

# LPC11

INTERFACE DIAGNOSTIC TEST  
MD-11-DZLPB-B

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

JUN 1978  
**digital**  
MADE IN USA

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



IDENTIFICATION

PRODUCT CODE:	MAINDEC-11-02LFB-3-0
PRODUCT NAME:	LPC11 INTERFACE DIAGNOSTIC TEST
DATE CREATED:	FEBRUARY 25, 1973
MAINTAINERS:	DIAGNOSTIC GROUP
AUTHOR:	R.J. 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 298 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 SEQUENTIALLY; (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 FLEXIBILITY 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
  - 1. 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 298 TO ENTER NEW VECTOR AND REGISTER ADDRESSES; SET CONSOLE SWITCHES BEFORE TYPING 'G'.
- B. LOAD AND START 284 TO BYPASS ENTRY OF NEW ADDRESSES; SET CONSOLE SWITCHES BEFORE TYPING 'G'.
- C. DUE TO FLOATABLE VECTOR AND REGISTER USAGE, A START AT 298 MUST BE DONE ON THE INITIAL RUN. THEREAFTER 284 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 288(8).

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 151 1=NO ERROR PRINTS  
2=PRINT ALL ERRORS
  
- SW 141 1=STOP ON ERROR  
2=CONTINUE ON ERROR
  
- SW 131 1=LOOP MODE  
2=SINGLE PASS
  
- SW 121 1=INHIBIT DATA ITERATIONS  
2=DO NOT INHIBIT ITERATIONS
  
- SW 111 1=GO TO MAINTENANCE MODE  
2=DO NOT ENTER MAINTENANCE MODE
  
- SW 101 1=DO ERROR BIT TEST  
2=DO NOT DO ERROR BIT TEST
  
- SW 91 1=USE TESTER INTERRUPT FOR MAINTENANCE MODE  
2=USE CONTROLLER INTERRUPT
  
- SW 81 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 CONDITION'S 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 CONDITION'S 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

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 (2<sup>8</sup> = 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 31 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 0-7 AND MAY BE VARIED DYNAMICALLY. THE DATA IS COMPLEMENTED ON EVERY OTHER PASS TO PROVIDE TRANSITIONS IN THE LOGIC.
- D. GROUP 41 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.

A. 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
!#INDEC-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			IREGISTER 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			IRAP CATCHERS 0-1000
54			
55		000000	.=0
56			.NEPT 200
57			.=2
58			HALT
59			.ENDP
60			
61			IPower FAIL VECTOR
62			
63		000024	.=24
64	000024	005352	PARFAL
65	000026	000340	340
66			
67			ILINE 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 000200 000200 001002  
90  
91 000204 000204 000770  
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  
      JMP      PBEG          !GO TO START AND SET VECTOR / REGISTERS  
  
      .=204  
      JMP      PSKIP        !GO TO START BUT DO NOT SET ADDRESSES
```

97				
98	001000		.#1000	
99				
100			IFLAGS AND COUNTERS	
101				
102	001000	000000	PVEC: 0	IVECTOR POINTER SAVE
103	001002	000000	PHEG: 0	IREGISTER POINTER SAVE
104	001004	000000	PTN: 0	ITEST NUMBER STORAGE
105	001006	000000	PSPB: 0	ISTACK POINTER SAVE LOC
106	001010	000000	PITA: 0	IINTERRUPT RETURN STORAGE
107	001012	000000	PGOF: 0	IGO FLAG
108	001014	000000	PIIM1: 0	ITIMEH
109	001016	000000	PIIM2: 0	IGROSS TIMEH
110	001020	000000	PICI: 0	ITERATION CNTR
111	001022	000000	PCHAR: 0	ITEST CHAR TEMP STORAGE
112	001024	000000	TUBI: 0	ITTY OUTPUT BUFFER STORAGE
113	001026	000000	TIBI: 0	ITTY INPUT BUFFER STORAGE
114	001030	000000	ROTA: 0	IROTATION FACTOR
115	001032	000000	PINTR: 0	IINTERRUPT RETURN POINTER
116	001034	000000	PLT: 0	IERNOM BIT TEMP STORAGE
117				
118			ADDRESS CONSTANTS	
119				
120	001036	177560	TKB: 177560	ITTY STATUS
121	001040	177562	TKB: 177562	ITTY BUFFER
122	001042	177564	TPB: 177564	ITTP STATUS
123	001044	177566	TPB: 177566	ITTP BUFFER
124	001046	177566	LKB: 177566	ILINE CLOCK REGISTER
125	001050	177570	SWR: 177570	ICONSOLE SWITCH REGISTER
126	001052	177776	PSW: 177776	IPROGRAM STATUS WORD
127	001054	000000	PCB: 0	IPACESETTER STATUS
128	001056	000000	PCB: 0	IPACESETTER BUFFER
129	001060	000000	PTB: 0	ITEST STATUS
130	001062	000000	PTB: 0	ITEST 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      R1,R2          !SET SKIP FLAG
149                  BR      PSTART        !GO TO START
150          PNEG:   CLR      R2           !RESET SKIP FLAG
151          PSTART: MOV      R500,R5      !SET STACK POINTER
152                  MOV      R340,R5W    !SET PSW
153                  RESET                !RESET AT START OR RESTART
154                  CLR      R0
155                  DEC     R0
156                  BNE     .-2          !DELAY FOR RESET
157                  MOV     R14,R1       !R1=NUMBER OF LOCATIONS TO CLEAN
158                  MOV     R0,R1        !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     P8MC        !IF NOT; BR
165
166          !PRINT PROGRAM NAME
167
168          MOV     R8,R4                !MSG1,R4
169          JSR    PC,TTOUT              !TYPE PROGRAM NAME
170          MOV     R8,R4                !MSG2,R4
171
172          !GET NEW VECTOR ADDRESS
173
174          JSR    PC,TTOUT              !TYPE VECTOR REQUEST
175          MOV     R4,R4                !R4=VECTOR STORAGE ADDR
176          MOV     R3,R3                !R3=INPUT CHAR NUMBER
177          MOV     R3,R3                !R3=ROTATION FACTOR
178          MOV     R5,R5                !R5=MASK
179          JSR    PC,READ              !GO READ IN VECTOR
180          BIT     R1,PVEC             !TEST FOR ODD ADDR
181          BNE     PVE                 !IF ODD; BR
182          CMP     R500,PVEC           !TEST FOR TOO HIGH
183          BMI     PHG                 !IF OK; BR
  
```

```

145 001344 012704 005024          PVEI  MOV      @MSG3,R4
186 001350 004767 003430          JSR      PC,ITOUT      ;TYPE ILLEGAL VECIUM
187 001354 000750          BH       PVI           ;TRY AGAIN
188
189                                ;GET NEW REGISTER ADDRESS
190
191 001356 012704 005060          PRG1  MOV      @MSG4,R4
192 001362 004767 003416          JSR      PC,ITOUT      ;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          PREG1  MOV      @MSG5,R4
205 001450 004767 003330          JSR      PC,ITOUT      ;TYPE ILLEGAL REGISTER ADDR
206 001454 000744          BR      PRG1         ;TRY AGAIN
207 001456 016700 177320          PSET1  MOV      @PREG,R0     ;SET P. S. STATUS ADDR IN R0
208 001462 012701 001054          MOV      @PCS,R1     ;HIDPACESSETTER STATUS ADDR
209
210                                .REPT 4
211                                MOV      R0,(R1)+     ;SET REGISTERS
212                                ADD      @2,R0
213                                .ENDR
214
215                                ;REQUEST ON LINE/TEST MODE
216
216 001516 012704 005760          MOV      @MSG6,R4
217 001522 004767 003256          JSR      PC,ITOUT      ;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
222 001550 004767 003326          JSR      PC,READ      ;SET MASK
223
224                                ;MAINTENANCE MODE CHECK
225
226 001554 032777 004000 177266  PSMCI  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  PLCCI  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

```



234  
 235  
 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 ASSURE THAT ALL FUNCTIONS OF EACH USABLE BIT IN BOTH THE TESTER AND THE CONTROLLED STATUS REGISTERS WILL SET AND RESET FROM PROPER CONDITIONS AND IN THE PROPER SEQUENCE. THE INTERRUPT SYSTEM OF BOTH THE TESTER AND CONTROLLED IS ALSO TESTED TO ASSURE THAT EACH WILL TRAP TO THE PROPER VECTOR AND PRIORITY LEVEL.  
 \*\*\*\*\*

LOGIC TEST 01: CLEAR CONT STATUS

```

PT1: CLR      OPCS      ;CLEAR CONT STATUS
      CLR      OPTS      ;CLEAR TESTER STATUS
      MOV      @30460,PTN ;SAVE TEST NO.
      JSR      PC,PRSTV   ;RESET VECTORS
      BIT      @77500,OPCS ;LOOK FOR CLEAR
      BEQ      PT1A1      ;IF OK: BR
      MOV      @MSG13,R3  ;SET ERROR MSG ADDR
      JMP      PLE        ;GO TYPE ERROR
  
```

LOGIC TEST 02: CLEAR TESTER STATUS

```

PT1A1: MOV      @31060,PTN ;SAVE TEST NO.
        BIT      @77577,OPTS ;LOOK FOR CLEAR
        BEQ      PT1A1A      ;IF OK: BR
        MOV      @MSG14,R3  ;SET ERROR MSG ADDR
        JMP      PLE        ;GO TYPE ERROR MSG
  
```

LOGIC TEST 03: CONT READY RESET FROM INIT

```

PT1A1A: MOV      @31460,PTN ;SAVE TEST NO.
         BIT      @200,OPCS  ;LOOK FOR CONT READY RESET
         BEQ      PT1A1B      ;IF OK: BR
         MOV      @MSG41,R3  ;SET ERROR MSG ADDR
         JMP      PLE        ;GO TO ERROR ROUTINE
  
```

LOGIC TEST 04: TESTER READY RESET FROM INIT

```

PT1A1B: MOV      @32060,PTN ;SAVE TEST NO.
         BIT      @200,OPTS  ;LOOK FOR TESTER READY RESET
         BEQ      PT1A2      ;IF OK: BR
         MOV      @MSG42,R3  ;SET ERROR MSG ADDR
         JMP      PLE        ;GO TO ERROR ROUTINE
  
```

LOGIC TEST 05: CONT READY RESET FROM CHAR LOAD

```

PT1A2: MOV      @32460,PTN ;SAVE TEST NO.
        MOV      @0,OPCS    ;CLEAR BUFFER
  
```

292	001764	032777	000200	177062		BIT	0200,0PCS	LOOK FOR READY
293	001777	001404				BEO	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	PT1A31	MOV	033060,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	0200,0PTS	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	PT1A41	MOV	034060,PTN	SAVE TEST NO.
311	002052	005277	177002			INC	0PTS	SET GO
312	002056	012767	000100	176730		MOV	0100,PTIM1	SET TIME
313	002064	005367	176724		PT1A1:	DEC	PTIM1	DELAY
314	002070	001375				BNE	PT1A	
315	002072	032777	000001	176760		BIT	01,0PTS	LOOK FOR GO RESET
316	002100	001404				BEO	PTIM1	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	PT1B11	MOV	030061,PTN	SAVE TEST NO.
323	002120	032777	000200	176732		BIT	0200,0PTS	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	PT1B21	MOV	030461,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	0200,0PCS	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	PT1B31	MOV	032601,PTN	SAVE TEST NO.
342	002206	012777	000000	176642		MOV	00,0PCH	LOAD CHAR
343	002214	005067	176574			CLR	PTIM1	SET TIME
344	002220	005367	176570		PT1B1:	DEC	PTIM1	DELAY
345	002224	001375				BNE	PT1B	

```

346 002226 032777 000200 176024 BIT      @200,@PTS      ;LOOK FOR TESTER READY
347 002234 001004 BNE      PT1C      ;IF OK: BR
348 002236 012703 MOV      @MSG21,F3 ;SET ERHOM MSG ADDR
349 002242 000167 JMP      PLE       ;GO TYPE ERROR MSG
350
351 ;LOGIC TEST 13: CONT HEADY RESET AT END OF CYCLE
352
353 002246 012767 031461 176530 PT1C:  MOV      @31461,PTM ;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 JMP      PLE       ;GO TYPE ERROR MSG
360
361 ;LOGIC TEST 14: TESTER INTERRUPT
362
363 002302 012767 031461 176474 PT1E:  MOV      @31461,PTM ;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 CHAN
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 ERHOM MSG ADDR
381 002416 000167 JMP      PLE       ;GO TYPE ERROR MSG
382
383 ;LOGIC TEST 15: CONT INTERRUPT
384
385 002422 012767 032461 176354 PT1FA: MOV      @32461,PTM ;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	OPTS	);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	001944			JMP	PLE	);GO TYPE ERROR MSG
407								
408								
409								);LOGIC TEST 16: TESTER INT ENABLE OFF
410	002552	012767	033061	176224	PT1H:	MOV	033061,PTM	);SAVE TEST NO.
411	002560	005077	176270			CLR	OPCB	
412	002564	005077	176270			CLR	OPTS	
413	002570	005005				CLR	R5	);CLEAR FLAG
414	002572	012767	002704	176210		MOV	PTIM,PITA	);SET POINTER
415	002600	012704	002674			MOV	PTIL,R4	);SET RETURN ADDRESS
416	002604	005003				CLR	R3	
417	002606	016700	176166			MOV	PVEC,R0	);HOBTAINING VECTOR ADDRESS
418	002612	022020				CMP	(R0)+,(R0)+	);POINT TO TESTER VECTOR
419	002614	012720	005224		PT1HA:	MOV	PTINT,(R0)+	);SET VECTOR TO SERVICE ROUTINE
420	002620	012720	000340			MOV	0340,(R0)+	);SET PSW TO PRIORITY 7
421	002624	010467	176202		PT1HB:	MOV	R4,PINTR	);SET INTERRUPT RETURN
422	002630	010377	176216			MOV	R3,OPSW	);SET PSW
423	002634	005067	176154			CLR	PTIM1	);CLEAR TIMER
424	002640	012777	000000	176210		MOV	00,OPCB	);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	0340,OPSW	);RESET PSW TO PRIORITY 7
430	002670	000177	176114			JMP	OPITA	);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	033461,PTM	);SAVE TEST NO.
437	002712	005077	176144			CLR	OPTB	
438	002716	005077	176136			CLR	OPTS	
439	002722	005077	176126			CLR	OPCB	
440	002726	005705				TST	R5	);SEE IF TESTER COMPLETE
441	002730	001029				BNE	PT1P	);IF SO:WR
442	002732	005405				INC	R5	);SET FLAG
443	002734	005367	176054		PT1M1:	DEC	PTIM1	
444	002740	001375				BNE	PTIM1	);DELAY
445	002742	005077	176110			CLR	OPCB	);LOAD CHAR
446	002746	052777	000100	176104		BIS	0100,OPTS	);SET TESTER INTERRUPT ENABLE
447	002754	012777	000200	176070		MOV	0200,OPSW	);PSW SETTING = PRIORITY 4
448	002762	012767	002774	176042		MOV	PTIN,PINTR	);LOAD RETURN ADDRESS
449	002770	000167	177652			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,PTM      ;SAVE TEST NO.
458 003012 005077 176044          CLR      OPTB
459 003016 005077 176036          CLR      OPTS
460 003022 005077 176026          CLM      @PCS
461 003026 005005          CLR      R5          ;CLEAN FLAG
462 003030 016700 175744          MOV      PVFC,R0      ;RO=VECTION ADDRESS
463 003034 005003          CLR      R3          ;R3=PSW
464 003036 012704 003104          MOV      @PT1R,R4      ;R4=INTERHUPT RETURN ADDR
465 003042 012767 003114 175740          MOV      @PT1S,PTA      ;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      OPTS         ;SET GO
472 003100 000167 177542          JMP      PT1J         ;TEST CONT INTERHUPT
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,PTM      ;SAVE TEST NO.
479 003124 005077 175734          CLR      OPTB
480 003126 005077 175726          CLR      OPTS
481 003132 005077 175716          CLR      @PCS
482 003136 005705          TST      R5          ;SEL IF DONE BOTH TESTS
483 003140 001016          RNE      PEBT        ;IF BOTH
484 003142 005205          INC      R5          ;SET FLAG
485 003144 052777 000100 175702          BIS      @100,@PCS    ;SET CONTROLLER INTERHUPT ENABLL
486 003152 012703 000200          MOV      @200,R3      ;SET PSW
487 003156 012704 003166          MOV      @PT1T,R4      ;SET INTERHUPT RETURN ADDR
488 003162 000167 177672          JMP      PT1Q        ;TEST CONTROLLER
489 003166 012703 007457          PT1T: MOV      @MSG37,R3
490 003174 000167 001120          JMP      PLE          ;GO TO ERROR ROUTINE
491
492
493
494 003176 012767 031062 175600 PEBT: MOV      @31062,PTM      ;SAVE TEST NUMBER
495 003204 004767 002116          JSR      PC,PRSTV      ;RESET VECTORS.
496 003210 032777 002000 175632          BIT      @2000,@BWR    ;SEL IF ERROR BIT TEST
497 003216 001002          BNE      PEBT0        ;IF 801 BK
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,PEHT1	
509	003276	012704	010174		MOV	MSG46,R4	
510	003302	004767	001476		JSR	PC,ITOUT	ITYPE ERROR HIT OK
511	003306	000167	000062		JMP	PDAI	IGO TO DATA TEST
512	003312	004767	001466	PEBT11	JSR	PC,ITOUT	ITYPE MSG
513	003316	016704	175470		MOV	PGOP,R4	
514	003322	012703	000001		MOV	R1,R3	
515	003326	012767	000003	175474	MOV	R3,MOIA	
516	003334	012705	177770		MOV	R177770,R5	
517	003340	004767	001536		JSR	PC,RFAD	IAWAIT GO REPLY
518	003344	017700	175504	PEBT12	MOV	OPCS,R0	ISAVE CONTROLLER STATUS
519	003350	042700	077777		BIC	R0,PEI	IMASK ERROR BIT
520	003354	020067	175454		CMP	R0,PEI	ISEE IF ERROR BIT IN PROPER STATE
521	003360	001404			BLO	PEBT3	IIF 501 BR
522	003362	012703	007035		MOV	MSG26,R3	
523	003366	000167	000724		JMP	PLE	IGO TYPE ERROR MSG
524	003372	000207		PEBT13	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 005910          CMP      @PPATE,R1          ;LOOK FOR END OF DATA TABLE
557 003432 001437                    BEQ      PDAT4              ;IF 801BR
558 003434 112167 175362          MOVB    (R1)+,PCHAR        ;SET CHAR
559 003440 116777 175356 175410 PDAT11: MOVB    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     PDAT12A            ;IF HAVE1 BH
563 003462 005367 175326          DEC     PTIM1              ;DELAY
564 003466 001371                    BNE     PDAT2              ;TYPE ERROR
565 003470 000167 000100          JMP     PDAT7A              ;NO INPUT CHAR
566 003474 117700 175362          PDAT2A: MOVB    @PTB,R0      ;COMPARE INPUT/OUTPUT
567 003500 120067 175316          CNPB   R0,PCHAR
568 003504 001402                    BLO     PDAT3              ;IF OK1 BR
569 003506 004767 000672          JSR    PC,PDL              ;GO TYPE ERROR
570 003512 032777 010000 175330 PDAT3: BIT      @10000,@S=R   ;LOOK FOR INHIBIT ITERATIONS
571 003520 001327                    BNE     PDAT0              ;IF 801 BR
572 003522 005367 175272          DEC     PIC                ;DECREMENT ITERATION COUNTER
573 003526 001344                    BNE     PDAT1              ;IF NOT DONE1 BH
574 003530 000723                    BH      PDAT0              ;ELSE GET NEXT PATTERN
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		CLP	PTIM1	ICLEAR 11MEN
582	003556	032777	000200	175274	PDAT7:	#200,#PTS	ILOOK FOR READY BIT SET
583	003564	001011			HNE	PDAT7B	IIF ONI BR
584	003566	005367	175222		DEC	PTIM1	
585	003572	001371			BNE	PDAT7	IWAIT READY
586	003574	012704	007205		PDAT7A:	MSG33,H4	
587	003600	004767	001200		JSR	PC,ITOUT	IIF TYPE NO READY IN DATA TEST
588	003604	000167	000746		JMP	PEND	IIF GO TO END ROUTINE
589	003610	117700	175246		PDAT7B:	WPTB,RO	IIF NO INPUT CHAR
590	003614	120067	175202		CMPL	RO,PCHAR	IIF TEST CHAR
591	003620	001402			BEG	PDAT10	IIF ONI BR
592	003622	004767	000556		JSR	PC,PDL	IIF GO TYPE ERROR
593	003626	005267	175170		PDAT10:	PCHAR	IIF SET NEW CHAR
594	003632	032767	000400	175162	BIT	#400,PCHAR	IIF SEE IF DONE ALL
595	003640	001741			BEG	PDA16	IIF NOTI BR
596	003642	032777	010000	175200	BIT	#10000,#S#R	IIF LOOK FOR INHIBIT ITERATIONS
597	003650	001003			BNE	PDAT11	IIF NOTI BR
598	003652	005367	175142		DEC	PIC	IIF LOOK FOR COMPLETE ITERATIONS
599	003656	001330			BNE	PDAT5	IIF NOTI BR
600	003660	000167	000672		PDAT11:	JMP	IIF GO TO END ROUTINE
601						PEND	

```

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      @SWR,R2      ;R2 = DATA FROM SWITCHES
626 003670 010403              MOV      R2,R3
627 003672 127702 175152      PSCOP0: CMPB     @SWR,R2      ;SEE IF NEW DATA
628 003676 001372              BNE     PSCOP      ;IF SUI BR
629 003700 012777 000340 175144  MOV      @340,@PSW      ;HSET PSW TO PRIORITY 7
630 003706 005077 175146      CLR     @PTS          ;CLEAR STATUS REGISTERS
631 003712 005077 175136      CLR     @PCB
632 003716 004767 001404      JSR     PC,PRSTV      ;HSET VECTORS
633 003722 032777 001000 175120  BIT     @1000,@SWR     ;SEE IF TESTER INTERRUPT
634 003730 001012              BNE     PSCOP1       ;IF SUI BR
635 003732 016700 175042      MOV     PVEC,R0       ;R0 = STARTING VECTOR
636 003736 012720 005224      MOV     @PINTS,(R0)+  ;SET VECTOR FOR MAINTENANCE MODE TRAP
637 003742 012720 000340      MOV     @340,(R0)+
638 003746 052777 000100 175100  BIS     @100,@PCB     ;SET CONTROLLER INTERRUPT ENABLE
639 003754 000412              BR      PSCOP2
640 003756 052777 000100 175074  PSCOP1: BIS     @100,@PTS   ;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     @PINTS,(R0)+  ;SET VECTOR ADDR TO MAINTENANCE MODE TRAP
644 003776 012720 000340      MOV     @340,(R0)+
645 004002 012767 004052 175022  PSCOP2: MOV     @PSCOP4,PINTR ;SET RETURN ADDR
646 004010 012767 000010 175000  MOV     @10,PTIM2     ;SET DELAY
647 004016 005067 174772      CLR     PTIM1         ;SET DELAY
648 004022 110377 175030      MOVB   R3,@PCB       ;LOAD CHAR
649 004026 005277 175026      INC     @PTS          ;SET GO BIT
650 004032 005077 175014      CLR     @PSW         ;SET TO PRIORITY 0
651 004036 005367 174752      PSCOP3: DEC     PTIM1
652 004042 001375              BNE     PSCOP3
653 004044 005367 174746      DEC     PTIM2
654 004050 001372              BNE     PSCOP3
655 004052 005103      PSCOP4: COM     R3
656 004054 032777 001000 174766  BIT     @1000,@SWR     ;SEE IF ON TESTER INTERRUPT
  
```

057	004062	001703		HEQ	PSCOPO		IF NOT: BR
058	004064	005067	174724	CLR	PTIM1		
059	004070	032777	000200 174762	BIT	8200,0PTS		LOOK FOR TESTER READY
060	004076	001275		BNE	PSCOPO		IF 801 BR
061	004100	005367	174710	DEC	PTIM1		
062	004104	001371		BNE	PSCUP5		DELAY FOR COMPLETE FROM TESTER
063	004106	000167	177560	JMP	PSCOPO		CONTINUE



```

669
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      OPCS          !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 000182          MOV      LINE,R1       !SET DIVISOR
688 004140 052777 000100 174712  BIS      @100,OPTS     !SET TESTER INTERRUPT ENABLE
689 004146 016700 174626          MOV      PVEC,R0
690 004154 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                   BLO     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                   BLO     PLCP           !IF 0: 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          !RETURN
718 004252 005077 174576          !LINE COUNT PRINT ROUTINE
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

```

719	004256	071103				DIV	R3,R1		!DIVIDE CHARACTER AMOUNT BY LINE FACTOR
720	004260	012704	007673			MOV	MSG43,R4		
721	004264	004767	000514			JSR	PC,TTOUT		!PRINT HEADER
722	004270	004767	000310			JSR	PC,DECPHT		!PRINT LINE AMOUNT IN DECIMAL
723	004274	032777	020000	174546		BIT	020000,08=R		!SEE IF CONTINUOUS MODE
724	004302	001001				RNE	PLCG		!IF 801 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	0100000,08=R		!SEE IF NO PRINT
746	004324	001012				BNE	PCONTL		!IF 801 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 MSG ADDR
751	004346	004767	000432			JSR	PC,TTOUT		!PRINT ERROR
752	004352	032777	040000	174470	PCONTL:	BIT	040000,08=R		!SEE IF HALT ON ERROR
753	004360	001401				BEO	PGOL		!IF NOT: BR
754	004362	000000				HALT			
755	004364	032777	002000	174456	PGOL:	BIT	02000,08=R		!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 010000,05WH ;SEL 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 ;SEL IF ONE OR ZERO
789 004460 001404 BEO 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 ;SEL 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 ;TYPE RECILVED DATA HEADER
800 004522 005203 INC R3 ;SET FLAG
801 004524 010004 MOV R0,R4 ;SET RECIEVED CHAN IN R4
802 004526 000167 177710 JMP PDE1 ;DO RECIEVED
803 004532 012704 007202 PDE5: MOV 0MSG32,R4
804 004536 004767 000242 JSR PC,TTOUT ;TYPE CR/LF
805 004542 032777 040000 174300 PCONTD: BIT 040000,05WH ;SEL IF HALT ON ERROR
806 004550 001401 BEO PGOD ;IF NOT: BR
807 004552 000000 HALT
808 004554 000207 PGOD: RTS PC ;RETURN
809

```

```

011
012
013
014
015
016
017
018
019
020
021
022 004556 032777 020000 174264 PENDI BIT 020000,05WR ;LOOK FOR CONTUOUS MODE
023 004564 001005 BNE PLOUP ;IF SU: 0N
024 004566 012704 006067 MOV 04567,R4
025 004572 004767 000206 JSR PC,ITUUT ;PRINT END OF TEST
026 004576 000000 HALT
027 004606 000167 174374 PLOUP: 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 ON TO
;RESTART WITH NO PRINT OUT DEPENDING
;ON CONSOLE SWITCH 13.
;*****
  
```

```

030
031
032
033
034
035
036
037 004604 012767 177773 000152  DECPNT: MOV      #=5,DIGCNT
038 004612 012767 004772 000150      MOV      @DECPNT+2,DECPNT
039 004620 005067 000142      CLW      ZERO
040 004624 012767 177777 000130  TYPI1:  MOV      #=1,DIGIT
041 004632 005267 000124      TYPI2:  INC      DIGIT
042 004636 167703 000126      SUB      @DECPNT,R3
043 004642 100373      BPL      TYPI2
044 004644 067703 000120      ADD      @DECPNT,R3
045 004650 004767 000020      JSR      PC,DECOLT
046 004654 005267 000104      INC      DIGCNT
047 004660 001001      HNE      TYPI3
048 004662 000207      RIS      PC
049 004664 062767 000002 000076  TYPI3:  ADD      #2,DECPNT
050 004672 000754      BK      TYPI1
051 004674 005767 000062      DECOU1: TST      DIGIT
052 004700 001010      HNE      DEC1
053 004702 022767 177777 000054      CMP      #=1,DIGCNT
054 004710 001404      BEQ      DEC1
055 004712 016767 000050 000042      MOV      ZERO,DIGIT
056 004720 000406      BK      DEC2
057 004722 012767 000260 000036  DEC1:  MOV      #260,ZERO
058 004730 052767 000260 000024      BIS      #260,DIGIT
059 004736 005767 000020      DEC2:  TST      DIGIT
060 004742 001406      BEQ      DEC3
061 004744 105777 174072      TST@    #IP5
062 004750 100372      BPL      DEC2
063 004752 016777 000004 174064      MOV      DIGIT,#IPB
064 004760 000207      DEC3:  RIS      PC
065
066 004762 000000      DIGIT:  0
067 004764 000000      DIGCNT: 0
068 004766 000000      ZERO:  0
069 004770 004772      DECPNT: .+2
070 004772 023420      10000.
071 004774 001750      1000.
072 004776 000144      100.
073 005000 000012      10.
074 005002 000001      1.
075
  
```



```

877
878
879
880
881
882
883
884 005004 112467 174014          TTOUT:  MOV#   (R4)+,TOB      ;TOB = OUTPUT CHAN
885 005010 122767 000043 174006    CMP#   #43,TOB      ;LOOK FOR TERMINATOR
886 005016 001430                    BEQ    TEX          ;EXIT
887 005020 122767 000045 173776    CMP#   #45,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:  MOV#   #15,TOB      ;TOB = CR
892 005044 004767 000014                    JSR    PC,TOG        ;GO TYPE CR
893 005050 112767 000012 173746    MOV#   #12,TOB      ;TOB = LF
894 005056 004767 000002                    JSR    PC,TOG        ;GO TYPE LF
895 005062 000750                    BR     TTOUT         ;CONTINUE
896 005064 105777 173752          TOG:    TST#   #TP#      ;LOOK FOR DONE
897 005070 100375                    RPL    TOG          ;AWAIT FLAG
898 005072 116777 173726 173744    MOV#   TOB,#TP#     ;ECHO CHAN
899 005100 000207          TEX:    RTS     PC          ;EXIT
900
901
902
903
904
905
906
907
908
909 005102 005014          READ:   CLR     #R4          ;CLEAR FOR INPUT
910 005104 004767 000042          READ1: JSR     PC,TTIN         ;GO READ KEYBOARD AND ECHO
911 005110 022767 000215 173710    CMP#   #215,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     #R4          ;ROTATE INPUT
916 005130 005301          DEC     R1          ;DECREMENT POSITION FACTOR
917 005132 001374          BNE    ROT          ;DO ALL ROTATIONS
918 005134 040567 173666          RIC     #5,TIB       ;STRIP ASCII
919 005140 056714 173662          BIS     TIB,#R4      ;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     #TKS      ;
927 005156 005077 173656          CLR     #TKB        ;
928 005162 005067 173640          CLR     TIB         ;
929 005166 005277 173644          INC     #TKS        ;SET READER GO BIT
930 005172 105777 173640          TTIN1: TST#   #TKS        ;AWAIT DONE

```

LPC11b.SFC

931	005176	100375				RPL	TTIN1	
932	005200	017767	173634	173620		MOV	@TKB, IIB	ITIB = INPUT CHAR
933	005206	105777	173630		TTIN2:	TSTM	@IP6	!AWAIT PUNCH READY
934	005212	100375				BPL	TTIN2	
935	005214	116777	173606	173622		MOVW	TIB, @IP6	!ECHO
936	005222	000207				RIS	PC	!EXIT
937								

```

939
940
941
942
943
944
945
946
947
948 005224 032626          PINT8: BIT      (SP)+,(SP)+      ;RESET STACK POINTER
949 005226 004767 000074    JSR      PC,PRSTV      ;RESET VECTORS
950 005232 000177 173574    JMP      @PINT8        ;RETURN
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965 005236 022626          PINT9: CMP      (SP)+,(SP)+      ;RESET STACK POINTER
966 005240 005077 173614    CLR      @PTS          ;RESET INTERRUPT ENABLE IN TESTER
967 005244 005077 173604    CLR      @PCS          ;RESET CONT INTERRUPT ENABLE
968 005250 024040          CMP      =(R0),=(R0)    ;POINT TO PROPER VECTOR
969 005252 012720 005316    MOV      @PINTL,(R0)+  ;SET TO ERROR VECTOR
970 005256 012720 000340    MOV      @340,(R0)+
971 005262 005077 173564    CLR      @PSW          ;SET PSW TO LOWEST PRIORITY
972 005266 005067 173522    CLR      PTIM1         ;SET TIME
973 005272 005367 173516          PINT10: DEC      PTIM1
974 005276 001375          BNE      PINT1
975 005300 012777 000340 173544    MOV      @340,@PSW     ;RESET PSW
976 005306 004767 000014          JSR      PC,PRSTV      ;RESET VECTORS
977 005312 000177 173514          JMP      @PINT10       ;RETURN
978 005316 012704 006105          PINT11: MOV      @MSG10,R4
979 005322 000167 176770          JMP      PLE           ;GO TO ERROR ROUTINE
980

```

```

962
963
964
965
966
967
968
969
970
971 005326 016700 173446 PRSIVI MOV PVEC,R0 ;RO = CONT VECTOR
972 005332 010001 MOV R0,R1
973 005334 005721 TST (R1)+ ;POINT TO CONT MALT LOC
974 005336 010120 MOV R1,(R0)+ ;SET CONT VECTOR ADDR
975 005340 005020 CLR (R0)+ ;SET CONT MALT
976 005342 022121 CMP (R1)+,(R1)+ ;POINT TO TESTER MALT
977 005344 010120 MOV R1,(R0)+ ;SET TESTER VECTOR ADDR
978 005346 005020 CLR (R0)+ ;SET TESTER MALT
979 005350 000207 RTS PC ;RETURN
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010 005352 010046 PWRFAIL 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 PWRUP: MOV @340,@PSP
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  
1037 005434 012601  
1038 005436 012600  
1039 005440 005005  
1040 005442 005305  
1041 005444 001376  
1042 005446 012704 007536  
1043 005452 004767 177326  
1044 005456 005305  
1045 005460 001376  
1046 005462 000167 173520  
1047

MOV (SP)+,R2  
MOV (SP)+,R1  
MOV (SP)+,R0  
CLR R5  
DEC R5  
BNE .-2  
MOV R5,R4  
JSR PC,TTUUT  
DEC R5  
BNE .-2  
JMP PHEG

ITYPE POWER FAIL MESSAGE

```

1049
1050
1051
1052
1053
1054
1055 005466      000      377      376  PPAT5:  .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          PPAT6:  .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      049  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
  
```

	005744	052040	054522	040440		
	005752	040507	047111	021445		
1070						
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 /%INTERHUPT 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	024504		
	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		

	006750	051105	051040	040505		
	006764	054504	021445			
1106						
1107	006770	047045	020117	047111	MSG25:	.ASCII /NO INTERRUPT FROM CONTROLLED 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	047522	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 /TESTED 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 /TESTED INTERRUPTS AT HIGHER THAN LEVEL 4/

	007330	051105	044440	052110		
	007336	051105	052522	052170		
	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 ENABLES/
	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				
1126						
1129	007457	045	047503	052116	MSG37:	.ASCII /%CONTROLLER INTERRUPTS AT HIGHER THAN LEVEL 400/
	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			
1136						
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	010029	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 /SERFOR BIT OK##/
	010202	041040	052111	047440	
	010210	022513	043		
1147		000001			.END

DECLUT	004674	845	851a																	
DECLFNT	004770	838a	842	844	849a	869a														
DECLFRT	004604	722	837a																	
DEC1	004722	852	854	857a																
DEC2	004736	856	859a	862																
DEC3	004760	860	864a																	
DIGCNT	004764	837a	846a	853	867a															
DIG11	004767	840a	841a	851	855a	858a	859	863	866a											
DINT	004237	691	710a																	
LINE	004312	687	728a																	
LRS	001046	124a	684a	695	697a	710a														
MSG0	005512	747	1065a																	
MSG1	005535	168	1067a																	
MSG10	006105	978	1082a																	
MSG11	006131	505	1084a																	
MSG12	006241	269	1087a																	
MSG13	006276	261	1089a																	
MSG14	006337	294	1091a																	
MSG15	006411	305	1093a																	
MSG16	006460	317	1095a																	
MSG17	006510	325	1097a																	
MSG2	005564	170	1069a																	
MSG20	006546	336	1099a																	
MSG21	006611	348	1101a																	
MSG22	006660	358	1103a																	
MSG24	006727	380	1105a																	
MSG25	006770	405	1107a																	
MSG26	007035	522	1109a																	
MSG27	007123	779	1112a																	
MSG3	005624	185	1071a																	
MSG30	007140	741	1114a																	
MSG31	007161	798	1116a																	
MSG32	007202	803	1118a																	
MSG33	007205	586	1121a																	
MSG34	007246	431	1123a																	
MSG35	007323	450	1125a																	
MSG36	007376	473	1127a																	
MSG37	007457	489	1129a																	
MSG4	005660	191	1073a																	
MSG40	007536	1042	1131a																	
MSG41	007573	277	1133a																	
MSG42	007632	285	1135a																	
MSG43	007673	720	1137a																	
MSG44	007720	500	1139a																	
MSG45	010061	503	1143a																	
MSG46	010174	509	1146a																	
MSG5	005722	204	1075a																	
MSG6	005760	216	507	1077a																
MSG7	006067	824	1080a																	
PHEG	001206	89	150a	1046																
PC	MSG00007	51a	169a	174a	179a	186a	192a	197a	205a	217a	222a	258a	392a	495a						
		501a	504a	506a	508a	510a	512a	517a	524a	569a	587a	592a	632a	731a						
		722a	749a	751a	780a	782a	799a	804a	808a	825a	845a	848a	864a	889a						
		892a	894a	899a	910a	922a	936a	949a	976a	999a	1043a									

LPC11-5PC CROSS REFERENCE TABLE

PCP	001056	1200	2910	3420	3750	4240	4450	5590	5800	5460	5400	704		
PCHAM	001022	1110	5550	5580	559	567	5790	560	590	5930	594			
PCLF	001244	1000	102											
PCOMTL	004542	770	8050											
PCOMTL	004392	740	7520											
PCB	001054	1270	208	2540	259	275	292	334	350	3650	3900	4110	4390	4600
		4810	4850	510	5520	6310	6380	6630	7180	9670				
PDA1	003374	498	511	5500										
PDA10	003400	5510	571	574										
PDA11	003440	5590	573											
PDA110	003620	591	5930											
PDA111	003660	597	6000											
PDA17	003452	5010	504											
PDA12A	003474	502	5660											
PDA13	003517	508	5700											
PDA14	003532	557	5700											
PDA15	003540	5790	599											
PDA16	003544	5800	595											
PDA17	003550	5820	505											
PDA17A	003574	505	5060											
PDA17H	003610	503	5090											
PDE	004404	569	592	7770										
PDE1	004442	7690	802											
PDE2	004446	7860	707	795										
PDE3	004472	769	7920											
PDE4	004500	791	7930											
PDE5	004532	797	8030											
PEB1	003176	483	4940											
PEB10	003224	497	4990											
PEB11	003312	501	504	506	508	5120								
PEB12	003344	5180	757											
PEB13	003372	521	5240											
PEND	004556	588	600	8220										
PET	001034	1100	150	4990	5020	520								
PGO	004400	7580												
PGOD	004554	806	8080											
PGOF	001012	1070	210	513										
PGOL	004364	753	7550	756										
PIC	001020	1100	5510	5720	5780	5980								
PINI	005236	300	394	9650										
PINTE	005316	969	9780											
PINIF	001032	1190	3730	3980	4210	4400	4600	6450	950	977				
PINIS	005224	419	460	636	643	9480								
PINT1	005272	9730	974											
PITA	001010	1060	4140	430	4650									
PLC	004112	234	6820											
PLCC	001570	227	2320	726										
PLCG	004306	724	7260											
PLCF	004242	705	7160											
PLC1	004174	6950	696											
PLC2	004212	6980	700											
PLF	004316	262	270	278	286	295	306	318	326	337	349	359	381	400
		432	451	474	490	523	7450	979						
PLOOF	004600	823	8270											

PPATE	005510	556	1057e												
PPA1S	005466	550	1055e												
PRFG	001002	103e	193	198	200	202	207								
PRGF	001444	199	201	204e											
PRG	001356	183	191e												
PRG1	001366	193e	206												
PRIV	005326	258	392	495	632	949	976	991e							
P&COP	0036e4	226	625e	628											
P&COP0	003672	627e	657	660	663										
P&COP1	003756	634	640e												
P&COP2	004002	639	645e												
P&COP3	004036	651e	652	654											
P&COP4	004052	648	655e												
P&COP5	004070	659e	662												
PSFT	001456	203	207e												
PSK1V	001200	92	148e	758	827										
PSMC	001954	164	226e												
PSPS	001006	105e	1017e	1031											
PSTAF1	001210	149	151e												
PSW	001052	126e	152e	374e	399e	422e	429e	447e	469e	629e	650e	694e	971e	975e	
		1030e													
PTH	001062	130e	437e	458e	479e	554e	566	589							
PTI*1	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									
PTI*2	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												
PT1A1r	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											
PT1rA	002422	373	385e												
PT1rP	002440	388e	369												
PT1rC	002446	390e	391												
PT1G	002526	401e	402	404											
PT1H	002552	398	410e												
PT1HA	002614	419e													

PT1HR	002624	4210												
PT1J	002646	4250	426	449	472									
PT1K	002654	4270	428											
PT1L	002674	415	4310											
PT1M	002704	414	4360											
PT1M1	002734	4430	444											
PT1N	002774	440	4500											
PT1F	003004	441	4570											
PT1G	003060	4680	488											
PT1H	003104	464	4730											
PT1B	003114	465	4780											
PT1T	003166	487	4890											
PVE	001344	181	1850											
PVEC	001000	1020	175	180	182	366	393	417	462	635	641	689	991	
PVJ	001276	1750	187											
PWRPAL	005352	64	10100											
PWRUP	005406	1010	10300											
READ	005102	179	197	222	517	9090								
READ1	005104	9100	921											
REX	005150	912	9220											
ROT	005124	9140	917											
ROTA	001030	1140	1770	1950	2200	5150	913							
RO	0000000	440	1540	1550	1580	1590	1600	2070	2120	3660	367	3680	3690	3930
		3940	3950	4170	418	4190	4200	4620	4660	4670	5180	5190	520	5660
		567	5890	590	6350	6360	6370	6410	642	6430	6440	6890	690	6910
		6920	6930	698	7040	801	968	9690	9700	9910	992	9940	9950	9970
		9980	1010	10380										
R1	0000001	450	1570	1610	2080	2120	5500	556	558	6870	7190	9130	9160	9920
		993	994	996	997	1011	10370							
R2	0000002	460	1480	1500	163	6250	626	627	1012	10360				
R3	0000003	470	1760	1940	2190	2610	2690	2770	2850	2940	3050	3170	3250	3360
		3480	3580	3800	4050	4160	422	4310	4500	4630	469	4730	4860	4890
		5140	5220	6260	648	6550	6850	7100	719	750	7830	796	8000	8420
		8440	9200	1013	10350									
R4	0000004	480	1680	1700	1750	1850	1910	1930	2040	2160	2180	4150	421	4640
		468	4870	5000	5030	5050	5070	5090	5130	5860	7200	7470	7480	7800
		7790	7810	7840	788	7930	7980	8010	8030	8240	884	9090	9150	9190
		9780	1014	10340	10420									
R5	0000005	490	1780	1960	2410	4130	440	4420	4610	482	4840	5160	7850	7940
		918	1015	10330	10390	10400	10440							
SP	0000006	500	1510	948	965	10100	10110	10120	10130	10140	10150	10160	1017	10310
		1032	1033	1034	1035	1036	1037	1038						
S=R	001050	1250	226	232	496	570	596	625	627	633	656	723	745	782
		755	777	805	822									
TCRLF	005036	888	8910											
TEX	005100	886	8990											
TIB	001026	1130	911	9180	919	9280	9320	935						
TIML	004314	693	7290											
TINT	004224	70	7040											
TKH	001040	1210	9270	932										
TAS	001036	1200	9260	9290	930									
TOR	001024	1120	8840	888	887	8910	8930	898						
TOG	005064	889	842	894	8960	897								
TPB	001044	1230	7900	7920	8630	8980	9350							



TP8	001042	1220	706	061	096	933								
TTIA	005152	910	9260											
TTIN1	005172	9300	931											
TTIN2	005206	9330	934											
TTOUT	005004	169	174	186	192	205	217	510	512	507	721	749	751	780
		782	799	804	825	8640	890	895	1043					
TYPI1	004024	8400	850											
TYPT2	004032	8410	843											
TYPI3	004004	847	8490											
ZERI	004766	8390	855	8570	8600									
.	010213	550	59	630	690	000	910	900	1330	156	869	1041	1045	10500

ADD	212	844	849												
BEO	227	233	260	268	276	284	293	316	357	521	557	568	591	595	657
	696	705	753	756	789	806	854	860	866	888	912				
RHI	183	201	203												
HIC	519	918													
BIS	370	396	446	485	638	640	688	697	958	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	895	723	745	752	755	777
	805	822	948												
RITE	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	463	497	562	564	571	573
	583	585	597	599	628	634	652	654	660	662	724	746	778	795	797
	623	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	763	839	909	926	927
	926	966	967	971	972	995	998	1039							
CMP	162	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	388
	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
	476	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
	764	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	1018
	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														

LPC11F.SRC CROSS REFERENCE TABLE

TST	163	440	462	796	851	859	993								
TSTe	766	861	896	930	933										
WAIT	699														
.AHS	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													
.HEPT	56	209													
.TITLE	2														

ERRORS DETECTED: 0

LPC11 MAC11.024 20-JUN-73 14132 PAGE 10-13  
LPC11B.SRC

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