PDA10	003400	551*	571	574										
PDA11	003440	559*	573											
PDA110	003626	591	593*											
PDA111	003660	597	600*											
PDA17	003452	561*	564											
PDA12A	003474	562	566*											
PDA13	003517	568	570*											
PDA14	003532	557	578*											
PDA15	003540	579*	599											
PDA16	003544	580*	595											
PDA17	003556	582*	585											
PDA17A	003574	565	566*											
PDA17B	003610	583	589*											
PDE	004404	569	592	777*										
PDE1	004442	765*	602											
PDE2	004446	786*	787	795										
PDE3	004472	769	792*											
PDE4	004500	791	793*											
PDE5	004532	797	803*											
PEB1	003176	483	494*											
PEBT0	003224	497	499*											
PEBT1	003312	501	504	506	508	512*								
PEBT2	003344	518*	757											
PEBT3	003372	521	524*											
PEND	004556	588	600	822*										
PET	001034	116*	158	499*	502*	520								
PGO	004400	758*												
PGOD	004554	806	808*											
PGOF	001012	107*	218	513										
PGOL	004364	753	755*	756										
PIC	001020	110*	551*	572*	578*	598*								
PINI	005236	368	394	965*										
PINTE	005316	969	978*											
PINIF	001032	115*	373*	398*	421*	448*	468*	645*	950	977				
PINIS	005224	419	466	636	643	948*								
PINT1	005272	973*	974											
PITA	001010	106*	414*	430	465*									
PLC	004112	234	682*											
PLCC	001570	227	232*	726										
PLCG	004306	724	726*											
PLCF	004242	705	716*											
PLC1	004174	695*	696											
PLC2	004212	698*	700											
PLE	004316	262	270	278	286	295	306	318	326	337	349	359	381	406
		432	451	474	490	523	745*	979						
PLOOP	004600	823	827*											

PPATE	005510	556	1057e												
PPA1S	005466	550	1055e												
PREG	001002	103e	193	198	200	202	207								
PREGE	001444	199	201	204e											
PRG	001356	183	191e												
PRG1	001366	193e	206												
PRSTV	005326	258	392	495	632	949	976	991e							
PSCOP	003664	228	625e	628											
PSCOP0	003672	627e	657	660	663										
PSCOP1	003756	634	640e												
PSCOP2	004002	639	645e												
PSCOP3	004036	651e	652	654											
PSCOP4	004052	645	655e												
PSCOP5	004070	659e	662												
PSET	001456	203	207e												
PSKIP	001200	92	148e	758	827										
PSMC	001554	164	226e												
PSPS	001006	105e	1017e	1031											
PSTART	001210	149	151e												
PSW	001052	126e	152e	374e	399e	422e	429e	447e	469e	629e	650e	694e	971e	975e	
		1030e													
PTB	001062	130e	437e	458e	479e	554e	566	589							
PTIM1	001014	108e	300e	301e	312e	313e	331e	332e	343e	344e	354e	371e	376e	387e	
		388e	390e	401e	423e	425e	427e	443e	470e	560e	563e	581e	584e	647e	
		651e	658e	661e	972e	973e									
PTIM2	001016	109e	372e	378e	397e	403e	646e	653e							
PTN	001004	104e	257e	266e	274e	282e	290e	299e	310e	322e	330e	341e	353e	363e	
		385e	410e	436e	457e	478e	494e	748							
PTS	001060	129e	256e	267	283	303	311e	315	323	346	364e	370e	386e	400e	
		412e	438e	446e	459e	471e	480e	553e	561	582	630e	640e	649e	659	
		682e	686e	688e	711e	717e	966e								
PT1	001604	233	255e												
PT1A	002064	313e	314												
PT1A1	001646	260	266e												
PT1A1A	001674	268	274e												
PT1A1B	001722	276	282e												
PT1A2	001750	284	290e												
PT1A3	002004	293	299e												
PT1A3A	002016	301e	302												
PT1A4	002044	304	310e												
PT1b	002220	344e	345												
PT1B1	002112	316	322e												
PT1B2	002140	324	330e												
PT1b2A	002152	332e	333												
PT1b3	002200	335	341e												
PT1C	002246	347	353e	355											
PT1E	002302	357	363e												
PT1F	002376	376e	377	379											
PT1fA	002422	373	385e												
PT1fP	002440	388e	389												
PT1fC	002446	390e	391												
PT1G	002526	401e	402	404											
PT1H	002552	398	410e												
PT1HA	002614	419e													

PT1HB	002624	421*												
PT1J	002646	425*	426	449	472									
PT1K	002654	427*	428											
PT1L	002674	415	431*											
PT1M	002704	414	436*											
PT1M1	002734	443*	444											
PT1N	002774	446	450*											
PT1P	003004	441	457*											
PT1Q	003060	468*	488											
PT1R	003104	464	473*											
PT1S	003114	465	478*											
PT1T	003166	487	489*											
PVE	001344	181	185*											
PVEC	001000	102*	175	180	182	366	393	417	462	635	641	689	991	
PV1	001276	175*	187											
PWRFAL	005352	64	1010*											
PWRUP	005406	1018	1030*											
READ	005102	179	197	222	517	909*								
READ1	005104	910*	921											
REX	005150	912	922*											
ROT	005124	914*	917											
ROTA	001030	114*	177*	195*	220*	515*	913							
RO	=0000000	44*	154*	155*	158*	159*	160*	207*	212*	366*	367	368*	369*	393*
		394*	395*	417*	418	419*	420*	462*	466*	467*	518*	519*	520	566*
		567	589*	590	635*	636*	637*	641*	642	643*	644*	689*	690	691*
		692*	693*	698	704*	801	968	969*	970*	991*	992	994*	995*	997*
		998*	1010	1038*										
R1	=0000001	45*	157*	161*	208*	212*	550*	556	558	687*	719*	913*	916*	992*
		993	994	996	997	1011	1037*							
R2	=0000002	46*	148*	150*	163	625*	626	627	1012	1036*				
R3	=0000003	47*	176*	194*	219*	261*	269*	277*	285*	294*	305*	317*	325*	336*
		348*	358*	380*	405*	416*	422	431*	450*	463*	469	473*	486*	489*
		514*	522*	626*	648	655*	685*	710*	719	750	783*	796	800*	842*
		844*	920*	1013	1035*									
R4	=0000004	48*	168*	170*	175*	185*	191*	193*	204*	216*	218*	415*	421	464*
		468	487*	500*	503*	505*	507*	509*	513*	586*	720*	747*	748*	750*
		779*	781*	784*	788	793*	798*	801*	803*	824*	884	909*	915*	919*
		978*	1014	1034*	1042*									
R5	=0000005	49*	178*	196*	221*	413*	440	442*	461*	482	484*	516*	785*	794*
		918	1015	1033*	1039*	1040*	1044*							
SP	=0000006	50*	151*	948	965	1010*	1011*	1012*	1013*	1014*	1015*	1016*	1017	1031*
		1032	1033	1034	1035	1036	1037	1038						
S#R	001050	125*	226	232	496	570	596	625	627	633	656	723	745	782
		755	777	805	822									
TCRLF	005036	888	891*											
TEX	005100	886	899*											
TIB	001026	113*	911	918*	919	928*	932*	935						
TIME	004314	693	729*											
TINT	004224	70	704*											
TKB	001040	121*	927*	932										
TAS	001036	120*	926*	929*	930									
TOB	001024	112*	884*	885	887	891*	893*	898						
TOG	005064	889	892	894	896*	897								
TPB	001044	123*	790*	792*	863*	898*	935*							

TPS	001042	122*	786	861	896	933								
TTIN	005152	910	926*											
TTIN1	005172	930*	931											
TTIN2	005206	933*	934											
TTOUT	005004	169	174	186	192	205	217	510	512	587	721	749	751	780
		782	799	804	825	864*	890	895	1043					
TYPT1	004624	840*	850											
TYPT2	004632	841*	843											
TYPT3	004664	847	849*											
ZERO	004766	839*	855	857*	868*									
.	= 010213	55*	59	63*	69*	88*	91*	98*	133*	156	869	1041	1045	1058*

LPC11F.SPC CROSS REFERENCE TABLE

ADD	212	844	849												
BEG	227	233	260	268	276	284	293	316	357	521	557	568	591	595	657
	696	705	753	756	789	806	854	860	886	888	912				
BHI	183	201	203												
BIC	519	918													
BIS	370	396	446	485	638	640	688	697	858	919					
BIT	180	198	226	232	259	267	275	283	292	303	315	323	334	346	356
	496	561	570	582	594	596	633	656	659	695	723	745	752	755	777
	805	822	948												
BITB	788														
RNE	156	162	164	181	199	302	304	314	324	333	335	345	347	355	377
	379	389	391	402	404	426	428	441	444	483	497	562	564	571	573
	583	585	597	599	628	634	652	654	660	662	724	746	778	795	797
	823	847	852	917	921	974	1041	1045							
BPL	787	843	862	897	931	934									
BH	149	187	206	574	639	791	850	856	890	895					
CLC	914														
CLR	150	154	159	160	255	256	300	331	343	364	365	371	374	386	387
	399	411	412	413	416	423	437	438	439	445	458	459	460	461	463
	470	479	480	481	499	552	553	554	555	560	579	581	630	631	647
	650	658	682	683	684	685	694	716	717	718	783	839	909	926	927
	928	966	967	971	972	995	998	1039							
CMP	182	200	202	367	418	520	556	642	690	853	911	965	968	996	
CMPH	567	590	627	885	887										
COM	655														
DEC	155	161	301	313	332	344	354	376	378	388	390	401	403	425	427
	443	563	572	584	598	651	653	661	704	794	916	920	973	1040	1044
DIV	719														
HALT	59	725	754	807	826	1019									
INC	311	400	442	471	484	593	649	686	710	711	800	841	846	929	
JMP	89	92	228	234	262	270	278	286	295	306	318	326	337	349	359
	381	406	430	432	449	451	472	474	488	490	498	511	523	565	588
	600	663	700	726	757	758	802	827	950	977	979	1046			
JSR	169	174	179	186	192	197	205	217	222	258	392	495	501	504	506
	508	510	512	517	569	587	592	632	721	722	749	751	780	782	799
	804	825	845	889	892	894	910	949	976	1043					
MOV	148	151	152	157	158	168	170	175	176	177	178	185	191	193	194
	195	196	204	207	208	212	216	218	219	220	221	257	261	266	269
	274	277	282	285	290	291	294	299	305	310	312	317	322	325	330
	336	341	342	348	353	358	363	366	368	369	372	373	375	380	385
	393	394	395	397	398	405	410	414	415	417	419	420	421	422	424
	429	431	436	447	448	450	457	462	464	465	466	467	468	469	473
	478	486	487	489	494	500	502	503	505	507	509	513	514	515	516
	518	522	550	551	578	580	586	625	626	629	635	636	637	641	643
	644	645	646	687	689	691	692	693	698	720	747	748	750	779	781
	784	785	790	792	798	801	803	824	837	838	840	855	857	863	913
	932	969	970	975	978	991	992	994	997	1010	1011	1012	1013	1014	1015
	1016	1017	1018	1030	1031	1032	1033	1034	1035	1036	1037	1038	1042		
MOVB	558	559	566	589	648	884	891	893	898	935					
RESET	153														
ROL	793	915													
RTI	706	712													
RTS	524	808	848	864	899	922	936	999							
SUB	842														

LPC11H.SRC CROSS REFERENCE TABLE

TST	163	440	482	796	851	859	993								
TSTh	786	861	896	930	933										
WAIT	699														
.ABS	3														
.ASCII	1065	1067	1069	1071	1073	1075	1077	1078	1080	1082	1084	1085	1087	1089	1091
	1093	1095	1097	1099	1101	1103	1105	1107	1109	1110	1112	1114	1116	1118	1121
	1123	1125	1127	1129	1131	1133	1135	1137	1139	1140	1141	1143	1144	1146	
.BYTE	1055	1056	1057												
.END	1147														
.EVEN	1054	1058													
.REPT	56	209													
.TITLE	2														

ERRORS DETECTED: 0

LPC11 MACY11.024 20-JUN-73 14132 PAGE 10-13
LPC11B.SRC

oLPC11B/I,LPC11B/CRF,LPC11B.SRC
RUN-TIME: 3 7 1 SECONDS
CORE USED: 6K