

•REM

IDENTIFICATION

PRODUCT CODE: AC-E908C-MC
PRODUCT NAME: CXLPECC LPD-11 MODULE
PRODUCT DATE: SEPTEMBER 1978
MAINTAINER: DEC/X11 SUPPORT GROUP

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE OR EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1975,1978 DIGITAL EQUIPMENT CORPORATION

MAIN DEC CHANGE NOTICE
MAY BE REQUIRED FOR
PROGRAM TO OPERATE

1. ABSTRACT

LPE IS A LOMOD THAT EXERCISES UP TO EIGHT LPD11 PHOTO-COMP INTERFACES. BY USING THE BUILT IN TESTER LOGIC, IT PERMITS AND CHECKS ALL POSSIBLE DATA PATTERNS IN ADDITION TO STATUS CHECKING ALL LPD11'S SELECTED FOR TEST (UP TO 8 LPD'S WITH CONTIGUOUS ADDRESSES AND VECTORS) ARE ACTIVATED AND RUN CONCURRENTLY. ALL STATUS AND DATA ERRORS ARE REPORTED ON THE CONSOLE TTY.

2. REQUIREMENTS

HARDWARE: UP TO EIGHT LPD11 INTERFACE LOGICS

STORAGE:: LPE REQUIRES:

- 1. DECIMAL WORDS: 560
- 2. OCTAL WORDS: 1060
- 3. OCTAL BYTES: 2140

3. PASS DEFINITION

ONE PASS OF THE LPE MODULE CONSISTS OF CONTINUOUSLY SENDING AND CHECKING A COMPLEMENTING BINARY COUNT PATTERN FOR THE TIME PERIOD DEFINED BELOW.

4. EXECUTION TIME

LPE OPERATING BY ITSELF WILL TAKE AN AVERAGE OF ONE MINUTE TO COMPLETE ONE PASS.

5. CONFIGURATION REQUIREMENTS

DEFAULT PARAMETERS:

DEVADR: 172710, VECTOR: 320, RR1: 4

REQUIRED PARAMETERS:

AT CONFIGURATION TIME THE USER MUST SPECIFY:

VCT: VECTOR ADDRESS OF FIRST LPD IF NOT 320
OVC: NO. OF LPD'S IF GREATER THAN 1

6. DEVICE/OPTIONS SETUP

PLACE THE LPD LOGIC(S) IN THE LOCAL TEST MODE. THIS IS ACCOMPLISHED BY SETTING SWITCH 9 OF SWITCH GROUP 2 ON THE 4523 MODULE TO THE "ON" POSITION.

7. MODULF OPERATION

7.1 TEST SEQUENCE

- A. START: THIS CODE SETS UP THE BASE ADDRESSES DEFINED BY 'ADDR'.
- B. SETVEC: THIS CODE CHECKS 'DVID1' FOR NON-ZERO SELECTION, AND SETS UP THE VECTORS FOR SELECTED LINES WITH INT SERVICE ADDRESSES AND BR LEVELS.
- C. SELECT: THIS CODE COUNTS THE SELECTED LPD'S AND SETS UP THE PROGRAM STATUS TARLF 'LPDSTS.'
- D. RESTRPT: THIS CODE CLEARS TABLES, INITIALIZES FLAGS & TIMERS AND THEN STARTS OR RESTARTS ALL SELECTED LPD'S BY SENDING DATA PARALING THE TESTER INTERRUPT ENABLE.
- E. TIMER: THIS IS AN 'END PASS' TIMER LOOP VIA 'BREAKS' TO THE MONITOR. FLAG 'EOP' IS SET WHEN THIS LOOP EXPIRES (ABOUT 1 MINUTE).
- F. TMOUT: THIS TIMER ALLOWS ENOUGH TIME FOR ALL LPD'S TO COMPLETE THE LAST DATA TRANSFER. IF ANY LPD IS STILL ACTIVE IT IS REPORTED AND THEN DROPPED. IN THE EVENT THAT ALL SELECTED LPD'S FAIL TO INTERRUPT THEN THE MODULE WILL RE DROPPED.
- G. TSPSRV: THE TESTER SERVICE ROUTINE SIMPLY QUEUES UP THE REQUEST FOR SERVICE IN A FIFO QUEUE. UPDATES THE POINTER AND RETURNS CONTROL BACK TO THE MONITOR WITH A 'PIRG' THE ELEMENT THAT GETS STORED IN THE QUEUE IS THE NUMBER OF THE INTERRUPTING TESTER. ACTUAL SERVICE IS DONE LATER WHERE THE 'SERVICE' IS EXPCTED AT LEVEL 7PRO.

H. TSERV:

THIS CODE RETRIEVES THE TESTER NO. THIS CHECKS STATUS AND DATA INFORMATION FOR THIS TESTER, REPORTS ERRORS, ENABLES THE CORRESPONDING LPD INTERRUPT ENABLE, THEN DOES AN EXIT BACK TO THE MONITOR.

I. LSRV:

THE LPD SERVICE ROUTINE SIMPLY QUEUES UP THE REQUEST FOR SERVICE IN A FIFO QUEUE. UPDATES THE POINTER AND RETURNS CONTROL BACK TO THE MONITOR WITH A 'PIRG' THE ELEMENT THAT GETS STORED IN THE QUEUE IS THE LPD NO. OF THE INTERRUPTING LPD. ACTUAL SERVICING IS DONE LATER WHERE THE SERVICING CODE IS EXECUTED AT LEVEL ZERO.

J. LPDSRV:

THIS CODE RETRIEVES THE LPD NO., CHECKS STATUS AND DATA INFORMATION FOR THIS LPD, REPORTS ERRORS AND GOES TO ROUTINE 'PASSCK'.

K. PASSCK:

THIS CODE DETERMINES IF THE CURRENT INTERRUPTING LPD SHOULD BE RESTARTED WITH THE NEXT DATA PATTERN. IF NOT IT IS TIME FOR 'END PASS' MESSAGE.

L. SENDAT:

THIS CODE SENDS DATA AND ENABLES THE TESTER INTERRUPT ENABLE FOR ANOTHER TESTER/LPD INTERRUPT SEQUENCE.

7.2 DESCRIPTION OF TABLES AND QUEUES

A. DATAR:

8 WORD DATA TABLE - CONTAINS THE CURRENT DATA WORDS SENT TO THE LPD TESTER LOGIC.

B. INTQ:

8 WORD INTERRUPT SERVICE FIFO QUEUE - CONTAINS LPD NUMBERS REQUIRING SERVICE.

C. LPDSTS:

8 BYTE TABLE WHICH SPECIFIES WHAT EACH LPD IS DOING: NOT SELECTED (0), TESTER/LPD DATA TRANSFER SEQUENCE IN PROGRESS (377), TESTER/LPD DATA TRANSFER SEQUENCE COMPLETED (177) - THIS CONDITION IS SET WHEN THE 'END PASS' MESSAGE IS DUE.

R. OPERATION OPTIONS

- A. THE USER CAN MODIFY "DVID1" TO SELECT OR DESELECT INDIVIDUAL LPD1'S.
- B. THE USER CAN USE THE "MOD" COMMAND TO DUMP THE TABLES AND QUEUES DESCRIBED IN 7.2 TO OBTAIN MORE DETAILED INFORMATION.

9. NON STANDARD PRINTOUTS

- A. "NO LPD(S) SELECTED" IS TYPED IF THE USER SETS "DVID1" TO ZERO IN ERROR.
- B. IF AN LPD FAILS TO COMPLETE A TESTER/LPD DATA TRANSFER IN THE TIME ALLOTTED BY ROUTINE "TMOU1", IT IS CONSIDERED HUNG. THE MESSAGE "LPD# N WAS DROPPED" IS TYPED WHERE N IS THE LPD NUMBER ACCORDING TO IT'S BUS ADDRESS.

NOTE: THE ABOVE PRINTOUT WILL OCCUR IF THE OPERATOR FAILS TO COMPLY WITH SECTION 6 OF THIS DOCUMENT.

```

202
203 000000
204 000000
205
206
207
208
209
210 000000
211 000000 050114 041505 040
212 000005 000
213 000006 172710
214 000010 004328
215 000012 00300
216 000013 000
217 000014 000001
218 000016 000000
219 000020 000000
220 000022 000000
221 000024 000000
222 000026 140000
223 000030 000320
224 000032 000320
225 000034 000000
226 000036 000010
227 000040 000000
228 000042 000000
229 000044 000000
230 000046 000000
231 000050 000000
232 000052 000000
233 000054 000000
234 000056 000000
235 000058 000000
236 000060 000000
237 000062 000000
238 000064 000000
239 000066 000000
240 000070 000000
241 000072 000000
242 000074 000000
243 000076 000000
244 000100 000000
245 000102 000000
246 000104 000000
247 000106 000000
248 000108 000000
249 000110 000000
250 000112 000546
251 000114 000000
252 000116 000000

```

```

IOMOD <LPEC> 172710 320 4 556 10 63
MODULE 140000 1822 172710 320 4 556 10 63
TITLE LPEC DFC/X11 SYSTEM EXERCISEP MODULE
DDXCOM VERSION 6 23-MAY-78
LIST BIN
*****
BEGIN:
MODNAM: .ASCII (/LPEC /)MODULE NAME
XPLAG: .BYTE OPEN ;USED TO KEEP TRACK OF WBUFF USAGE
ADDR: 172710+0 ;1ST DEVICE ADDR
VECTOR: 320+0 ;1ST DEVICE VECTOR.
BR1: .BYTE PRTY4+0 ;1ST BR LEVEL.
BR2: .BYTE PRTY+0 ;2ND BR LEVEL.
DVT01: .I ;DEVICE INDICATOR 1.
SR1: OPEN ;SWITCH REGISTER 1.
SR2: OPEN ;SWITCH REGISTER 2.
SR3: OPEN ;SWITCH REGISTER 3.
SR4: OPEN ;SWITCH REGISTER 4.
*****
STAT: 140000 ;STATUS WORD.
INIT: START ;MODULE START ADDR.
SPDINT: MODSP ;MODULE STACK POINTER.
PASCNT: 0 ;PASS COUNTER.
ICOUNT: 10 ;# OF ITERATIONS PER PASS=10
SOPCNT: 0 ;LOC TO COUNT ITERATIONS
HRDCNT: 0 ;LOC TO SAVE TOTAL SOFT ERRORS
SOPPAS: 0 ;LOC TO SAVE TOTAL HARD ERRORS
HRDPAS: 0 ;LOC TO SAVE SOFT ERRORS PER PASS
SYSCNT: 0 ;LOC TO SAVE HARD ERRORS PER PASS
RANNUM: C ;# OF SYS ERRORS ACCUMULATED
CONFIG: ;RESERVED FOR MONITOR USE WHEN RAND MACRO IS CALLED
RES1: C ;RESERVED FOR MONITOR USE
RES2: C ;RESERVED FOR MONITOR USE
RES3: C ;RESERVED FOR MONITOR USE
RES4: C ;RESERVED FOR MONITOR USE
RES5: C ;RESERVED FOR MONITOR USE
RES6: C ;RESERVED FOR MONITOR USE
RES7: C ;RESERVED FOR MONITOR USE
RES8: C ;RESERVED FOR MONITOR USE
RES9: C ;RESERVED FOR MONITOR USE
RES10: C ;RESERVED FOR MONITOR USE
RES11: C ;RESERVED FOR MONITOR USE
RES12: C ;RESERVED FOR MONITOR USE
RES13: C ;RESERVED FOR MONITOR USE
RES14: C ;RESERVED FOR MONITOR USE
RES15: C ;RESERVED FOR MONITOR USE
RES16: C ;RESERVED FOR MONITOR USE
RES17: C ;RESERVED FOR MONITOR USE
RES18: C ;RESERVED FOR MONITOR USE
RES19: C ;RESERVED FOR MONITOR USE
RES20: C ;RESERVED FOR MONITOR USE
RES21: C ;RESERVED FOR MONITOR USE
RES22: C ;RESERVED FOR MONITOR USE
RES23: C ;RESERVED FOR MONITOR USE
RES24: C ;RESERVED FOR MONITOR USE
RES25: C ;RESERVED FOR MONITOR USE
RES26: C ;RESERVED FOR MONITOR USE
RES27: C ;RESERVED FOR MONITOR USE
RES28: C ;RESERVED FOR MONITOR USE
RES29: C ;RESERVED FOR MONITOR USE
RES30: C ;RESERVED FOR MONITOR USE
RES31: C ;RESERVED FOR MONITOR USE
RES32: C ;RESERVED FOR MONITOR USE
RES33: C ;RESERVED FOR MONITOR USE
RES34: C ;RESERVED FOR MONITOR USE
RES35: C ;RESERVED FOR MONITOR USE
RES36: C ;RESERVED FOR MONITOR USE
RES37: C ;RESERVED FOR MONITOR USE
RES38: C ;RESERVED FOR MONITOR USE
RES39: C ;RESERVED FOR MONITOR USE
RES40: C ;RESERVED FOR MONITOR USE
RES41: C ;RESERVED FOR MONITOR USE
RES42: C ;RESERVED FOR MONITOR USE
RES43: C ;RESERVED FOR MONITOR USE
RES44: C ;RESERVED FOR MONITOR USE
RES45: C ;RESERVED FOR MONITOR USE
RES46: C ;RESERVED FOR MONITOR USE
RES47: C ;RESERVED FOR MONITOR USE
RES48: C ;RESERVED FOR MONITOR USE
RES49: C ;RESERVED FOR MONITOR USE
RES50: C ;RESERVED FOR MONITOR USE
RES51: C ;RESERVED FOR MONITOR USE
RES52: C ;RESERVED FOR MONITOR USE
RES53: C ;RESERVED FOR MONITOR USE
RES54: C ;RESERVED FOR MONITOR USE
RES55: C ;RESERVED FOR MONITOR USE
RES56: C ;RESERVED FOR MONITOR USE
RES57: C ;RESERVED FOR MONITOR USE
RES58: C ;RESERVED FOR MONITOR USE
RES59: C ;RESERVED FOR MONITOR USE
RES60: C ;RESERVED FOR MONITOR USE
RES61: C ;RESERVED FOR MONITOR USE
RES62: C ;RESERVED FOR MONITOR USE
RES63: C ;RESERVED FOR MONITOR USE
RES64: C ;RESERVED FOR MONITOR USE
RES65: C ;RESERVED FOR MONITOR USE
RES66: C ;RESERVED FOR MONITOR USE
RES67: C ;RESERVED FOR MONITOR USE
RES68: C ;RESERVED FOR MONITOR USE
RES69: C ;RESERVED FOR MONITOR USE
RES70: C ;RESERVED FOR MONITOR USE
RES71: C ;RESERVED FOR MONITOR USE
RES72: C ;RESERVED FOR MONITOR USE
RES73: C ;RESERVED FOR MONITOR USE
RES74: C ;RESERVED FOR MONITOR USE
RES75: C ;RESERVED FOR MONITOR USE
RES76: C ;RESERVED FOR MONITOR USE
RES77: C ;RESERVED FOR MONITOR USE
RES78: C ;RESERVED FOR MONITOR USE
RES79: C ;RESERVED FOR MONITOR USE
RES80: C ;RESERVED FOR MONITOR USE
RES81: C ;RESERVED FOR MONITOR USE
RES82: C ;RESERVED FOR MONITOR USE
RES83: C ;RESERVED FOR MONITOR USE
RES84: C ;RESERVED FOR MONITOR USE
RES85: C ;RESERVED FOR MONITOR USE
RES86: C ;RESERVED FOR MONITOR USE
RES87: C ;RESERVED FOR MONITOR USE
RES88: C ;RESERVED FOR MONITOR USE
RES89: C ;RESERVED FOR MONITOR USE
RES90: C ;RESERVED FOR MONITOR USE
RES91: C ;RESERVED FOR MONITOR USE
RES92: C ;RESERVED FOR MONITOR USE
RES93: C ;RESERVED FOR MONITOR USE
RES94: C ;RESERVED FOR MONITOR USE
RES95: C ;RESERVED FOR MONITOR USE
RES96: C ;RESERVED FOR MONITOR USE
RES97: C ;RESERVED FOR MONITOR USE
RES98: C ;RESERVED FOR MONITOR USE
RES99: C ;RESERVED FOR MONITOR USE
RES100: C ;RESERVED FOR MONITOR USE

```

```

255 000120 000000
256 000122 000063
257 000124 000040
258
259
260
261
262
263

```

```

INTR: OPEN ;# OF INTERRUPTS PER ITERATION
IDNUM: 63 ;MODULE IDENTIFICATION NUMBER=63
      .REPT SPSIZ ;MODULE STACK STARTS HERE.
      .WLIST
      .WORD 0
      .LIST
      .ENDR
MODSP:
*****

```

264
265
266
267 000234 000000
268 000230 000000
269 000232 000000
270 000234 000000
271 000236 000000
272 000238 000000
273 000240 000000
274 000242 000000
275 000244 000000
276 000246 000000
277
278
279
280 000250 000010
281 000270 000010
282 000310 000004
283
284
285
286
287
288 000320 012767 000011 177572
289 000324 012767 000001 177560
290 000328 012767 000001 177554
291 000332 012767 177441
292 000336 010567 177452
293 000340 010567
294 000344 010567 177444
295 000348 010567 177644
296 000352 010567
297 000356 010567 177636
298 000360 010567 177636
299
300
301
302
303
304 000374 011705 177410
305 000378 012764 001111
306 000382 012764 001636
307 000386 012764 177480
308 000390 011005
309 000394 012443 000001 002056
310 000398 014110 000001
311 000402 013443
312 000406 013443
313 000410 013443 000010
314 000414 013443
315 000418 013443 177342
316 000422 013443
317 000426 013443 177332
318 000430 013443
319 000434 013443 177332

); SOME POINTERS, VARIABLES AND CONSTANTS UNIQUE TO THIS MODULE
LPDCSR: 0 ;BASE BUS ADRS OF LPD CSR
LPDDR: 0 ;BASE BUS ADRS OF LPD DBR
TSRCSR: 0 ;BASE BUS ADRS OF TESTER CSR
TSRDBR: 0 ;BASE BUS ADRS OF TESTER DBR
ACTDEV: 0 ;NO. OF LPDS ACTIVE
QPTR1: OPEN ;MULTI INT FIFO QUEUE POINTER - LOAD
QPTR2: OPEN ;MULTI INT FIFO QUEUE POINTER - UNLOAD
CNR: 0 ;COUNTER FOR END PASS & TIMEOUT TUNING
EOP: 0 ;SET TO NON-ZERO WHEN DUE FOR END PASS
TIMOFF: 0 ;NON-ZERO TURNS OFF END PASS TIMER DURING ER PRINTOUTS

); DATA, ERROR & INTERRUPT FIFO TABLES
DATAB: .BLKW 8. ;8 WORD DATA TABLE
INTQ: .BLKW 8. ;8 WORD INTERRUPT SERVICE QUEUE (FIFO)
LPDSTS: .BLKW 4. ;8 BYTE LPD STATUS TABLE

); THIS CODE SETS UP THE BASE ADDRESSES FOR
;THE LPD & LPD TESTER DEVICES

START: MOV #1,INTR ;1 INTERRUPT/ITERATION
MOV #1,WDT0 ;1 WORD TO MEM/ITERATION
MOV #0,R5 ;1 WORD FROM MEM/ITERATION
MOV #0,R5 ;1 WORD FROM MEM/ITERATION
MOV R5,LPDCSR ;SET UP BASE DEVICE ADDRESSES
TST (R5)+ ;POINT TO LPD DBR - 1ST ACCESS TO
MOV R5,LPDDR ;DEVICE - IF BUS ER TRAP - CK LPD BUS ADRS
TST (R5)+ ;POINT TO TESTER CSR
MOV R5,TSRCSR ;POINT TO TESTER DBR
TST (R5)+
MOV R5,TSRDBR

); THIS CODE CHECKS TO SEE THAT AT LEAST ONE LPD IS
;SELECTED AND THEN SETS UP THE VECTORS FOR THOSE
;WHICH ARE SELECTED

SETVEC: MOV VECTOR,R5 ;SET UP INITIAL VECTOR ADRS
MOV #LSRV0,R4 ;GET 1ST LPD SERVICE ADRS
MOV #TSRV0,R3 ;GET 1ST TESTER SERVICE ADRS
MOV# DVIND1,P1 ;GET CONFIGURATION
BNE 1S ;BRR IF SOMETHING SELECTED
MSGNS,REGIN,M*BL0 ;ASCII MESSAGE CALL WITH COMMON HEADER
ENDS,REGIN ;DRAW MESSAGE - NOTHING SELECTED
1S: ASR R1 ;SHIFT SELECT BIT INTO "C"
BCS 2S ;BRR IF LPD SELECTED
ADD #10,R5 ;ADVANCE VECTOR ADRS TO NEXT LPD
BR 2S ;GO PREPARE FOR NEXT LPD
2S: MOV #34,(R5)+ ;SET VECTOR - LPD
MOVB BR1,(R5)+ ;SET BR - LPD
INC R5 ;MOVE POINTER
MOV #3,(R5)+ ;SET VECTOR - LPD TESTER
MOVB BR1,(R5)+ ;SET BR - TESTER

320 000460 005205
321 000462 022704 000006
322 000466 022703 000006
323 000470 022704 001175
324 000474 001354
325
326
327
328 000500 005067 177530
329 000504 012767 177324
330 000510 005061
331 000512 006200
332 000514 003006
333 000516 012767
334 000520 012767 177512
335 000522 003377 000310
336 000524 006402
337 000528 005061 000310
338 000530 005061
339 000532 005061 000010
340 000534 022701
341 000536 001362
342
343
344
345 000540 012764 000250
346 000544 022704 000270
347 000548 001374
348 000552 005067 177460
349 000556 005067 177452
350 000560 005067 177444
351 000564 002270 177432
352 000568 012767 000270 177426
353 000572 016700 177412
354 000576 005061
355 000580 016702 177402
356 000584 005061
357 000588 005061 000310
358 000592 001474
359 000596 001474
360 000600 000377 000310
361 000604 012767 000250
362 000608 012767 00101
363 000612 005061
364 000616 005061 000010
365 000620 005061
366 000624 005061 000010
367 000628 005061 000010
368 000632 005061 000010

3S: INC R5 ;MOVE POINTER
ADD #6,R4 ;ADVANCE LPD SERVICE ADRS
ADD #6,R3 ;ADVANCE TESTER SERVICE ADRS
CMP #TSRSRV,R4 ;SEE IF 8 LPDS ARE LOOKED FOR
BNE 1S ;BRR IF NOT

); THIS CODE COUNTS ALL SELECTED LPDS AND SETS UP
;THE BYTE STATUS TABLE LPDSTS

SELECT: CLP ACTDEV ;ZERO THE ACTIVE DEVICE COUNT
CLR R1 ;GET SELECTION PARAMETER
CLR R0 ;ZERO LPDSTS OFFSET
1S: ASR R0 ;SHIFT SELECTION PARAMETER INTO "C"
BCC 2S ;BRR IF NOT SELECTED
INC ACTDEV ;RECORD LPD
MOVB #377,LPDSTS(R1) ;#377 MEANS THIS LPD ACTIVE
BR 2S ;GO SEE IF MORE LPDS
2S: CLR BR ;INDICATE THIS LPD INACTIVE
3S: INC R1 ;ADVANCE OFFSET
CMP #10,R1 ;HAVE WE LOOKED AT ALL 8 LPDS?
BNE 1S ;BRR IF NOT

); THIS CODE CLEARS TABLES, INITIALIZES FLAGS & TIMERS
;AND THEN STARTS OR RESTARTS ALL LPD TESTERS

RESTR: MOV #DATAB,R4 ;SET UP ADRS OF DATA TABLE
1S: CLP (R4)+ ;CLR DATA LOCATIONS
CMP #INTQ,R4 ;SEE IF ALL CLEAR
BNE 3S ;BRR IF NOT
CLR TIMOFF ;INITIALIZE TIMER ON
CLR EOP ;CLR "END PASS" INDICATOR
CNR ;SET UP "END PASS" COUNTER
MOV #INTQ,QPTR1 ;SET UP FIFO QUEUE POINTERS
MOV #INTQ,QPTR2
CLR TSRCSR,R0 ;CLR LPD STATUS OFFSET ADRS
MOV LPDDR,R0
CLR R1 ;CLR LPD STATUS OFFSET ADRS
2S: ASR R1 ;SEE IF THIS LPD IS SELECTED
MOVB #377,LPDSTS(R1) ;BRR IF NOT
MOVB DATAB(R3),(R2) ;INDICATE IN STATUS TABLE THIS LINE ACTIVE
3S: MOV #101,(R0) ;SEND ZERO
INC R1 ;SET TESTER GO & INT ENABLE BITS
ADD #10,R2 ;ADVANCE BYTE OFFSET
TST R1 ;HAVE WE STARTED ALL LPDS?
ADD #10,R0
CMP R1,#10
BNE 2S ;BRR IF NOT

```

369 ;THIS CODE RETURNS TO THE MONITOR VIA "BREAKS" FOR "END
370 ;OF PASS" TIMING - SETS FLAG "EOP" WHEN DONE
371
372 000674* 104407 000000*
373 000760* 104407 000000*
374 000764* 005767 177336
375 000710* 001371
376 000712* 005367 177324
377 000716* 001366
378 000720* 012767 177777 177316
379
380
381 ;THIS CODE ALLOWS ENOUGH TIME FOR ALL LINES TO COMPLETE A DATA TRANSFER
382 ;REPORTS ANY LPD THAT HAS FAILED TO INTERRUPT
383 ;THEN DROPS MODULE IF ALL SELECTED LPD'S HAVE BECOME HUNG
384 ;REPORTS "END PASS"
385
386 000726* 012767 001000 177306
387 000734* 104407 000000*
388 000740* 104407 000000*
389 000744* 005767 177276
390 000750* 001371
391 000752* 005367 177264
392 000756* 001366
393 000760* 016701 177022
394 000764* 185760 000316*
395 000770* 002015
396 000776* 002487 000774
397 000780* 000004
398 001006* 002701 000004
399 001012* 005367 177216
400 001016* 001004
401 001020* 104410 000000*
402 001024* 002701 000010
403 001030* 104413 000000*
404
405 001034* 000753
406
407
408 ;TESTER SERVICE LINKS
409
410
411 001036* 004567 001134
412 001040* 000000
413
414 001044* 004567 000126
415 001050* 000010
416
417 001052* 004567 000120
418 001056* 000020
419
420 001060* 004567 000110
421 001064* 000030
422
423 001066* 004567 000104
424 001070* 000040

```

```

TIMER: BREAKS,BEGIN ;TEMPORARY RETURN TO MONITOR
BREAKS,BEGIN ;THEN CONTINUE AT NEXT INSTRUCTION
TST TIMOFF ;SEE IF ER TYPEOUT IN PROGRESS
BNE TIMER ;BR IF SO - DON'T CNT ER REPORTING TIME
DEC CNTR ;COUNT RETURN TO MONITOR
BNE TIMER ;RECORD TIME FOR END PASS
MOV #1,EOP

;THIS CODE ALLOWS ENOUGH TIME FOR ALL LINES TO COMPLETE A DATA TRANSFER
;REPORTS ANY LPD THAT HAS FAILED TO INTERRUPT
;THEN DROPS MODULE IF ALL SELECTED LPD'S HAVE BECOME HUNG
;REPORTS "END PASS"
TMOUT: MOV #1000,CNTR ;SET UP TIMEOUT VALUE
IS: BREAKS,BEGIN ;TEMPORARY RETURN TO MONITOR
BREAKS,BEGIN ;THEN CONTINUE AT NEXT INSTRUCTION
TST TIMOFF ;SEE IF ER TYPEOUT IN PROGRESS
BNE IS ;BR IF SO - DON'T CNT ER REPORTING TIME
DEC CNTR ;NEED TIME FOR ALL LINES TO SHUT DOWN
BNE IS ;BR IF STALL NOT DONE
MOV ADDR,R1 ;SET UP BASE LPD ADRS
TSTP LPD$15(R0) ;LOOK AT STATUS FOR ALL LINES
BGE IS ;BR IF LPD NOT HUNG
JSE PC,CSRTST ;GO SEE IF LPD FAILED TO INTERRUPT
ADD #4,R1 ;ADVANCE TO LPD TESTER CSR ADRS
JSP PC,CSRTST ;GO SEE IF TESTER FAILED TO INT
AFC ACDEV ;SOM LPD FROM TOTAL SELECTED
BNE 4S ;BR IF OTHERS STILL SELECTED
EMSG,REGIN ;DROP MODULE - SELECTED LPD'S FAILED TO INTERRUPT
AND #10,R1 ;ADVANCE TO NEXT LPD CSR ADRS

ENDITS,BEGIN ;SIGNAL END OF ITERATION.
RS 2S ;MONITOR TEST END OF PASS
;BR IF NOT

;TESTER SERVICE LINKS
TSRV0: JSR R5,TSRSRV ;GO TO TESTER SERVICE ROUTINE
;TESTER INTERRUPT NO.
TSRV1: JSR R5,TSRSPV ;GO TO TESTER SERVICE ROUTINE
;TESTER INTERRUPT NO.
TSRV2: JSR R5,TSRSPV ;GO TO TESTER SOURCE ROUTINE
;TESTER INTERRUPT NUMBER
TSRV3: JSR R5,TSRSRV ;GO TO TESTER SOURCE ROUTINE
;TESTER INTERRUPT NUMBER
TSRV4: JSF R5,TSRSRV ;GO TO TESTER SOURCE ROUTINE
;TESTER INTERRUPT NUMBER

```

```

425
426 001074* 004567 000076
427 001100* 000050 000076
428
429 001102* 004567 000070
430 001106* 000060
431
432 001110* 004567 000062
433 001114* 000070
434
435 ;LPD SERVICE LINKS
436
437 001116* 004567 000274
438 001122* 000000
439
440 001124* 004567 000266
441 001130* 000010
442
443 001132* 004567 000260
444 001136* 000020
445
446 001140* 004567 000252
447 001144* 000030
448
449 001146* 004567 000244
450 001152* 000040
451
452 001154* 004567 000236
453 001160* 000050
454
455 001162* 004567 000230
456 001166* 000060
457
458 001170* 004567 000222
459 001174* 000070
460
461 ;TESTER INTERRUPT SERVICE - ENTERED VIA APPROPRIATE JSR TABLE
462 ;ENTRY WITH R5 POINTING TO THE INTERRUPTING TESTER - CONTAINS
463 ;OF R5 GETS QUEUED UP IN THE FIFO QUEUE AND THE ROUTINE RETURNS
464 ;CONTROL BACK TO THE MONITOR VIA A "PIRQ" TO DEFER SERVICING
465 ;THE TESTER AT LEVEL 0
466
467 001176* 011577 177034
468 001202* 002767 000002 177026
469 001218* 002787 000310 177020
470
471 001220* 012767 000270 177010
472 001226* 012605
473
474 001230* 000004 000300 001236*
475
476
477
478
479 001236* 017705 176776
480

```

```

TSRV5: JSR R5,TSRSRV ;GO TO TESTER SOURCE ROUTINE
;TESTER INTERRUPT NUMBER
TSRV6: JSR R5,TSRSRV ;GO TO TESTER SERVICE ROUTINE
;TESTER INTERRUPT NUMBER
TSRV7: JSR R5,TSRSRV ;GO TO TESTER SERVICE ROUTINE
;TESTER INTERRUPT NUMBER

;LPD SERVICE LINKS
LSRV0: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER
LSRV1: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER
LSRV2: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER
LSRV3: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER
LSRV4: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER
LSRV5: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER
LSRV6: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER
LSRV7: JSR R5,LSRV ;GO TO LPD SERVICE ROUTINE
;LPD INTERRUPT NUMBER

;DEFERRED TESTER SERVICE - THIS ROUTINE RETRIEVES THE TESTER NO.
;FROM THE FIFO QUEUE AND SERVICES THE LINE AT LEVEL 0
TSRSRV: MOV @QPTR2,R5 ;GET OLDEST INT OFFSET
;GET TESTER ID
ADD #2,QPTR1
CMP #INTQ+20,QPTR1 ;UPDATE QUEUE POINTER
BNE IS ;POINTER NOT END OF QUEUE?
MOV #INTQ,QPTR1 ;BR IF NOT
MOV (R6)+,R5 ;RESET POINTER
;RESTORE R5
;-----
;IPQS,BEGIN,TSTSRV ; QUEUE UP TO CONTINUE AT TSTSRV AND RTI
;-----

```

481 001242 062767 000002 176778 ADD #2,QPTR2 ;UPDATE QUEUE POINTER
482 001250 022767 000310 178782 CMP #INTQ+20,QPTR2 ;POINTER AT END OF QUEUE?
483 001250 001003 BNE IS ;BR IF NOT
484 001260 012767 000270 176752 MOV #INTQ,QPTR2 ;RESET POINTER
485 001260 012767 176784 176684 1S: MOV #SRCSA,CSRA ;SET UP ACTIVE DEVICE ADRS
486 001274 000567 176600 ADD R5,CSRA ;MAKE UP CURRENT TESTER ADRS
487 001300 017767 176574 MOV #CSRA,ACSR ;GET TESTER STATUS
488 001306 005377 176566 CLR #CSRA ;CLR TESTER INT EN
489 001310 022767 000300 JSR #PC,MKOFF ;GO SET UP WORD & BYTE OFFSETS
490 001310 022767 000300 JSR #R0,ACSR ;LOOK FOR INT EN & READY
491 001324 001412 BEQ #3 ;BR IF TESTER STATUS OK
492 001326 005267 176714 INC TIMOFF ;TURN OFF END PASS TIMER
493 001332 012767 000025 176546 MOV #25,ERRTYP ;BIT STUCK IN REG.
494 ***** ;*****
495 HRDERS,BEGIN,NULL ;INCORRECT TESTER STATUS
496 ***** ;*****
497 CLP TIMOFF ;TURN ON END PASS TIMER
498 001352 062767 000032 176520 2S: ADD #2,CSRA ;SET UP DBR ADRS
499 001360 004767 000326 JSR #PC,CMPDAT ;GO COMPARE DATA
500 001364 001406 BEQ #3 ;BR IF CORRECT
501 001364 005267 176654 INC TIMOFF ;TURN OFF END PASS TIMER
502 ***** ;*****
503 DATERS,BEGIN ;DATA ERROR!!!
504 ***** ;*****
505 CLR TIMOFF ;TURN ON END PASS TIMER
506 001376 005067 176644 ADD #LPCSR,R5
507 001406 000300 176646 MOV #R0,(R5) ;SET UP FOR LPD INT
508 001412 104400 000000 EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
509
510 ;LPD INTERRUPT SERVICE - ENETERD VIA APPROPRIATE JSR TABLE ENTRY
511 ;WITH R5 POINTING TO THE INTERRUPTING LPD - CONTENTS OF R5
512 ;GET QUEUED UP IN THE FIFO QUEUE AND THE ROUTINE RETURNS CONTROL
513 ;CHECK TO THE MONITOR VIA A "PIRG" TO DEFER SERVICING THR LPD
514 ;AT LEVEL 0
515
516 001416 011577 176614 LSRV: MOV (R5),QPTR1 ;GET LPD ID
517 001430 011577 000010 176606 ADD #2,QPTR1 ;UPDATE QUEUE POINTER
518 001430 022767 000310 176600 CMP #INTQ+20,QPTR1 ;POINTER AT END OF QUEUE?
519 001436 001003 BNE IS ;BR IF NOT
520 001440 011577 000270 176570 MOV #INTQ,QPTR1 ;RESET POINTER
521 001446 011577 000300 MOV (R6),R5 ;RESTORE R5
522
523 001450 000004 000000 001456 PIPQS,BEGIN,LPDSRV ;QUEUE UP TO CONTINUE AT LPDSRV AND RTI
524 -----
525 ;DEFERRED LPD SERVICE - THIS ROUTINE RETRIEVES THE LPD NO.
526 ;FROM THE FIFO QUEUE AND SERVICES THE LPD AT LEVEL 0
527
528
529
530 001462 062767 000002 176550 LPDSRV: MOV QPTR2,R5 ;GET OLDEST INT OFFSET
531 001470 022767 000310 176542 ADD #2,QPTR2 ;UPDATE QUEUE POINTER
532 001476 001003 BNE IS ;BR IF NOT
533 001500 012767 000270 176532 MOV #INTQ,QPTR2 ;RESET QUEUE POINTER
534 001506 012767 176512 176384 1S: MOV #LPCSR,CSRA ;SET UP ACTIVE DEVICE ADRS
535 001520 011577 176354 176354 MOV #CSRA,ACSR ;MAKE UP CURRENT LPDCSR ADRS
536

537 001526 005077 176346 CLR #CSRA ;CLR LPD INT EN
538 001532 004767 000226 JSR #PC,MKOFF ;GO SET UP WORD & BYTE OFFSETS
539 001532 022767 000305 176336 JSR #R0,ACSR ;LOOK AT LPD STATUS
540 001538 001412 BEQ #3 ;BR IF OK
541 001546 005267 176474 INC TIMOFF ;TURN OFF END PASS TIMER
542 001552 012767 000025 176326 MOV #25,ERRTYP ;STUCK BIT
543 ***** ;*****
544 HRDERS,BEGIN,NULL ;INCORRECT LPD STATUS
545 ***** ;*****
546 001560 011406 000000 000000 CLR TIMOFF ;TURN ON END PASS TIMER
547 001572 062767 000006 176300 2S: ADD #2,CSRA ;SET UP DBR ADRS
548 001600 004767 000101 JSR #PC,CMPDAT ;GO MAKE SURE DATA DID NOT CHANGE
549 001604 001406 BEQ #3 ;BR IF OK
550 001604 005267 176434 INC PASSCK ;TURN OFF END PASS TIMER
551 ***** ;*****
552 DATERS,BEGIN ;DATA ERROR!!!
553 ***** ;*****
554 001616 005067 176424 CLR TIMOFF ;TURN ON END PASS TIMER
555 ***** ;*****
556 ;THIS CODE CHECKS FOR "END PASS"
557 ;UPDATES THE DATA PATTERN
558
559
560 001622 005767 176416 PASSCK: TST EOP ;SEE IF DUE FOR "END PASS"
561 001626 001406 BEQ #3 ;BR IF NOT
562 001630 012767 000177 000310 MOVB #177,LPDSTS(R3) ;INDICATE IN STATUS TABLE THIS LINE COMPLETED
563 001642 005164 000280 1S: EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
564 001646 100402 COM DATAB(R4) ;COMPLEMENT DATA PATTERN
565 001650 005264 000250 BMI SFNDAT ;BR IF PATTERN NOT SENT BEFORE
566 001650 005264 000250 INC DATAB(R4) ;ADVANCE TO NEXT PATTERN
567
568 ;THIS CODE LOADS THE DATA PATTEPN & ENABLES THE TESTER INT
569
570
571 001654 066705 176346 SENDAT: ADD LPDDBR,R5
572 001660 016415 000280 MOV DATAB(R4),(R5) ;PLACE DATA IN LPD OUTPUT REG
573 001664 166705 176336 SUB LPDDBR,R5
574 001670 066705 176334 ADD #SRCSR,R5
575 001676 104400 000000 MOV #R0,(R5) ;PNABLE TESTER INTERRUPT
576 EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
577
578 ;THIS ROUTINE CHECKS THAT THE CORRECT DATA HAS BEEN
579 ;RECEIVED BY THE LPD TESTER LOAD
580
581
582 001704 066705 176322 CMPDAT: ADD #SRDBR,R5
583 001710 011567 176174 MOV (R5),#ASB ;READ AND SAVE DATA THAT WAS XMITTED
584 001714 166705 176312 SUB #SRDBR,R5
585 001720 012767 000280 MOV #DATAB,SBADR ;SET UP ADRS OF GOOD DATA
586 001726 000467 176150 ADD #R4,SBADR ;SELECT CORRECT DATA LOC
587 001732 012767 000280 MOVB #ASAS,ASAB ;SET UP ADRS OF BAD DATA
588 001736 012767 000280 MOVB #ASAS,ASAB ;GET DATA THAT WAS SENT
589 001742 000280 176146 CMP #ASAS,ASR ;DOES DATA EQUAL WHAT WAS SENT
590 001746 000280 176136 176132 RTS PC ;RETURN WITH RESULT
591
592 ;THIS ROUTINE MAKES UP WORD AND BYTE OFFSET VALUES FROM
593 ;THE DEVICE OFFSET VALUE IN R5 - OFFSETS LEFT IN R4 & R3 RESPECTIVELY
594
595 001756 010504 MKOFF: MOV R5,R4 ;GET DEVICE OFFSET
596

LPEC DEC/V11 SYSTEM EXERCISER MODULE
XLPECO.P11 12-OCT-78 11:58

MACV11 30A(1052) 12-OCT-78 16:51 PAGE 20
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0018

DIAGNOSTIC ENGINEERING

digital

DECO DEPO SUBMISSION

FOR RELEASE ENG. USE
 NEW CHANGE DELETE

PRODUCT IDENTIFICATION										
MD	LIBRARY	PRODUCT NUMBER	REV	PATCH	ECO TALLY	PRODUCT DATE	STATUS	DISTRIBUTION	1ST COPY - RIGHT YEAR	LAST COPY - RIGHT YEAR
	ZZ	CXLPE	C	1	01	DD MMM YY 3 APR 79	<input type="checkbox"/> OBSOLETE	X G <input type="checkbox"/> R	1975	1979

TITLE **CXLPEc1 LPD-11 MODULE**

AUTHOR **D. BUTENHOF** MAINTAINING GROUP **DEC/X11 SUPT GP** MAINTAINER **D. BUTENHOF** SUBMITTING ENGINEER **D. BUTENHOF**

PRODUCT COMPONENTS						
CK	DESCRIPTION	PRODUCT NO.	REV	CK	DESCRIPTION	PRODUCT NO.
	DOCUMENT				INDEX	
	LISTING				SOURCE MEDIA	
	OBJECT MEDIA				TEST MEDIA	
X	DECO	AF-E908C-M1				

PRODUCTS OBSOLETE (other than previous version)								
LIBRARY	PRODUCT NUMBER	REV	LIBRARY	PRODUCT NUMBER	REV	LIBRARY	PRODUCT NUMBER	REV
MD			MD			MD		

PRODUCT CHARACTERISTICS

PROCESSORS PRODUCT OPERATES WITH (Enter all applicable 2-digit codes representing the Processor the product operates with. See separate instructions.)

OPERATIONAL CODES (Enter all applicable 2-digit codes that describe the product. See separate instructions.)

ACT/APT/XXDP	EXT	ACT SEQ NUMBER	ACT/XXDP COMPATIBLE? <input type="checkbox"/> Y <input type="checkbox"/> N	APT COMPATIBLE? <input type="checkbox"/> Y <input type="checkbox"/> N	1ST PASS RUN TIME SECONDS	SUBSEQUENT PASS RUN TIME SECONDS
INFORMATION FIELD						

DECO/DEPO INFORMATION

PROBLEM REPORTS CLOSED: _____

DEVICE AFFECTED **DEC/X11** MULTIMEDIA AFFECTED? YES NO

KIT NUMBERS	ZJ130-RB	ZJ240-PB,	RB	ZJ240-FR	ZJ215-FR	
	ZJ129-RZ,	FR	ZJ240-RE,	RZ	ZJ215-RY,	RZ

PROBLEM:
 UNABLE TO CORRECTLY HANDLE MULTIPLE DEVICES

SOLUTION:
 WILL HANDLE ONLY ONE DEVICE WITH FOLLOWING PATCH

DEPO PATCH AREA					
CHANGE LOC	FROM	TO	CHANGE LOC	FROM	TO
722	--	177777	572	5067	177446
			574	177444	240
764	105760	105767			
766	310	177320			
566	5067	12767			
570	177452	140000			

SUBMITTING ENGINEER <i>D. Butenhof</i>	MANUFACTURING ENGINEER <i>John P. Beck</i>	SUPPORT ENGINEER	CHARGE DECO/DEPO TO DISCRETE PROJECT NUMBER Q9805460
DATE: 3 APR 79	DATE: 25-APR-79	DATE:	
MAINTAINER <i>D. Butenhof</i>	FIELD SERVICE	WAIVERING MANAGER	COORDINATION NO. 3066
DATE:	DATE:	DATE:	