

IDENTIFICATION

PRODUCT CODE: AC-E893B-MC
PRODUCT NAME: CXDPBB0 DUP-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 DIGITALS 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 ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1975,1978 DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT

DPB IS AN IOMOD THAT EXERCISES UP TO EIGHT CONSECUTIVELY
ADDRESSED SYNCHRONOUS LINE INTERFACES (DUP11) BY TRANSMITTING A STANDARD
BINARY COUNT PATTERN USING THE SYSTEM TEST MODE FEATURE. THE
RECEIVED DATA IS COMPARED WITH THE TRANSMITTED DATA AND ANY ERRORS
ARE REPORTED VIA THE CONSOLE TTY. ALL AVAILABLE INTERFACES
(UP TO 8) ARE ACTIVATED AND RUNNING SIMULTANEOUSLY.

2. REQUIREMENTS

HARDWARE: DUP11 SYNCHRONOUS INTERFACE
STORAGE:: DPB REQUIRES:
1. DECIMAL WORDS: 585
2. OCTAL WORDS: 1111
3. OCTAL BYTES: 2222

3. PASS DEFINITION

ONE PASS OF THE DPB MODULE CONSISTS OF TRANSMITTING AND RECEIVING
48,080 8-BIT CHARACTERS (TOTAL) PER ACTIVE DEVICE.

4. EXECUTION TIME

DPB RUNNING ALONE ON A PDP11/40 PROCESSOR TAKES APPROXIMATELY
1 MINUTE TO COMPLETE ONE PASS. THIS TIME INCREASES SLIGHTLY
AS MORE DEVICES ARE ADDED.

5. CONFIGURATION PARAMETERS

DEFAULT PARAMETERS:

DEVADR: 000001, VECTOR:001, BR1:5, BR2:5, DEVCNT:1
THE DEVICE IS TESTED IN SDLC MODE AS A SECONDARY STATION,
WITH A HARDWARE BCC CHECK USING THE CRC/CCITT POLYNOMIAL.

REQUIRED PARAMETERS: THE CSR AND VECTOR MUST BE SET UP BY THE USER.

6. DEVICE/OPTION SETUP

NONE: NO SPECIAL SETUP IS REQUIRED IN SYSTEM TEST MODE

7. MODULE OPERATION

TEST SEQUENCE:

- A. TEST UP TO 8 POSSIBLE DEVICES FOR SELECTION
- B. STORE THE NO. OF DEVICES TO BE TESTED AND SET UP THE VECTORS AND PRIORITIES FOR THESE DEVICES
- C. LOAD ALL REGISTERS--SECONDARY STATION, RECEIVER AND TRANSMITTER ON, SYSTEM TEST MODE AND INTERRUPT ENABLES FOR ALL ACTIVE DEVICES. ENABLE SELECTED DEVICES.
- D. TRANSMITTER INTERRUPT SERVICE:
 - 1.) TEST FOR FALSE INTERRUPT (READY (0)); REPORT ERRORS
 - 2.) OUTPUT NEXT CHARACTER TO THE DEVICE
 - 3.) RETURN TO MONITOR TO WAIT FOR RECEIVER INTERRUPT.
- E. RECEIVER INTERRUPT SERVICE:
 - 1.) TEST FOR FALSE INTERRUPT (DONE (0)); REPORT ERRORS
 - 2.) CHECK FOR DATA ERROR; REPORT ERRORS
 - 3.) RETURN TO MONITOR TO WAIT FOR TRANSMITTER INTERRUPT
- F. REPEAT D AND E UNTIL ALL DEVICES HAVE BEEN PROCESSED, THEN CHECK FOR BCC ERRORS
- G. TURN OFF ALL ACTIVE DEVICES AND DECREMENT ITERATION COUNT. IF NOT 0, RESTART AT B.
- H. SIGNAL END PASS.

8. OPERATION OPTIONS

- A. LOCATION DVID1 (DPB 14) MAY BE CHANGED TO SELECT ANY COMBINATION OF DEVICES BIT0=DEVO, BIT1=DEV1BIT7=DEV7.
NOTE: IF DVID1 IS INITIALLY 0 DPB WILL BE DROPPED FROM TEST.

9. NON STANDARD PRINTOUTS

NONE: ALL PRINTOUTS HAVE STANDARD FORMATS AS DESCRIBED IN THE DEC/X11 DOCUMENT.

203
204
205 100000
206 040000
207 020000
208 010000
209 010000
210 002000
211 001000
212 000400
213 000200
214 000100
215 000040
216 000020
217 000010
218 000004
219 000002
220 000001
221
222
223
224
225
226 100000
227 040000
228 020000
229 010000
230 004000
231 002000
232 001000
233 000400
234 000200
235 000100
236 000040
237 000020
238 000010
239 000004
240 000002
241 000001
242
243 100000
244 040000
245 020000
246 010000
247 002000
248 001000
249 000400
250 100000
251 001000
252 010000
253
254 100000
255 040000
256 020000
257 010000
258 004000

;BIT DEFINITIONS
BIT15=100000
BIT14=400000
BIT13=200000
BIT12=100000
BIT11=40000
BIT10=20000
BIT9=10000
BIT8=4000
BIT7=2000
BIT6=1000
BIT5=400
BIT4=200
BIT3=100
BIT2=40
BIT1=20
BIT0=1

CONTROL REGISTER DEFINITIONS

;RXCSR BIT DEFINITIONS
DSCA=BIT15 ;DATA SET CHANGE A
RINC=BIT14 ;RING
CR=BIT13 ;CLR TO SEND
CARDET=BIT12 ;CARRIER DETECT
RECACT=BIT11 ;REC ACTIVE
SRD=BIT10 ;SEC REC DATA
DS=BIT9 ;DATA SET RDY
SPSVN=BIT8 ;STRIP SYNC
RXDDNE=BIT7 ;REC DONE
RINTEN=BIT6 ;REC INTR ENABLE
DSMFE=BIT5 ;PRC INTR ENABLE
RCVEN=BIT4 ;REC ENABLE
STD=BIT3 ;SEC XMIT DATA
RTS=BIT2 ;REC TO SEND
DTS=BIT1 ;DATA TERM RDY
DSCB=BIT0 ;DATA SET CHANGE B
;RXDBUF BIT DEFINITIONS
RXDERR=BIT15 ;REC DATA ERROR
OVRERR=BIT14 ;OVERRUN ERROR
CRCERR=BIT12 ;CRC ERROR
RABORT=BIT10 ;REC ABORT
REOM=BIT9 ;REC END OF MESSAGE
RSUM=BIT8 ;REC START OF MESSAGE
;PARCSR BIT DEFINITIONS
DECMOD=BIT15 ;DEC MODE (DDCMP)
CRCEH=BIT9 ;CRC ENABLE
PRISEC=BIT12 ;PR/SEC SELECT
;TXCSR BIT DEFINITIONS
TXDLAT=BIT15 ;TX DATA LATE
MTDATA=BIT14 ;MAINT DATA OUT
CLK=BIT13 ;CLK
WMODEB=BIT12 ;MAINT MODE B
WMODEA=BIT11 ;MAINT MODE A

259 002000
260 001000
261 000400
262 000200
263 000100
264 000020
265 000010
266
267 000000
268 014000
269 010000
270 004000
271
272
273 100000
274 040000
275 020000
276 010000
277 004000
278 002000
279 001000
280 000400
281
282
283 000000
284 000002
285 000004
286 000004
287 000006
288
289 000224 000000
290 000226 000000
291 000230 000000
292 000234 000000
293 000234 000000
294 000236 000000
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312

;BIT WINDOW INPUT
TXACT=BIT9 ;TX ACTIVE
HRSST=BIT8 ;HARDY RESET
TXDDNE=BIT7 ;XMIT DONE
TXINTE=BIT6 ;XMIT DONE INTR ENABLE
SEMD=BIT4 ;SEMD
HDXEN=BIT3 ;DI/FOX
;TXCSR WRD DEFINITIONS
USER=0 ;USER MODE
WMODE=14000 ;MAINT INT MODE
WMT=10000 ;MAINT EXT MODE
SYSTST=4000 ;SYSTEM TEST MODE
;TXDBUF BIT DEFINITIONS
RCRC7=BIT15 ;CRC BIT WINDOW
RCRCIN=BIT14 ;DITTO
TCRC7=BIT13 ;DITTO
TCRCIN=BIT12 ;DITTO
THER=BIT11 ;MAINTENANCE TIMER
TABORT=BIT10 ;TRANSMIT ABORT
TEOM=BIT9 ;TRANSMIT END OF MESSAGE
TSUM=BIT8 ;TRANSMIT START OF MESSAGE
;MISC. PROGRAM DEFINITIONS AND EQUATES
RXCSR=R0 ;RECEIVER CONTROL REGISTER
RXDBUF=2 ;RECEIVER DATA BUFFER REGISTER
PARCSR=2 ;PARAMETER STATUS REGISTER
TXCSR=4 ;TRANSMITTER CONTROL REGISTER
TXDBUF=6 ;TRANSMITTER DATA BUFFER REGISTER
N.DUPS: OPEN ;SOFTWARE MEMORY
TOTAL: OPEN ;STORAGE FOR DATA CHECK
COUNT: OPEN ;ITERATION COUNT
RXBF: OPEN ;REC BUFFER ADORS STORAGE
SAVB: OPEN ;REC ISR TEMPORARY STORAGE
SELECT: OPEN ;SOFTWARE POINTER TO ACTIVE DEVICES
;MACRO SPUSH SA,SB,SC,SD,SE,SF
MOV SA, -(SP) ;SAVE *SA ON THE STACK
;IFB NB <SA>>
MOV SB, -(SP) ;SAVE *SB ON THE STACK
;IFB NB <SB>>
MOV SC, -(SP) ;SAVE *SC ON THE STACK
;IFB NB <SC>>
MOV SD, -(SP) ;SAVE *SD ON THE STACK
;IFB NB <SD>>
MOV SE, -(SP) ;SAVE *SE ON THE STACK
;IFB NB <SE>>
MOV SF, -(SP) ;SAVE *SF ON THE STACK
;IFB NB <SF>>
;ENDH SPUSH
;MACRO SPOP SA,SB,SC,SD,SE,SF
MOV (SP)+, SA ;POP STACK TO *SA
;IFB NB <SA>>
MOV (SP)+, SB ;POP STACK TO *SB
;IFB NB <SB>>
MOV (SP)+, SC ;POP STACK TO *SC
;IFB NB <SC>>
MOV (SP)+, SD ;POP STACK TO *SD
;IFB NB <SD>>
MOV (SP)+, SE ;POP STACK TO *SE
;IFB NB <SE>>
MOV (SP)+, SF ;POP STACK TO *SF
;IFB NB <SF>>
;ENDH SPOP

```

313
314
315
316 000240* 016767 177550 177770 START: ;BEGIN TESTING FOR THE DUP-11
317 000240* 010100 MOV DVID1,SELECT ;GET ACTIVE DEVICES
318 000250* BME RESTR1 ;BR IF ANY ARE SELECTED
319 000250* 104410 000000* DROP: ENDS,BEGIN ;;INCORRECT NUMBER OF DUP11'S
320
321
322 000254* 032767 177400 177754 RESTR1: ;SETUP VECTORS FOR ACTIVE DEVICES
323 000262* 001372 BIT #C377,SELECT ;MAKE SURE NO MORE THAN 8 SELECTED.
324 000264* 016701 BME DROP ;INVALID-NO MORE THAN 8 DEVICES!!
325 000270* 001767 MOV SELECT,R1 ;GET IMAGE OF RUNNING DEVICES
326 000272* 005067 CLR N,DUPS ;CLEAR THE # OF DUP'S (SOFTWARE)
327 000276* 016702 MOV VECTOR,R2 ;GET INITIAL VECTOR
328 000302* 016700 MOV ADDR,R0 ;GET INITIAL ADDRESS
329 000306* 012703 MOV #LXKFB,R3 ;SET ISR POINTER
330 000312* 012767 002114* 177712 MOV #RXBF0,RXBF ;SET UP BUFFER POINTERS
331 000320* 006201 1$: ASR R1 ;ACTIVE?
332 000322* 103410 BCS 38 ;BR IF DONE
333 000324* 001437 BEQ SETUP1 ;BR IF DONE
334 000326* 062703 000024 ADD #2,R3 ;PCP ISR POINTER
335 000332* 062702 000010 2$: ADD #10,R2 ;POP VECTOR
336 000336* 062700 000010 BR #10,R0 ;POP CSR
337 000342* 005267 INC 1 ;CONTINUE
338 000344* 005267 177654 3$: INC N,DUPS ;UPDATE THE # TO RUN
339 000350* 010312 MOV R3,(R2) ;LOAD ISR POINTER (RECEIVER)
340 000352* 016762 177434 000002 MOV BR1,(R2) ;LOAD PRIORITY
341 000354* 016762 000004 MOV BR2,(R3) ;LOAD PRIORITY
342 000364* 005063 000010 CLR 10(R3) ;CLR REC BYTE COUNT
343 000370* 062703 000012 ADD #12,R3 ;UPDATE POINTER
344 000374* 010362 000004 MOV R3,(R2) ;LOAD TRANSMITTER ISR
345 000400* 016762 000004 000006 MOV BR2,(R2) ;LOAD PRIORITY
346 000406* 010063 000004 MOV R0,(R3) ;LOAD CSR POINTER
347 000412* 005063 000010 CLR 10(R3) ;CLEAR TX BYTE COUNT
348 000416* 062703 000012 ADD #12,R3 ;POP ISR POINTER
349 000422* 000743 BR 2$ ;CONTINUE
350
351 ;SET UP BUFFERS
352 000424* 016767 177574 177466 SETUP1: MOV N,DUPS,INTR ;SET # OF INTERRUPTS
353 000426* 016767 177566 MOV N,DUPS,WDTO ;SET # OF WORDS TO MEM
354 000440* 016767 177560 177450 MOV N,DUPS,WDFR ;SET # OF WORDS FROM MEM
355 000446* 006367 177446 ASL INTR ;DOUBLE INTERRUPTS
356 000452* 012700 000010 MOV #8,R0 ;LOAD BUFFER SIZE
357 000454* 012703 002074* MOV #15,R3 ;LOAD BUFFER START
358 000462* 012723 000135 1$: MOV #15,(R3)+ ;UPDATE
359 000466* 005300 DEC R0 ;FOR FINISH
360 000470* 001374 BME 1$ ;BR IF MORE TO GO
361 000472* 012700 000010 MOV #8,R0 ;LOAD BUFFER STZ
362 000476* 012703 002114* MOV #RXBF0,R3 ;LOAD START ADDRESS
363 000502* 012723 000136 2$: MOV #136,(R3)+ ;CLEAR
364 000506* 005300 DEC R0 ;LOWER THE # TO DO
365 000510* 001374 BME 2$ ;BR IF MORE TO GO
366
367 ;PRELIMINARY DEVICE SETUP
368 000512* 016700 177270 SETUP2: MOV ADDR,R0 ;LOAD FIRST CSR
  
```

```

369 000516* 016701 177514 MOV SELECT,R1 ;SET UP TO GET ACTIVE DEVICES
370 000522* 006201 177566 1$: ASR R1 ;GET AN ACTIVE
371 000524* 103404 BCS 2$ ;DONE
372 000526* 001410 BEQ ACTIV ;BR IF DONE
373 000530* 062700 000010 ADD #10,R0 ;UPDATE FOR THE NEXT ONE
374 000534* 000772 BR 1$ ;CONTINUE
375 000536* 004767 000766 2$: JSR PC,PRELIM ;GO DO THE DEVICE SETUP
376 000542* 062700 000010 ADD #10,R0 ;UPDATE FOR THE NEXT ONE
377 000546* 000765 BR 1$ ;CONTINUE
378 000550* 016701 177462 177444 ACTIV: MOV SELECT,R1 ;GET THE ACTIVE DUP'S
379 000554* 016767 177444 MOV N,DUPS,TOTAL ;SET UP FOR DATA CHECK
380 000562* 016700 177220 MOV ADDR,R0 ;GET FIRST CSR
381 000566* 006201 177566 1$: ASR R1 ;GET AN ACTIVE ONE
382 000570* 103404 BCS 3$ ;BR IF ACTIVE
383 000572* 001406 BEQ WAIT ;BR IF DONE
384 000574* 062700 000010 ADD #10,R0 ;UPDATE CSR
385 000600* 000772 BR 1$ ;CONTINUE
386 000602* 004767 000764 3$: JSR PC,READY ;TURN ON DEVICE
387 000606* 000772 BR 2$ ;CONTINUE
388
389 ;DELAY AND SCAN FOR FINISH ROUTINE
390 000610* 005005 WAIT: CLR R5 ;SET FOR A LONG DELAY
391 000612* 012704 000002 MOV #2,R4 ;APPROX 30 SECONDS
392
393 SCAN:
394 000616* 1$:
395 000616* 104407 000000* BREAK$,BEGIN ;TEMPORARY RETURN TO MONITOR....
396 000622* 104407 000000* BREAK$,BEGIN ;THEN CONTINUE AT NEXT INSTRUCTION.
397 000626* 005767 177374 TST TOTAL ;GET THE # OF ACTIVE DUPS
398 000632* 001004 BME 3$ ;BR IF MORE TO GO
399
400 -----
401 ;-----
402 000634* 104413 000000* ENDT$,BEGIN ;SIGNAL END OF ITERATION.
403 000640* 000167 177410 2$: JMP RESTR1 ;MONITOR SHALL TEST END OF PASS
404 000644* 005305 3$: DEC R5 ;RESTART PROGRAM
405 000646* 001363 BNE 1$ ;STALL FOR ALL DUP'S TO FINISH
406 000650* 105304 DECB R4 ;BR IF DELAY NOT ZERO
407 000652* 001361 BNE 1$ ;OR ANOTHER DELAY
408 000654* 104403 000000* 000666* MSGNS,BEGIN,HUNG ;BR IF MORE TO GO
409 000666* 002154* ENDS,BEGIN ;ASCII MESSAGE CALL WITH COMMON HEADER
410 000670* 177777 HUNG: HUNG ;DROPPED THE MODULE
  
```

```

411      000672* 010046      TXISR: ;TRANSMITTER INTERRUPT SERVICE ROUTINE
412      000672* 010046      MOV    RO,-(SP)      ;SAVE RO ON THE STACK
413      000674* 010146      MOV    R1,-(SP)      ;SAVE R1 ON THE STACK
414      000676* 012500      MOV    (R5)+,RO      ;GET DEVICE CSR AND POP OFFSET TO R5
415      000700* 017501      MOV    (R5)+,R1      ;GET DATA ADDRESS AND LEAVE R5=COUNT
416      000702* 032760      BIT    #TXDLAY,TXCSR(RO) ;CHECK FOR OVERFLOW?
417      000710* 001427      BEQ    #5           ;BR IF NO
418      000712* 010067      CLR    RO,CSRA      ;LOAD FOR PRINTOUT
419      000715* 025067      MOV    ACSR         ;CLEAR
420      000730* 005367      MOV    #MRESET,TXCSR(RO) ;TURN OFF DEVICE
421      000734* 012601      DEC    TOTAL        ;LOWER THE # TO DO
422      000736* 017727      MOV    (SP)+,R1      ;POP STACK TO R1
423      000738* 012600      MOV    (SP)+,RO      ;POP STACK TO RO
424      000740* 012605      MOV    (SP)+,R5      ;POP STACK TO R5
425      000742* 000004      ;-----
426      000742* 000004      ;PIRQS,BEGIN,1$      ; QUEUE UP TO CONTINUE AT 1$ AND RTI
427      000750* 012767      1$:  MOV    #41,ERRTYP    ;XMITTER DATA LATE
428      000756* 104406      1$:  ;*****
429      000756* 104406      1$:  SOPERS,BEGIN,NULL    ;OVERFLOW ERROR IN TRANSMITTER
430      000756* 104406      1$:  ;*****
431      000764* 000167      2$:  JMP    SCAN          ;CONTINUE SCANNING FOR END
432      000770* 011160      2$:  MOV    (R1),TXDBUF(RO) ;PUSH OUT DATA
433      000774* 105211      2$:  INCB   (R1)         ;UPDATE DATA
434      000776* 002262      2$:  INC    (R5)         ;UPDATE THE COUNT
435      001000* 022715      2$:  CMP    #1202,(R5)    ;CHECK FOR FINISH
436      001004* 001006      2$:  BNE    #3          ;BR IF MORE TO GO
437      001006* 012760      2$:  MOV    #TEOM,TXDBUF(RO) ;END MESSAGE
438      001008* 042760      2$:  BIC    #TXINTE,TXCSR(RO) ;TURN OFF INTERRUPTS
439      001022* 012601      3$:  MOV    (SP)+,R1      ;POP STACK TO R1
440      001024* 012600      3$:  MOV    (SP)+,RO      ;POP STACK TO RO
441      001026* 012605      3$:  MOV    (SP)+,R5      ;POP STACK TO R5
442      001030* 000002      RTI
  
```

```

446      001032* 010046      RXISR: ;RECEIVER INTERRUPT SERVICE ROUTINE
447      001032* 010046      MOV    RO,-(SP)      ;SAVE RO ON THE STACK
448      001034* 010146      MOV    R1,-(SP)      ;SAVE R1 ON THE STACK
449      001036* 012500      MOV    (R5)+,RO      ;GET BUFFER ADDR AND LEAVE R5=BYTE COUNT
450      001040* 012501      MOV    (R5)+,R1      ;CHECK FOR DONE
451      001042* 105710      TSTB   (RO)         ;CHECK FOR SET
452      001044* 100427      BMI    #2$         ;BR IF SET
453      001046* 010067      MOV    RO,CSRA      ;SETUP FOR TYPEOUT
454      001048* 010067      MOV    (RO),ACSR     ;DITTO
455      001052* 011067      MOV    #MRESET,TXCSR(RO) ;TURN OFF DEVICE
456      001056* 005367      DEC    TOTAL        ;LOWER THE # TO D
457      001064* 005367      MOV    (SP)+,R1      ;POP STACK TO R1
458      001070* 012600      MOV    (SP)+,RO      ;POP STACK TO RO
459      001072* 012605      MOV    (SP)+,R5      ;POP STACK TO R5
460      001074* 012605      ;-----
461      001076* 000004      ;PIRQS,BEGIN,1$      ; QUEUE UP TO CONTINUE AT 1$ AND RTI
462      001104* 012767      1$:  MOV    #11,ERRTYP    ;ILLEGAL INTERRUPT
463      001104* 012767      1$:  ;*****
464      001112* 104405      1$:  HDRERS,BEGIN,NULL    ;FALSE INTERRUPT
465      001112* 104405      1$:  ;*****
466      001120* 000167      2$:  JMP    SCAN          ;CONTINUE SCANNING FOR END
467      001124* 016067      2$:  MOV    RXDBUF(RO),SAVBF ;GET THE BUFFER
468      001132* 005767      2$:  TST   SAVBF        ;CHECK FOR DATA ERROR
469      001134* 005767      2$:  BMI    #3$         ;BR IF ERROR FOUND
470      001140* 032710      2$:  BIT    #RECACT,(RO) ;CHECK FOR ACTIVE
471      001144* 001440      2$:  BEQ    #5$         ;BR IF NO ACTIVE
472      001146* 005215      2$:  INC    (R5)         ;UPDATE THE # TO DO
473      001154* 100516      2$:  CMP    #1200,(R5)    ;CHECK FOR DATA END
474      001156* 121167      2$:  BMI    #1$         ;BR IF DATA DONE
475      001162* 001060      2$:  CMPB   (R1),SAVBF   ;CHECK DATA
476      001164* 001060      2$:  BNE    #7$         ;BR IF NO A MATCH
477      001166* 000554      2$:  INCB   (R1)         ;UPDATE DATA
478      001170* 010067      2$:  BR    #14$        ;LEAVE
479      001174* 005067      2$:  MOV    RO,CSRA      ;SETUP FOR TYPEOUT
480      001200* 012760      3$:  CLR    ACSR         ;DITTO
481      001206* 005367      3$:  MOV    #MRESET,TXCSR(RO) ;TURN OFF DEVICE
482      001212* 012601      3$:  DEC    TOTAL        ;DECREMENT THE # TO DO
483      001214* 012600      3$:  MOV    (SP)+,R1      ;POP STACK TO R1
484      001216* 012605      3$:  MOV    (SP)+,RO      ;POP STACK TO RO
485      001218* 012605      3$:  MOV    (SP)+,R5      ;POP STACK TO R5
486      001220* 000004      ;-----
487      001220* 000004      ;PIRQS,BEGIN,4$      ; QUEUE UP TO CONTINUE AT 4$ AND RTI
488      001226* 012767      4$:  MOV    #17,ERRTYP    ;OVERRUN OR CRC ERROR
489      001226* 012767      4$:  ;*****
490      001234* 104405      4$:  HDRERS,BEGIN,NULL    ;HARDWARE DATA ERROR
491      001234* 104405      4$:  ;*****
492      001242* 000167      5$:  JMP    SCAN          ;CONTINUE SCANNING FOR END
493      001246* 010067      5$:  MOV    RO,CSRA      ;SETUP FOR TYPEOUT
494      001252* 011067      5$:  MOV    (RO),ACSR     ;DITTO
495      001256* 012760      5$:  MOV    #MRESET,TXCSR(RO) ;TURN OFF DEVICE
496      001262* 005367      5$:  DEC    TOTAL        ;LOWER THE # TO DO
497      001264* 012601      5$:  MOV    (SP)+,R1      ;POP STACK TO R1
498      001266* 012600      5$:  MOV    (SP)+,RO      ;POP STACK TO RO
499      001268* 012605      5$:  MOV    (SP)+,R5      ;POP STACK TO R5
500      001270* 012601      RTI
501      001272* 012600
  
```

```

502 001274 012605 MOV (SP)+,R5 ;POP STACK TO R5
503 -----
504 001276 000004 000000 001304 PIRQS,BEGIN,6$ ; QUEUE UP TO CONTINUE AT 6$ AND RTI
505
506 001304 012767 000042 176574 6$: MOV #42,ERRTYP ;ACTIVE CLEARED AFTER DONE
507 *****
508 001312 104405 000000 000000 HRDRS,BEGIN,NULL ;ACTIVE CLEARED AFTER A DONE
509 *****
510 001320 000167 177272 7$: JMP SCAN ;CONTINUE SCANNING FOR END
511 001324 010167 176552 MOV R1,SBADR ;LOAD GOOD DATA ADRS
512 001330 010067 176544 MOV R0,CSRA ;LOAD CSR
513 001334 012767 000234 176542 MOV #SAVBF,WASADR ;LOAD BAD DATA ADRS
514 001342 118167 176540 MOVB (R1),ASB ;LOAD GOOD DATA
515 001346 118167 176662 176534 MOVB SAVBF,WAS ;LOAD BAD DATA
516 001354 012760 000400 000004 MOV #MRESET, TXCSR(R0) ;TURN OFF DEVICE
517 001362 005367 176640 DEC TOTAL ;LOWER THE # TO DO
518 001366 012601 MOV (SP)+,R1 ;POP STACK TO R1
519 001370 012600 MOV (SP)+,R0 ;POP STACK TO R0
520 001372 012605 MOV (SP)+,R5 ;POP STACK TO R5
521 -----
522 001374 000004 000000 001402 PIRQS,BEGIN,10$ ; QUEUE UP TO CONTINUE AT 10$ AND RTI
523
524 001402 104404 000000 10$:
525 *****
526 001406 000167 177204 DATERS,BEGIN ;DATA ERROR!!!
527 *****
528 001412 022715 002262 11$: JMP SCAN ;CONTINUE SCANNING FOR END
529 001416 001940 CMP #1202,(R5) ;CHECK FOR CRC DONE
530 001420 010400 BNE #4$,ERRTYP ;BR IF NO
531 001426 001427 BIT #CRCERR,SAVBF ;CHECK FOR CRC ERROR
532 001430 010067 BEQ 13$ ;BR IF NO
533 001434 005967 MOV R0,CSRA ;SET UP FOR TYPEOUT
534 001440 004400 CLR ACSR
535 001446 005367 MOV #MRESET, TXCSR(R0) ;TURN OFF DEVICE
536 001452 012601 DEC TOTAL ;LOWER THE # TO DO
537 001454 012600 MOV (SP)+,R1 ;POP STACK TO R1
538 001456 012605 MOV (SP)+,R0 ;POP STACK TO R0
539 001460 000004 000000 001466 PIRQS,BEGIN,12$ ; QUEUE UP TO CONTINUE AT 12$ AND RTI
540
541 001466 012767 000043 176412 12$: MOV #43,ERRTYP ;CRC ERROR
542 *****
543 001474 104405 000000 000000 HRDRS,BEGIN,NULL ;HARDWARE DETECTED CRC ERROR
544 *****
545 001502 000167 177110 JMP SCAN ;CONTINUE SCANNING FOR END

```

```

548 001506 012760 000400 000004 13$: MOV #MRESET, TXCSR(R0) ;TURN OFF DEVICE
549 001514 005367 176506 DEC TOTAL ;LOWER THE # TO DO
550
551 001520 012601 14$: MOV (SP)+,R1 ;POP STACK TO R1
552 001524 012600 MOV (SP)+,R0 ;POP STACK TO R0
553 001526 000002 MOV (SP)+,R5 ;POP STACK TO R5
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```



```

579 ;SERVICE CODE FOR LINKING A PARTICULAR DEVICE
580 ;TO A COMMON TRANSMIT OR RECEIVE INTERRUPT SERVICE ROUTINE.
581
582 001614*
583 LNKTAB:
584 .MACRO $JS,$Q
585 JSR R5,RXISR ;ANSWER FOR DEVICE *SQ REC ISR
586 0 ;CSR FOR DEVICE *SQ REC ISR
587 RXBF*$Q ;BUFFER LOC FOR DEVICE *SQ
588 BCNT*$Q ;BYTE COUNT ADDRESS FOR RXBF*$Q
589
590 JSR R5,TXISR ;ANSWER FOR DEVICE *SQ TX ISR
591 0 ;CSR FOR DEVICE *SQ TX ISR
592 TXBF*$Q ;DATA ADDRESS FOR DEVICE *SQ
593 OFSET*$Q ;BYTE COUNT ADDRESS FOR DEVICE *SQ
594
595 .ENDM $JS
596
597 001614* 004567 177212 JSR R5,RXISR ;ANSWER FOR DEVICE 0 REC ISR
598 0 ;CSR FOR DEVICE 0 REC ISR
599 RXBF0 ;BUFFER LOC FOR DEVICE 0
600 BCNT0 ;BYTE COUNT ADDRESS FOR RXBF0
601
602 001626* 004567 177040 JSR R5,TXISR ;ANSWER FOR DEVICE 0 TX ISR
603 0 ;CSR FOR DEVICE 0 TX ISR
604 TXBF0 ;DATA ADDRESS FOR DEVICE 0
605 OFSET0 ;BYTE COUNT ADDRESS FOR DEVICE 0
606
607 001640* 004567 177166 JSR R5,RXISR ;ANSWER FOR DEVICE 1 REC ISR
608 0 ;CSR FOR DEVICE 1 REC ISR
609 RXBF1 ;BUFFER LOC FOR DEVICE 1
610 BCNT1 ;BYTE COUNT ADDRESS FOR RXBF1
611
612 001652* 004567 177014 JSR R5,TXISR ;ANSWER FOR DEVICE 1 TX ISR
613 0 ;CSR FOR DEVICE 1 TX ISR
614 TXBF1 ;DATA ADDRESS FOR DEVICE 1
615 OFSET1 ;BYTE COUNT ADDRESS FOR DEVICE 1
616
617 001664* 004567 177142 JSR R5,RXISR ;ANSWER FOR DEVICE 2 REC ISR
618 0 ;CSR FOR DEVICE 2 REC ISR
619 RXBF2 ;BUFFER LOC FOR DEVICE 2
620 BCNT2 ;BYTE COUNT ADDRESS FOR RXBF2
621
622 001676* 004567 176770 JSR R5,TXISR ;ANSWER FOR DEVICE 2 TX ISR
623 0 ;CSR FOR DEVICE 2 TX ISR
624 TXBF2 ;DATA ADDRESS FOR DEVICE 2
625 OFSET2 ;BYTE COUNT ADDRESS FOR DEVICE 2
626
627 001710* 004567 177116 JSR R5,RXISR ;ANSWER FOR DEVICE 3 REC ISR
628 0 ;CSR FOR DEVICE 3 REC ISR
629 RXBF3 ;BUFFER LOC FOR DEVICE 3
630 BCNT3 ;BYTE COUNT ADDRESS FOR RXBF3
631
632 001722* 004567 176744 JSR R5,TXISR ;ANSWER FOR DEVICE 3 TX ISR
633 0 ;CSR FOR DEVICE 3 TX ISR
634 TXBF3 ;DATA ADDRESS FOR DEVICE 3
635 OFSET3 ;BYTE COUNT ADDRESS FOR DEVICE 3

```

```

635
636 001734* 004567 177072 JSR R5,RXISR ;ANSWER FOR DEVICE 4 REC ISR
637 0 ;CSR FOR DEVICE 4 REC ISR
638 RXBF4 ;BUFFER LOC FOR DEVICE 4
639 BCNT4 ;BYTE COUNT ADDRESS FOR RXBF4
640
641 001746* 004567 176720 JSR R5,TXISR ;ANSWER FOR DEVICE 4 TX ISR
642 0 ;CSR FOR DEVICE 4 TX ISR
643 TXBF4 ;DATA ADDRESS FOR DEVICE 4
644 OFSET4 ;BYTE COUNT ADDRESS FOR DEVICE 4
645
646 001760* 004567 177046 JSR R5,RXISR ;ANSWER FOR DEVICE 5 REC ISR
647 0 ;CSR FOR DEVICE 5 REC ISR
648 RXBF5 ;BUFFER LOC FOR DEVICE 5
649 BCNT5 ;BYTE COUNT ADDRESS FOR RXBF5
650
651 001772* 004567 176674 JSR R5,TXISR ;ANSWER FOR DEVICE 5 TX ISR
652 0 ;CSR FOR DEVICE 5 TX ISR
653 TXBF5 ;DATA ADDRESS FOR DEVICE 5
654 OFSET5 ;BYTE COUNT ADDRESS FOR DEVICE 5
655
656 002004* 004567 177022 JSR R5,RXISR ;ANSWER FOR DEVICE 6 REC ISR
657 0 ;CSR FOR DEVICE 6 REC ISR
658 RXBF6 ;BUFFER LOC FOR DEVICE 6
659 BCNT6 ;BYTE COUNT ADDRESS FOR RXBF6
660
661 002016* 004567 176650 JSR R5,TXISR ;ANSWER FOR DEVICE 6 TX ISR
662 0 ;CSR FOR DEVICE 6 TX ISR
663 TXBF6 ;DATA ADDRESS FOR DEVICE 6
664 OFSET6 ;BYTE COUNT ADDRESS FOR DEVICE 6
665
666 002030* 004567 176776 JSR R5,RXISR ;ANSWER FOR DEVICE 7 REC ISR
667 0 ;CSR FOR DEVICE 7 REC ISR
668 RXBF7 ;BUFFER LOC FOR DEVICE 7
669 BCNT7 ;BYTE COUNT ADDRESS FOR RXBF7
670
671 002042* 004567 176624 JSR R5,TXISR ;ANSWER FOR DEVICE 7 TX ISR
672 0 ;CSR FOR DEVICE 7 TX ISR
673 TXBF7 ;DATA ADDRESS FOR DEVICE 7
674 OFSET7 ;BYTE COUNT ADDRESS FOR DEVICE 7
675

```

```
676 ;BUFFER AREAS
677
678 .MACRO $OF,$Q
679 OFSET$Q: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE *$Q
680 .ENDM
681 002054 000000 OFSET0: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 0
682 002056 000000 OFSET1: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 1
683 002060 000000 OFSET2: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 2
684 002062 000000 OFSET3: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 3
685 002064 000000 OFSET4: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 4
686 002066 000000 OFSET5: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 5
687 002070 000000 OFSET6: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 6
688 002072 000000 OFSET7: .WORD 0 ;TRANSMITTER BYTE COUNT POINTER FOR DEVICE 7
689
690 .MACRO $TB,$Q
691 TXBF$Q: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE *$Q
692 .ENDM
693
694 002074 000000 TXBF0: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 0
695 002076 000000 TXBF1: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 1
696 002100 000000 TXBF2: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 2
697 002102 000000 TXBF3: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 3
698 002104 000000 TXBF4: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 4
699 002106 000000 TXBF5: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 5
700 002110 000000 TXBF6: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 6
701 002112 000000 TXBF7: .WORD 0 ;TRANSMITTER BUFFER FOR DEVICE 7
702
703 .MACRO $RB,$Q
704 RXBF$Q: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #*$Q
705 .ENDM
706
707 002114 000000 RXBF0: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #0
708 002116 000000 RXBF1: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #1
709 002120 000000 RXBF2: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #2
710 002122 000000 RXBF3: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #3
711 002124 000000 RXBF4: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #4
712 002126 000000 RXBF5: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #5
713 002130 000000 RXBF6: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #6
714 002132 000000 RXBF7: .WORD 0 ;RECEIVER BUFFER FOR DEVICE #7
715
716 .MACRO $BC,$Q
717 BCNT$Q: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE *$Q
718 .ENDM
719
720 002134 000000 BCNT0: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 0
721 002136 000000 BCNT1: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 1
722 002140 000000 BCNT2: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 2
723 002142 000000 BCNT3: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 3
724 002144 000000 BCNT4: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 4
725 002146 000000 BCNT5: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 5
726 002150 000000 BCNT6: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 6
727 002152 000000 BCNT7: .WORD 0 ;RECEIVER BYTE COUNT POINTER FOR DEVICE 7
728
729
730
731 ;ASCII MESSAGES
```

```
732 002154 042045 050125 030461 MHUNG: .ASCII "&DUP11 MODULE IS HUNG - SEE LISTING!"
733 002222 .EVEN
000001 .END
```

ACSR	000102R	185#	420*	455*	482*	497*	534*											
ACTIV	000550R	372#	378#															
ADDR	000066R	163#	328#	368	380													
ADDR22=	001000R	203#																
ASB	000106R	189#	514*															
ASTAT	000104R	187#																
AWAS	000110R	187#	515*															
BCNT0	002134R	599#	720#															
BCNT1	002136R	609#	721#															
BCNT2	002140R	619#	722#															
BCNT3	002142R	639#	723#															
BCNT4	002144R	639#	724#															
BCNT5	002146R	649#	725#															
BCNT6	002150R	659#	726#															
BCNT7	002152R	669#	727#															
BEGIN	000000R	149#	319#	394	395	400	407	408	427	431	462	466	489	493				
		504#	508#	522	526	541	545											
BITW	002000	259#	220#															
BIT0	000001	203#	241#															
BIT1	000002	203#	240#															
BIT10	002000	203#	231#		246	259	278											
BIT11	004000	203#	230#		248	257	276											
BIT12	010000	203#	229#		245	257	276											
BIT13	001000	203#	228#		246	256	275											
BIT14	004000	203#	227#		244	255	274											
BIT15	100000	203#	226#		243	254	273											
BIT2	000004	203#	219#															
BIT3	000008	203#	218#		265													
BIT4	000020	203#	216#		264													
BIT5	000040	203#	215#															
BIT6	000100	203#	214#		263													
BIT7	000100	203#	213#		262													
BIT8	000400	203#	212#		261	280												
BIT9	001000	203#	211#		247	260	279											
BREAKS	104407	203#	304#		395													
BR1	000102R	153#	340#															
BR2	000013R	154#	345#															
BTODS	104421	203#																
CARDET	010000	203#																
CDATAS	001000	203#																
CLK	020000	203#																
CONFIG	000056R	173#																
COUNT	000230R	229#																
CRCEM	001000	203#																
CRCEMR	010000	244#	531*															
CSRA	000100R	183#	419*	454*	481*	496*	512*	533*										
CTS	020000	228#																
DATCKE	104411	203#																
DATER	104404	203#	526															
DECMOD	100000	250#																
DRDP	000250R	319#	323	325														
DSCA	100000	241#																
DSCR	000001	241#																
DSINTE	000040	236#																
DSR	001000	232#																
DTR	000002	240#																

DVID1	000014R	155#	316															
ENDIT	104413	203#	400															
ENDS	104410	203#	319															
ERRV	004106R	188#	429*	408	464*	491*	506*	543*										
EXITS	104106	203#																
GETPA	104415	203#																
GWBUF	104414	203#																
HDAEN	000044R	169#																
HRDWR	104405	203#	466	493	508	545												
HRDPA	000050R	170#																
HUNG	000066R	165#	409#															
ICOUNT	000040R	166#																
IDNUM	000122R	195#																
INIT	000000R	162#																
INTR	000120R	184#	352*	355*														
LWKTAB	001614R	329#	582#															
MAP22	104416	203#																
NEXT	010000	269#																
NHUNG	002154R	409#	732#															
NMODE	014000	268#																
NMODEA	004000	258#																
NMODEB	010000	149#																
MODNAM	000000R	149#																
MODSP	000224R	163#	201#															
MRESET	000400	263#	421	456	483	498	516	535	548	564								
MSGCS	104403	203#	407															
MSGZ	104402	203#																
MSGS	104401	203#																
MTDATA	040000	255#																
NULL	000029R	203#	431	466	493	508	545											
N.DUPS	000000	289#	326*	338*	352	354	379											
OPSET0	002054R	604#	681#															
OPSET1	002056R	614#	682#															
OPSET2	002060R	624#	683#															
OPSET3	002062R	634#	684#															
OPSET4	002064R	644#	685#															
OPSET5	002066R	654#	686#															
OPSET6	002070R	664#	687#															
OPSET7	002072R	674#	688#															
OPEN	000000	150#	156	157	158	159	176	177	178	179	180	181	182					

PRTV2	==	000100	203#																	
PRTV3	==	000140	203#																	
PRTV4	==	000200	203#																	
PRTV5	==	000240	154	203#																
PRTV6	==	000300	203#																	
PRTV7	==	000340	203#																	
PS	==	177776	203#																	
PSH	==	177776	203#																	
PUSH	==	005746	203#																	
PUSH2	==	04646	203#																	
RABORT	==	002100	246#																	
RANDS	==	104417	203#																	
RANNOU	==	000054R	172#																	
RCRCIN	==	040000	274#																	
RCRCIT	==	100000	273#																	
RCVEN	==	000020	237#																	
READY	==	001572R	386	571#																
RECACT	==	004000	240	472																
REOM	==	001000	247#																	
RESTR	==	000254R	191	317	322#	402														
RES1	==	000056R	174#																	
RES2	==	000060R	174#																	
RETURN	==	001160R	570	573#																
RING	==	040000	227#																	
RINTEN	==	000100	235#	568																
RSOM	==	000400	489#																	
RSTRT	==	000112R	191#																	
RPS	==	000004	239#																	
RXBF	==	000232R	292#																	
RXBF0	==	002114R	250	330*	598	707#														
RXBF1	==	002114R	250	362																
RXBF2	==	002120R	608	708#																
RXBF3	==	002122R	618	709#																
RXBF4	==	002124R	628	710#																
RXBF5	==	002124R	638	711#																
RXBF6	==	002130R	658	712#																
RXBF7	==	002132R	668	713#																
RXDBUF	==	000002	268	714#																
RXDERR	==	100000	263#																	
RXDONE	==	000200	234#																	
RXISR	==	001032R	447#	596	606	616	626	636	646	656	666									
SAVBR	==	000248R	262	470	477	513	515	531												
SADR	==	001120R	193	511*																
SCAN	==	000616R	392	433	468	495	510	528	547											
SELECT	==	000236R	294	316*	322	324	369	378												
SEND	==	000020	264	369																
SETUP1	==	000248R	264	352#																
SETUP2	==	000512R	368#																	
SMALL	==	001610R	565	576#																
SOPCMT	==	000042R	167																	
SOPERS	==	104406	203#																	
SOPFAS	==	000046R	169	431																
SPOINT	==	000032R	163																	
SPSIZ	==	000040	231	196																
SRO	==	002000	913#																	
SRI	==	000016R	156#																	

SR2	==	000020R	157#																	
SR3	==	000022R	158#																	
SR4	==	000044R	162#																	
START	==	000240R	162	316#																
STAT	==	000026R	161#																	
STD	==	000010	238#																	
STPSVN	==	000400	333#																	
SVR0	==	000062R	176#																	
SVR1	==	000064R	177#																	
SVR2	==	000066R	178#																	
SVR3	==	000070R	179#																	
SVR4	==	000072R	180#																	
SVR5	==	000074R	181#																	
SVR6	==	000076R	182#																	
SVSCMT	==	000052R	111																	
SVYSTST	==	004000	270#	566																
TABORT	==	002000	278#																	
TCRCIN	==	010000	276																	
TCRCIT	==	020000	275																	
TEOM	==	001000	279#	439																
TIMER	==	004000	277																	
TOTAL	==	000226R	290	379*	396	422*	457*	484*	499*	517*	536*	549*								
TRPDFD	==	000022	203																	
TSDM	==	000460	280#	571																
TXACT	==	001000	260#																	
TXBF0	==	002074R	357	603	694#															
TXBF1	==	002076R	613	695#																
TXBF2	==	002100R	623	696#																
TXBF3	==	002102R	633	697#																
TXBF4	==	002104R	643	698#																
TXBF5	==	002106R	653	699#																
TXBF6	==	002110R	663	700#																
TXBF7	==	002112R	673	701#																
TXCSR	==	000004	286#	417	421*	440*	456*	483*	498*	516*	535*	548*	564*	566*	569*					
TXDBUF	==	000006	287#	434*	439*	571*														
TXDLAT	==	100000	254	417																
TXDONE	==	000200	262																	
TXIWE	==	000100	263																	
TXISR	==	000022R	417	601	611	621	631	641	651	661	671									
USER	==	000000	267																	
VECTOR	==	000010R	152	327																
WAIT	==	000610R	383	390#																
WASADR	==	000104R	186	513*																
WDFR	==	000116R	193	354*																
WDT0	==	000114R	192	353*																
XFLAG	==	000005R	160																	
.		002222R	732																	

* ABS. 000000 000
 002222 001

ERRORS DETECTED: 0
 DEFAULT GLOBALS GENERATED: 0

DPBB DEC/X11 SYSTEM EXERCISER MODULE MACV11 30A(1052) 12-OCT-78 16:30 PAGE 24
XDPBB0.P11 12-OCT-78 11:53 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0022

XDPBB0, XDPBB0/SOL /CRF:SYM=DDXCOM, XDPBB0
RUN-TIME: 2 2 4 SECONDS
RUN-TIME RATIO: 12/5=2.5
CORE USED: 7K (13 PAGES)