

*REM !

IDENTIFICATION

PRODUCT CODE: AC-E914D-MC
PRODUCT NAME: CXIBAD0 IBV-11A 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 OR EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1976,1978 DIGITAL EQUIPMENT CORPORATION

1.0

ABSTRACT

THE IBA IS AN IOMOD THAT EXERCISES THE IBV11-A INSTRUMENTATION BUS INTERFACE (IB-BUS). ON START IT DOES A BRIEF LOGIC TEST. ON RESTART AND AFTER ENDPASS IT EXERCISES THE IBV11-A BY MAKING IT INTERRUPT AND PASS DATA TO ITSELF.

IT IS RECOMMENDED THAT YOU REMOVE THE IR-BUS CABLE FROM THE IBV11 BEFORE YOU RUN THIS MODULE. HOWEVER, IF YOU CHOOSE NOT TO AND YOU DO GET ERRORS, THE ERRORS COULD BE CAUSED FROM SOME DEVICE ON THE IR-BUS AND NOT THE IBV11-A.

2.0

REQUIREMENTS

HARDWARE: ONE IBV11-A.

STORAGE: IBA REQUIRES:

1. DECIMAL WORDS: 417
2. OCTAL WORDS: 0641
3. OCTAL BYTES: 1502

3.0

PASS DEFINITION

ONE PASS OF THE IBA MODULE CONSISTS OF 17700 (OCTAL) DATA TRANSFERS TO AND FROM THE IBV11-A.

4.0

EXECUTION TIME

ONE PASS OF THE IBA MODULE RUNNING ALONE TAKES APPROXIMATELY ONE MINUTE.

5.0

CONFIGURATION REQUIREMENTS

DEFAULT PARAMETERS:
DEVADR:160150, VECTOR:640, BRI:4, DEVCNT:1, SRI:0

REQUIRED PARAMETERS:
NONE.

6.0

DEVICE/OUTPUT SET-UP:

NONE.

7.0 MODULE OPERATION

TEST	SEQUENCE
1. (START)	BIT EXERCISE CSR, DATA REGISTER.
2. (RESTART)	DATA TRANSFERS
A.	SET INTERRUPTS ENABLE AND TCS.
B.	SET INTERRUPTS TO "KRSRV"
C.	SET "ION" AND "LON", CLEAR TCS
D.	IBV INTERRUPTS TO "KRSRV"
E.	(DATA PATTERN GENERATED AND LOADED INTO IBV (DATA REGISTER)
F.	DATA GETS "GATED" ONTO IB-BUS AND RECEIVED BY IBV
G.	THUS GAINING A "LWR" INTERRUPT.
H.	"LWR" INTERRUPTS TO "LNRSRV"
I.	DATA RECEIVED IS COMPARED AGAINST DATA SENT, IF UNEQUAL, AN ERROR IS REPORTED. IF ALL DATA TRANSFERS HAVE BEEN MADE, AN "ENDPASS" IS REPORTED, ELSE PROGRAM CONTROL IS RETURNED TO STEP D.

8.0 OPERATION OPTIONS

VALID SRI VALUES

SRI BIT	ENABLE/DISABLE	FUNCTION
0	0	ENABLE RANDOM PATTERN TRANSFERS
	1	ENABLE COMPLEMENT PATTERN TRANSFERS. PATTERN IN LOCATION "PATTERN" IS COMPLEMENTED EACH TIME IT IS USED.

9.0 NON-STANDARD PRINTOUTS

ALL PRINTOUTS HAVE THE STANDARD FORMATS DESCRIBED IN THE
DEC/X11 DOCUMENT.

!

```

126 ; .TITLE IBAD DEC/X11 SYSTEM EXERCISER MODULE
127 ; DDXCOM VERSION 6 23-MAY-78
128 ; .LIST BIN
129 ;*****
130 BEGIN:
131 MODNAM: .ASCII /IBAD / ;MODULE NAME.
132 XFLAG: .BYTE OPEN ;USED TO KEEP TRACK OF WBUFF USAGE
133 ADDR: 160150+0 ;1ST DEVICE ADDR
134 VECTOR: 640+0 ;1ST DEVICE VECTOR.
135 BR1: .BYTE PRTY6+0 ;1ST BR LEVEL.
136 BR2: .BYTE PRTY6+0 ;2ND BR LEVEL.
137 DVID1: +1 ;DEVICE INDICATOR 1.
138 SR1: OPEN ;SWITCH REGISTER 1.
139 SR2: OPEN ;SWITCH REGISTER 2.
140 SR3: OPEN ;SWITCH REGISTER 3.
141 SR4: OPEN ;SWITCH REGISTER 4.
142 ;*****
143 STAT: 140000 ;STATUS WORD.
144 INIT: START ;MODULE START ADDR.
145 SPOINT: MODSP ;MODULE STACK POINTER.
146 PASCNT: 0 ;PASS COUNTER.
147 ICOUNT: 177000 ;# OF ITERATIONS PER PASS=177000
148 ICOUNT: 0 ;LOC TO COUNT ITERATIONS
149 SOFCNT: 0 ;LOC TO SAVE TCTAL SOFT ERRORS
150 HRDCNT: 0 ;LOC TO SAVE TOTAL HARD ERRORS
151 SOFPAS: 0 ;LOC TO SAVE SOFT ERRORS PER PASS
152 HRDPAS: 0 ;LOC TO SAVE HARD ERRORS PER PASS
153 SYSCNT: 0 ;# OF SYS ERRORS ACCUMULATED
154 CSRAND: 0 ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
155 CONFIG:
156 RES1: 0 ;RESERVED FOR MONITOR USE
157 RES2: 0 ;RESERVED FOR MONITOR USE
158 RES3: 0 ;RESERVED FOR MONITOR USE
159 SVR1: OPEN ;LOC TO SAVE R1.
160 SVR2: OPEN ;LOC TO SAVE R2.
161 SVR3: OPEN ;LOC TO SAVE R3.
162 SVR4: OPEN ;LOC TO SAVE R4.
163 SVR5: OPEN ;LOC TO SAVE R5.
164 SVR6: OPEN ;LOC TO SAVE R6.
165 CSADR: OPEN ;ADDR OF CURRENT CSR.
166 ACSR: OPEN ;ADDR OF GOOD DATA, OR
167 WASADR: OPEN ;ADDR OF BAD DATA, OR
168 ASADR: OPEN ;STATUS REG CONTENTS.
169 ERSTYP: 644 ;TYPE OF ERROR
170 ASB: OPEN ;EXPECTED DATA.
171 AWAS: OPEN ;ACTUAL DATA.
172 RSTRT: RESTR ;RESTART ADDRESS AFTER END OF PASS
173 WDT0: OPEN ;WORDS TO MEMORY PER ITERATION
174 WDFR: OPEN ;WORDS FROM MEMORY PER ITERATION
175 INTR: OPEN ;# OF INTERRUPTS PER ITERATION
176 IDNUM: 141 ;MODULE IDENTIFICATION NUMBER=141
177 MODSP:
178 ;*****
179

```

```

180 ;MODULE REQUIRED REGISTERS - SET UP BY THIS MODULE.
181
182 000224 160150 IBS: .WORD 160150 ;CONTROL AND STATUS
183 000226 160152 IBD: .WORD 160152 ;DATA REGISTER
184
185
186 000230 000640 VECTA: .WORD 640 ;ERROR INTR. VECTOR
187 000232 000642 PRA: .WORD 642 ;SRQ INTR. VECTOR
188 000234 000644 PRB: .WORD 644 ;TKR OR CMD INTR. VECTOR
189 000236 000646 PRD: .WORD 646 ;LNR INTR. VECTOR
190 000240 000650 PRC: .WORD 650
191 000242 000652 PRD: .WORD 652
192 000244 000654 PRD: .WORD 654
193 000246 000656 PRD: .WORD 656
194
195 ;CONSTANTS AND STORAGE LOCATIONS
196
197
198 000250 000000 CNT: .WORD 0 ;ICONT-1
199 000252 000252 PATERN: .WORD 252 ;CURRENT PATTERN BEING
;TRANSFERRED BETWEEN TKR AND LNR
;ON IBV11 BUS.
200
201
202 000254 123456 RANA: 123456 ;RANDOM NUMBERS GENERATED
203 000256 071234 RANB: 071234 ;BY RANDOM NUMBER GENERATOR
;TO GET "PATERN" IF NOT
;INHIBITED BY "SR1"
204
205
206
207
208
209 000260 016767 177552 177762 START: MOV ICOUNT,CNT ;SAVE ITERATION COUNT
210 000265 005367 177756 DEC CNT ;DEFINE NEXT TO LAST ITERATION
211 000270 016767 177510 177724 MOV ADDR,IBS ;GET BASE CSR ADDR.
212 000300 016767 177720 177720 MOV IBS,IBD ;NOW FORM DBR ADDR.
213 000306 062767 000002 177712 ADD #2,IBD ;=CSR+2.
214
215
216 000314 016700 177470 MOV VECTOR,RO ;NOW LETS FIX VECTOR ADDRESS
217 000320 010067 177704 MOV RO,VECTA ;/SET THIS VECTOR ADDR.
218 000324 062700 000002 ADD #2,RO ;/UPDATE ADDR FOR NEXT VECTOR.
219 000330 010067 177676 MOV RO,PRA ;/SET THIS VECTOR ADDR.
220 000334 062700 000002 ADD #2,RO ;/UPDATE ADDR FOR NEXT VECTOR.
221 000340 010067 177648 MOV RO,PRB ;/UPDATE ADDR FOR NEXT VECTOR.
222 000344 062700 000002 ADD #2,RO ;/SET THIS VECTOR ADDR.
223 000350 010067 177620 MOV RO,PRD ;/UPDATE ADDR FOR NEXT VECTOR.
224 000354 062700 000002 ADD #2,RO ;/SET THIS VECTOR ADDR.
225 000360 010067 177654 MOV RO,PRC ;/UPDATE ADDR FOR NEXT VECTOR.
226 000364 062700 000002 ADD #2,RO ;/SET THIS VECTOR ADDR.
227 000370 010067 177646 MOV RO,PRD ;/UPDATE ADDR FOR NEXT VECTOR.
228 000374 062700 000002 ADD #2,RO ;/SET THIS VECTOR ADDR.
229 000400 010067 177640 MOV RO,VECTD ;/UPDATE ADDR FOR NEXT VECTOR.
230 000410 010067 177632 MOV RO,PRD ;/SET THIS VECTOR ADDR.
231
232
233 .REM I
234
235 IBS - IBV11-A CONTROL & STATUS REGISTER

```

236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291

IBS INSTRUMENT BUS CONTROL & STATUS REGISTER 16X150
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
SRQ ER2 ER1 EPO 0 CMD TKR LNR ACC IE TON LON IBC PEM EOP TCS
R R R R R R R R W W W W W W W W

R MEANS THIS BIT MAY ONLY BE READ BY THE LSI-11.
W MEANS THIS BIT MAY BE BOTH READ AND WRITTEN BY THE LSI-11.

IBD - IBV11-A DATA REGISTER

THE HIGH BYTE MONITORS THE 5 IB CONTROL LINES AND THE 3 IB DATA HANDSHAKE LINES. THE DATA ON THESE LINES IS LATCHED TO REMAIN STABLE WHILE BEING READ. THE LOW BYTE MONITORS THE IB DIO LINES DIRECTLY WHEN READ. DATA WRITTEN INTO THE LOW BYTE IS LATCHED INTO A BUFFER AND SENT OUT ON THE IB DIO LINES IF TON OR TCS AND ATN ARE SET. THE BUFFER IS ALWAYS CLEARED BY DAC.

IBD INSTRUMENT BUS DATA REGISTER 16X152
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
EOI ATN IFC REN SRQ RFD DAV DAC IOB IO7 IO6 IO5 IO4 IO3 IO2 IO1
R R R R R R R R W W W W W W W W

R MEANS THIS BIT MAY ONLY BE READ BY THE LSI-11.
W MEANS THIS BIT MAY BE BOTH READ AND WRITTEN BY THE LSI-11.

!
;* LOGIC TEST #1 BE SURE AN IBV11-A EXISTS AT THE
;* SPECIFIED ADDR. IF NO IBV11-A, THEN A DEC/X11
;* SYS ERROR WILL OCCUR.
;*

000414* 005777 177604 LOG1: TST @IBS ;ADDRESS THE IBV. IF SYS ERROR
;OCCURS, THEN IBV DID NOT RESPOND
;WHEN ADDRESSED.
;MAKE SURE THAT THE ADDRESS
;FORMED BY THE SWITCH PACK ON THE
;OPTION AGREE WITH THE ADDRESS
;THIS SOFTWARE HAS IN "ADDR".

;* LOGIC TEST #2
;* IN THIS TEST WE ARE GOING TO SET
;* TCS (CSR BIT0), EOP (CSR BIT01),
;* REM (BIT02), LON (CSR BIT04),
;* TON (BIT05), AND ACC (CSR BIT07)
;* NOW, TCS SHOULD CAUSE CMD (CSR BIT10)
;* AND DAC (IBD BIT8) TO SET. EOP
;* SHOULD CAUSE EOI (IBD BIT15) TO SET.
;* REM SHOULD CAUSE REN (IBD BIT12) TO SET.
;*

292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347

;* LON SHOULD CAUSE LNR (CSR BIT8) TO SET.
;* TON SHOULD CAUSE TKR (CSR BIT9) TO SET.
;* TCS SHOULD ALSO CAUSE ATN (IBD BIT14) TO SET.
;* WE ARE GOING TO CHECK THAT ALL THESE BITS SET.
;* IF YOU GET ANYTHING LIKE AN ERROR1 OR ERROR2 OR SRQ
;* SET - READ DESCRIPTION PRECEDING "ERRSRV".
;*

000420* 012777 000267 177576 LOG2: MOV #267,@IBS ;LOAD IBS WITH ACC, TON, LON, EOP
;AND TCS
MOV @IBS,ACSR ;READ BACK IBS
MOV #002267,ASTAT ;EXPECT ACC, TON, LON, EOP, TCS, REM
;AND CMD TO
;BE SET
CMP ACSR,ASTAT ;DID THEY ALL SET?
BEQ 15 ;YES - THEN NEXT CHECK
MOV @IBS,CSRA ;NO - RECORD CSR'S ADDR.
MOV #25,ERRTYP
;*****
HDRS,BEGIN,NULL ;CSR PROBLEM WITH CSR BITS ASTAT=GOOD, ACSR=BAD
;*****

000474* 017767 177526 177400 1S: MOV @IBD,ACSR ;NOW READ DATA REGISTER
MOV #152000,ASTAT ;EXPECT EOI, ATN, REN AND
;RFD TO BE SET.
CMP ACSR,ASTAT ;DID THEY ALL SET?
BEQ 25 ;YES - NEXT TEST.
MOV @IBD,CSRA ;NO - RECORD DBR'S ADDR.
MOV #25,ERRTYP
;*****
HDRS,BEGIN,NULL ;DBR PROBLEM WITH DBR BITS ASTAT=GOOD, ACSR=BAD
;*****

000542* 005077 177456 2S: CLR @IBS ;CLEAR THE CSR.
;*

;* LOGIC TEST #3
;* IN THIS TEST WE'LL MAKE
;* SURE THE INTERRUPT ENABLE (BIT6)
;* WILL SET AND CLEAR.
;*

000546* 104407 000000* LOG3: BREAKS,BEGIN ;TEMPORARY RETURN TO MONITOR....
000552* 104407 000000* BREAKS,BEGIN ;THEN CONTINUE AT NEXT INSTRUCTION.

000556* 012777 000100 177440 MOV #BIT6,@IBS ;SET INTERRUPT ENABLE.
000554* 032777 000100 177432 BIT #BIT6,@IBS ;DID IT SET?
000572* 001017 BNE 15 ;YES - LETS TRY CLEARING IT.
000574* 017767 177424 177300 MOV @IBS,ACSR ;RECORD CSR FOR ERROR TYPEOUT.
000602* 012767 000100 177274 MOV #BIT6,ASTAT ;RECORD SHOULD BE.
000610* 016767 177410 177262 MOV @IBS,CSRA ;RECORD IBV ADDR.

000616* 012767 000027 177262 MOV #27,ERRTYP ;IE FAILED TO CLEAR


```

460 001176* 1S:
461
462 001176* 000004 000000* 001204*
463 -----
464 PIRQS,BEGIN,2S ; QUEUE UP TO CONTINUE AT 2S AND RTI
465 -----
466 001204* 004767 000212 177010 2S: JSR PC,PATGEN ;GO GET A PATTERN TO SEND
467 001210* 016777 177036 MOV PATTERN,@IBD ;LOAD PATTERN.
468 001216* 104400 000000* EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
469
470
471
472 ;LNRSRV THIS ROUTINE RECEIVES "LNR" INTERRUPTS FROM THE IBV11
473 ; IT WILL DELAY, COMPARE DATA ADMITTED TO DATA RECEIVED,
474 ; IF BAD, WE'LL REPORT AN ERROR, IF GOOD
475 ; IT ALLOWS THE NEXT "TKR" INTERRUPT.
476 ; WE'LL ALSO CHECK FOR "END PASS".
477 001222* 042777 000100 176774 LNRSRV: BIC #100,@IBS ;TMP CLEAR OF INTERRUPT ENABLE.
478 001230* 000004 000000* 001236*
479 -----
480 PIRQS,BEGIN,1S ; QUEUE UP TO CONTINUE AT 1S AND RTI
481 -----
482 001236* 016767 177010 176640 1S: MOV PATTERN,ASTAT ;RECORD CURRENT PATTERN.
483 001244* 017767 176756 176630 MOV @IBD,ACSR ;READ DATA FROM DBR.
484 001252* 042767 177400 176622 BIC #177400,ACSR ;STRIP UNUSED DATA BITS
485 001260* 026767 176620 176614 CMP ASTAT,ACSR ;DATA SENT = DATA RECEIVED?
486 001266* 001411 BEQ 2S ;YES - CONTINUE
487 001270* 016767 176732 176602 MOV IBID,CSRA ;RECORD ADDR. OF DATA REG.
488
489 001276* 012767 000001 176602 MOV #1,ERRTYP
490 001304* 104405 000000* 000000
491 -----
492 HRDRS,BEGIN,NULL ;BAD DATA TRANSFER-ASTAT=DATA SENT ;ACSR=DATA RECEIVED
493 *****
494 001312* 026767 176732 176520 2S: CMP CNT,ICOUNT ;DONE ENOUGH XFRS?
495 001320* 001407 BEQ 3S ;YES - REPORT END PASS
496 001322* 104413 000000* ENDTIS,BEGIN ;SIGNAL END OF ITERATION.
497 001326* 052777 000100 176670 BIS #BIT6,@IBS ;MONITOR SHALL TEST END OF PASS
498 ; NO - RESET INTERRUPT ENABLE - THIS
499 ; WILL CAUSE A "TKR" INTERRUPT
500 ; FOR NEXT DATA.
501 001334* 104400 000000* EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
502
503 001340* 005077 176660 3S: CLR @IBS ;CLEAR THE CSR.
504 001344* 104413 000000* ENDTIS,BEGIN ;SIGNAL END OF ITERATION.
505 ; MONITOR SHALL TEST END OF PASS
506
507
508 ;ERRSRV
509 THIS ROUTINE TAKES CARE OF ERROR INTERRUPTS
510 AS WE'LL AS INTERRUPTS THAT ARE FATAL
511 TO THIS PIECE OF SOFTWARE IT WILL
512 RECORD THE CSR, CLEAR THE CSR - DELAY (VIA
513 A PIRQ) THEN REPORT THE ERROR.
514
515 SINCE THIS TYPE OF ERROR SHOULD NEVER
516 OCCUR, JUST IN CASE YOU GET ONE, WE
517 WON'T DROP THE MODULE. IN THIS WAY, YOU

```

```

516 CAN ACCUMULATE DATA ON THE PROBLEM.
517 THIS POINTS TO SOME DEVICE STILL CONNECTED
518 SEQUENCE FROM THE POINT OF SETTING "TCS"
519 TO GET A "CMD" INTERRUPT.
520
521 TYPES OF ERRORS THAT GOT US HERE AND THEIR
522 CAUSES:
523
524 1. SRQ INTERRUPT (BIT15 SET IN IBS)
525 THIS POINTS TO SOME DEVICE STILL CONNECTED
526 TO THE IB BUS. DISCONNECT THE CABLE FROM
527 THE IBV11. IF YOU STILL GET A SRQ INTERRUPT,
528 YOU'VE GOT A PROBLEM.
529
530 2. ER1 INTERRUPT (BIT13 SET IN IBS)
531 THIS BIT IS FATAL TO THE IBV11 WHEN SET.
532 THE IB BUS IS SCREWED-UP.
533 THERE ARE TWO CAUSES FOR IT:
534 A. ATN, IFC, OR REN ARE ASSERTED ON THE IB BUS
535 BY SOMEONE OTHER THAN THIS IBV11. TRY
536 DISCONNECTING THE IB BUS CABLE FROM THE
537 IBV11 (THE CUSTOMER MAY HAVE SOME MISD
538 OPTION ON THE IB BUS THAT IS IN DIRECT
539 CONFLICT WITH THE IBV11 SPEC.). IF THIS
540 DOESN'T HELP, START LOOKING FOR A PROBLEM
541 IN THE GATES THAT GENERATE AN ER1.
542 B. ATN, IFC, OR REN ARE ASSERTED ON THE IB BUS
543 BY THE IBV11, BUT ARE NOT RECEIVED BACK
544 BY IT. I'VE ALREADY CHECKED THIS CONDITION
545 ONCE IN THE LOGIC TESTS - BUT IT WAS A
546 ONE TIME CHECK AND MAYBE EXERCISING THIS
547 BABY A LOT CAUSED A PROBLEM IN THAT CIRCUITRY.
548
549 NOW TO TRY AND TELL THE DIFFERENCE
550 BETWEEN A AND B:
551
552 IF TCS IS SET IN THE CSR AND ATN IS NOT
553 SET - B IS THE PROBLEM.
554 IF TCS IS CLEAR BUT ATN IS SET STEP A IS
555 THE PROBLEM.
556 NOW I DON'T USE REN OR IRC IN THE DATA
557 TRANSFERS SO STEP A IS MOST LIKELY TO
558 BE THE PROBLEM - EXAMINE THE CSR TO SEE IF
559 EITHER OR BOTH ARE SET.
560
561
562 3. ER2 INTERRUPT (BIT14 SET)
563 ERROR 2'S ARE CAUSED BY A TALKER ON
564 THE BUS AND NO LISTENERS.
565
566 I DON'T LET THAT HAPPEN AS I MAKE THE
567 IBV11 A LISTENER AS WELL AS A TALKER.
568 YOU BETTER LOOK BACK AT THE CIRCUITRY THAT
569 GENERATES AN ERROR 2 - ITS PAD!
570
571

```


LNRSRV	001222R	425	476#																	
LOG1	000414R	275#																		
LOG2	000420R	300#																		
LOG3	000546R	333#																		
LOG4	000676R	379#																		
MAP22s=	104416	180#																		
MODNAM	000000R	131#																		
MODSP	000224R	140#	178#																	
MSGSCS	= 104403	180#																		
MSGSS	= 104402	180#																		
MSGSS	= 104401	180#																		
NULL	= 000000	180#																		
OPEN	= 000000	167	312	324	349	360	400	414	490	583	162	163	164	165						
OTDAS	= 104420	180#	169	171	172	174	175	176	180#											
PASCNT	000034R	149#																		
PATERN	000252R	180#																		
PATGEN	001422R	466#	481	599*	606*	608*														
PIRGS	= 000004	180#	462#	478	576															
POPSP	= 005726	180#																		
POPSP2	= 022629	180#																		
PRA	000232R	187#	218*	428*																
PRB	000236R	189#	222*	430*																
PRC	000240R	189#	226*	424*																
PRD	000246R	193#	230*	426*																
PRTV	= 000000	180#																		
PRTV0	= 000000	180#																		
PRTV1	= 000040	180#																		
PRTV2	= 000100	180#																		
PRTV3	= 000140	180#																		
PRTV4	= 000200	180#																		
PRTV5	= 000240	180#																		
PRTV6	= 000300	180#	136	180#																
PRTV7	= 000340	180#																		
PS	= 177776	180#																		
PSW	= 177776	180#																		
PUSH	= 005746	180#																		
PUSH2	= 024646	180#																		
RANA	000254R	203#	602*	603*	604*	606														
RANB	000256R	204#	602*	604*	605*															
RANDS	= 104441	180#																		
RANNUM	000054R	154#																		
RESTR1	001060R	173#	420#																	
RES1	000066R	152#																		
RES2	000066R	152#																		
RSTRT	000112R	175#																		
SBADR	000102R	166#																		
SOPCNT	000042R	149#																		
SOPFRS	= 104406	151#																		
SOPPAS	000046R	151#																		
SPOINT	= 000032R	145#																		
SPSTZ	= 000040	145#																		
SR1	= 000016R	138#	178	597																
SR2	000020R	139#																		
SR3	000022R	140#																		
SR4	000024R	141#																		

START	000260R	144	208#																	
STAT	000026R	143#																		
SVR0	000062R	158#																		
SVR1	000064R	159#																		
SVR2	000066R	160#																		
SVR3	000070R	161#																		
SVR4	000072R	162#																		
SVR5	000074R	163#																		
SVR6	000076R	164#																		
SYS CNT	000052R	153#																		
TKRSV	001156R	423#	453#																	
TRPDFD	= 000052R	186#																		
VECTA	000230R	186#	216*	427*																
VECTB	000234R	188#	220*	429*																
VECTC	000240R	190#	224*	423*																
VECTD	000244R	192#	228*	425*																
VECTOR	000010P	134#	215																	
WASADR	000104R	168#																		
WDR	000116R	174#																		
WDTO	000114R	173#																		
XFLAG	000005R	132#																		

ARS. 000000 000
 001502 001

ERRORS DETECTED: 0
 DEFAULT GLOBALS GENERATED: 0
 XIBADO, XIBADO/SOL/CRF:SYN=DDXCON, XIBADO
 RUN-TIME: 1 2 2 SECONDS
 RUN-TIME RATIO: 15/4=3.7
 CORE USED: 7K (13 PAGES)