

.REM -

SEQ 0001

IDENTIFICATION

PRODUCT CODE: AC-E688G-MC
PRODUCT NAME: CXKWAGO DEC/X11 KW11-L LINE CLOCK MODULE
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) 1973,1978 DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT

THE KWA IS AN IOMOD THAT EXERCISES THE KW11-L LINE
FREQUENCY CLOCK.
IT IS ALSO USED AS THE SYSTEM CLOCK BY THE MONITOR.

2. REQUIREMENTS:

HARDWARE: ONE KW11-L OPTION MODULE
SOFTWARE: CAN BE USED AS A SYSTEM CLOCK

STORAGE:: KWA REQUIRES:

- 1: DECIMAL WORDS: 467
- 2: OCTAL WORDS: 0723
- 3: OCTAL BYTES: 1646

3. PASS DEFINITION

ONE PASS CONSISTS OF 3600. INTERRUPTS AT 60HZ OR 3000. AT 50HZ
EXECUTION TIME

THE KWA RUNNING ALONE ON A PDP11/05 PROCESSOR TAKE
APPROXIMATELY 1 MINUTE

5. CONFIGURATION REQUIREMENTS

DEFAULT PARAMETERS:

DEVADR: 177564, VECTOR: 100, BR1: 6, DEVCNT: 1, SR1:0

REQUIRED PARAMETERS:

NONE

6. DEVICE/OPTION SETUP:

NONE

7. MODULE OPERATION:

TEST SEQUENCE:

- A. SET UP DEVICE ADDRESS AND VECTOR, ANY TIME MESSAGES, AND 50 OR 60 HZ.
- B. ENABLE INTERRUPTS
- C. COUNT THE INTERRUPTS 50/60. INTERRUPTS = 1 SECOND
- D. REPEAT B THRU C 360. TIMES (60HZ) OR 3000. TIMES (50HZ)
- E. REPORT ENDPAS BUT KEEP CLOCK RUNNING
- F. AT RESTART, CHECK IF IT'S TIME FOR MESSAGE
- G. CONTINUE AT C

SINCE THE KW11L IS A GO/NO GO TYPE OF OPTION THERE ARE NO ERROR PRINTOUTS. FAILURE TO GET AN END OF PASS REPORT INDICATES NO GO.

IF THE CLOCK HAS NOT BEEN RESTARTED FROM AN ENDPAS CALL AFTER 15. MINUTES HAVE ELAPSED, IT WILL MAKE A JMP INTO THE MONITOR (CKHUNG) TO CHECK FOR ANY HUNG MODULES.

8. OPERATION OPTIONS:

VALID SRI VALUES:

SRI	HERTZ	TIME MSG EVERY X MINUTES
0	60	NEVER
1	50	NEVER
2	60	5
3	50	5
4	60	15
5	50	15
6	60	60
7	50	60

; ALL TIMES ARE APPROXIMATE

9. NON-STANDARD PRINTOUTS

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

FKW11L DEC/X11 EXERCISER MODULE

```

000000 IOMOD <KWAG> 177546,100,6,50,,11
000000 MODULE 140000 KWAG,177546,100,6,50,,11
          TITLE KWAG DEC/X11 SYSTEM EXERCISER MODULE
          DDXCOM VERSION 6 23-MAY-78
          .LIST BIN
*****
000000 BEGIN:
000000 053513 043501 040 MODNAM: .ASCII /KWAG / ;MODULE NAME
000005 .000 XFLAG: .BYTE OPEN ;USED TO KEEP TRACK OF WBUFF USAGE
000006 177546 ADDR: 177546+0 ;1ST DEVICE ADDR.
000010 000100 VECTOR: 100+0 ;1ST DEVICE VECTOR.
000012 .300 BR1: .BYTE PRTV6+0 ;1ST BR LEVEL.
000014 .000 BR2: .BYTE PRTV+0 ;2ND BR LEVEL.
000016 000001 DVID1: +1 ;DEVICE INDICATOR 1.
000020 000000 SR1: OPEN ;SWITCH REGISTER 1
000022 000000 SR2: OPEN ;SWITCH REGISTER 2
000024 000000 SR3: OPEN ;SWITCH REGISTER 3
000024 000000 SR4: OPEN ;SWITCH REGISTER 4
*****
000026 140000 STAT: 140000 ;STATUS WORD
000030 000224 INIT: START ;MODULE START ADDR.
000032 000224 SPOINT: MODSP ;MODULE STACK POINTER.
000034 000000 PASCH: 0 ;PASS COUNTER.
000036 000062 ICONF: 50. ;# OF ITERATIONS PER PASS=50.
000040 000000 ICOUNT: 0 ;LOC TO COUNT ITERATIONS
000042 000000 SDFCMT: 0 ;LOC TO SAVE TOTAL SOFT ERRORS
000044 000000 HRDCHT: 0 ;LOC TO SAVE TOTAL HARD ERRORS
000046 000000 SOFPAS: 0 ;LOC TO SAVE SOFT ERRORS PER PASS
000050 000000 HRDPAS: 0 ;LOC TO SAVE HARD ERRORS PER PASS
000052 000000 SYSCNT: 0 ;# OF SYS ERRORS ACCUMULATED
000054 000000 RANUM: 0 ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
000056 000000 CONFCG: 0 ;RESERVED FOR MONITOR USE
000058 000000 RES1: 0 ;RESERVED FOR MONITOR USE
000060 000000 RES2: 0 ;RESERVED FOR MONITOR USE
000062 000000 SVR0: OPEN ;LOC TO SAVE R0.
000064 000000 SVR1: OPEN ;LOC TO SAVE R1.
000066 000000 SVR2: OPEN ;LOC TO SAVE R2.
000068 000000 SVR3: OPEN ;LOC TO SAVE R3.
000070 000000 SVR4: OPEN ;LOC TO SAVE R4.
000072 000000 SVR5: OPEN ;LOC TO SAVE R5.
000074 000000 SVR6: OPEN ;LOC TO SAVE R6.
000076 000000 CSRA: OPEN ;ADDR OF CURRENT CSR.
000102 000000 SBADR: OPEN ;ADDR OF GOOD DATA, OR
000104 000000 WASADR: OPEN ;ADDR OF BAD DATA, OR
000106 000000 ASTAT: OPEN ;CONTENTS OF CSR.
000108 000000 ERRDTP: OPEN ;STATUS REG CONTENTS.
000110 000000 ASB: OPEN ;TYPE OF ERROR.
000112 000000 AWAS: OPEN ;EXPECTED DATA.
000114 000000 RSTRT: RSTRT ;ACTUAL DATA.
000116 000000 MDTY: OPEN ;RESTART ADDRESS AFTER END OF PASS
000120 000000 WDFR: OPEN ;WORDS TO MEMORY PER ITERATION
          INTR: OPEN ;# OF INTERRUPTS PER ITERATION

```

```

000122 000011 IDNUM: 11 ;MODULE IDENTIFICATION NUMBER=11
          000040 .REPT SPSIZ ;MODULE STACK STARTS HERE.
          .NLIST
          .WORD 0
          .LIST
          .ENDR
000224 MODSP:
*****

```

```

199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254

```

.GLOBL KW11L,HMS,CLOCK,CKHUNG,LCLEAR
START: MOV #50,ICONT ;NEEDED IN CASE OF RESTART
TST CLOCK ;ANY TIMING ANALYSIS ERROR?
BNE IS ;BR IF NOT
ENDS,BEG ;POWER FAIL RESTART?
1\$: TST PASCNT ;CLOCK NOT AVAILABLE, DROP IT
BNE 2\$;YES, DON'T ZERO TIME INFO
JSR PC,LCLEAR ;GO CLEAR PASS TIME TABLE
MOV #1,RUNING ;SHOW WE ARE IN A PASS
MOV #1,R1 ;WHAT IS MSG INTERVAL?
ASR R1 ;GET RID OF 50/60 HZ BIT
MOV #300,INTER ;SETUP FOR 5 MIN. MESSAGES
MOV #1,R1 ;DOES HE WANT 5 MIN. MESSAGES?
REQ #5 ;BR IF YES
ADD #600,INTER ;NO ADD 10 MORE MINUTES
CMP #2,R1 ;IS IT 15 MINUTES?
REQ #3 ;BR IF YES
ADD #2700,INTER ;NO, MUST BE 60 MINUTES
MOV INTER,MSGTIM ;SO ADD 45 MORE MINUTES
CLR #TIME ;TIME IS COUNT FOR WHEN TO TYPE MSG
MOV ADDR,R1 ;GET DEVICE ADDRESS
MOV ADDR,CLKCSR ;AND PUT IT IN CLKCSR
MOV VECTOR,RO ;GET VECTOR
MOV #CLKINT,(RO)+ ;AND STORE ITS ADDRESS THERE
MOV #56,BASE ;SET UP THE BR LEVEL
MOV #1,SRI ;ASSUM IT'S A 50HZ MACHINE
BIT #45 ;IS IT REALLY?
BNE 4\$;BR AROUND IF YES, ELSE
ADD #10,BASE ;ADD 10 MORE TO = 60HZ
MOV #1,ICONT ;FOR 60 CPS,ICONT=60
4\$: MOV BASE,TIMCTR ;COUNT TICS TO GET SECONDS
MOV ICONT,INTR ;ICONT WILL BE # OF INTERRUPTS
CLKGO: MOV BASE,CLKCTR ;COUNT TICS TO GET SECS FOR ENDPASS
MOV #BIT6,@CLKCSR ;START THE CLOCK GOING
EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
;INTERUPT SERVICE ROUTINE, EVEN WHEN MODULE IS BETWEEN PASSES
KLKINT: CLR @CLKCSR ;DISABLE INTERRUPTS
DEC TIMCTR ;KEEP TRACK OF TICS FOR MSG
BNE 1\$;BR IF NOT 1 SECOND ELAPSED
ADD #1,TIME ;COUNT A SECOND IN TOTAL TIME
INC TIMEXT ;EXTENDED BITS FOR THE TIME WORD
MOV #1,ICONT ;COUNT A SECOND FOR MSG TIME
MOV #1,ICONT ;COUNT TICS AGAIN FOR NEXT SECOND
TST RUNING ;ARE WE BETWEEN PASSES?
BNE 1\$;NO, CONTINUE
MOV #TIME,HDPTIM ;GET THE PRESENT TIME
SUB #1,HDPTIM ;GET ELAPSED TIME SINCE LAST EOP
CMP #900,HDPTIM ;HAS 15 MINUTE LIMIT BEEN PASSED?
BNE 1\$;NO, CONTINUE
MOV #900,HDPTIM ;YES, CHECK AGAIN AFTER 15. MINUTES
ADD #1,HDPTIM ;CHECK FOR ANY "HUNG" MODULES
BR 3\$

```

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
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310

```

1\$: TST RUNING ;ARE WE BETWEEN PASSES?
DEC CLKCTR ;BR IF NOT
BEQ 2\$;IF COUNTED 60, GO ADD A SEC
MOV #BIT6,@CLKCSR ;START CLOCK OFF AGAIN
RTI
3\$:
;IRQS,BEGIN,4\$; QUEUE UP TO CONTINUE AT 4\$ AND RTI
4\$: MOV #BIT6,@CLKCSR ;TURN THE CLOCK BACK ON
MOV #56,(SP) ;SAVE R5
MOV #56,R5 ;SET UP R5 STACK
JSR PC,CKHUNG ;SET UP MONITOR. CHECK FOR HUNG MODULES
MOV #56,(SP)+,R5 ;RESTORE R5
EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
5\$:
;IRQS,BEGIN,6\$; QUEUE UP TO CONTINUE AT 6\$ AND RTI
6\$: MOV BASE,RO ;GET BASE INTO RO
DEC RO ;SET UP RO
CMP #COUNT,RO ;IS IT TIME FOR END OF PASS??
BNE 7\$;BRANCH IF NOT
CLR RUNING ;CLEAR BETWEEN PASS FLAG
MOV #TIME,HDPTIM ;SAVE THE TIME OF EOP
7\$: MOV #BIT6,@CLKCSR ;KEEP CLOCK GOING
ENDITS,BEGIN ;SIGNAL END OF ITERATION.
BR CLKGO ;MONITOR SHALL TEST END OF PASS
; NO, KEEP ON TICKIN*
RESTRT:
8\$: TST PASCNT ; THIS IS FOR CSS BUS SWITCHES
BNE REST ; CONTINUE
JMP START ; BEGIN AT START
CMP #TIME,INTER ; IS IT TIME FOR A MSG?
BNE 9\$; NO, BRANCH
CLR #TIME ; YES, RESET MSG TIME COUNT
CMP #1,R1 ; ARE ANY TIME MSG'S WANTED
BLT 1\$; BR IF NOT
;+
; SET UP R5 STACK AND CONVERT TIME AND TIMEXT TO HOURS, MINUTES AND SECONDS.
;- CALLING SEQUENCE: CALL HMS IN <TIME,TIMEXT> OUT <RO>
MOV RO,-(SP) ;SAVE RO
MOV #56,(SP) ;SAVE R5
MOV #56,R5 ;SET UP R5 STACK
SUB #1,#2,R5 ;SAVE SPACE ON STACK FOR OUTPUT

```
311 000760 010546          MOV R5,-(SP)          ;ARGUMENT FOR HMS CALL
312 000762 016745 000224  MOV TIMEXT,-(R5)     ;SAVE R5 STACK POINTER ON R6 STACK
313 000764 016745 000216  MOV TIME,-(R5)       ;PLACE EXTENDED ELAPSED TIME
314 000772 004767 000000G JSR PC,HMS           ;ARGUMENT ON R5 STACK
315 000776 012605          MOV (SP)+,R5         ;PLACE ELAPSED TIME ARGUMENT
316 001000 012800          MOV (R5)+,R0         ;ON R5 STACK
317 001002 112067 000155  MOV (R0)+,RUNT1+1   ;CALL HOURS, MINUTES, SECOND CONVERSION
318 001006 112067 000152  MOV (R0)+,RUNT1+2   ;RESTORE R5 STACK POINTER
319 001012 112067 000147  MOV (R0)+,RUNT1+3   ;GET OUTPUT ARGUMENT OF HMS
320 001016 112067 000145  MOV (R0)+,RUNT2     ;ROUTINE INTO R0
321 001022 112067 000142  MOV (R0)+,RUNT2+1   ;LOAD
322 001028 112067 000140  MOV (R0)+,RUNT3     ;CONVERTED
323 001032 112067 000135  MOV (R0)+,RUNT3+1   ;TIME
324 001036 112605          MOV (SP)+,R5         ;INTO
325 001040 012600          MOV (SP)+,R0         ;MESSAGE
326 001042 104403          MSGS,BEGIN,EXPRI    ;RESTORE R5
327 001050 012767 000001  MOV #,RUNT3         ;RESTORE R0
328 001056 000167 177354  JMP CLKCO           ;ASCII MESSAGE CALL WITH COMMON HEADER
329                                     ;START COUNTING SEC.S FOR PASSES AGAIN
330
331
332
333
```

```
334 001062 012700 001214  LCLEAR: MOV #MODTIM,R0 ;GET BEGINING OF TABLE
335 001066 012701 001574  MOV #MODEND,R1      ;GET END
336 001072 005067 000112  CLR TIME           ;ZERO TOTAL RUNTIME
337 001076 005067 000110  CLR TIMEXT        ;ZERO EXTENDED TIME BITS
338 001102 005020          CLR (R0)+          ;CLEAR ENTRY
339 001104 020001          CMP R0,R1         ;DONE?
340 001106 001375          BNE IS            ;BR IF NO
341 001110 000207          RTS PC           ;EXIT
342
343
344 001112 001142  EXPRI: RUNT1          ; RUNTIME MESSAGE POINTER
345 001114 177777          MSGTIM: OPEN      ; TELLS WHEN TO TYPE A MSG
346 001116 000000          INTER: OPEN     ; HOW OFTEN BETWEEN MSG.S
347 001122 000000          BASE: OPEN      ; HOLDS SQ. FOR 50HZ, 60. FOR 60HZ
348 001124 000000          CLKCSR: OPEN    ; HOLDS ADDRESS OF CLOCK'S CSR
349 001126 000000          CLKCTR: OPEN    ; NUMBER OF INTERRUPTS LEFT BETWEEN SECONDS
350 001128 000000          RUNING: OPEN    ; 0 IF BETWEEN PASSES
351 001130 000000          RTIME: OPEN     ; COUNTS SECONDS BETWEEN MSG.S
352 001132 000000          TIMCTR: OPEN    ; COUNTS TIC'S FOR MSG SECONDS
353 001134 000000          TMTPTM: OPEN   ; TEMP LOCATION HOLDS TIME
354 001136 000000          HLDPTM: OPEN   ; TEMP LOCATION HOLDS TIME SINCE LAST EOP
355 001142 020045 051040 047125 RUNT1: .ASCII "% RUN TIME IS "
356 001150 052040 046511 020105
357 001156 051511 020040
358 001162 020040 072      RUNT1: .ASCII " : " ; HOURS
359 001167 040      035040  RUNT2: .ASCII " : " ; MINUTES
360 001172 020040 024040 035110  RUNT3: .ASCII " (H:M:S)%" ; SECONDS
361 001200 035115 024523 000045
362
363
364                                     ;
365                                     ; DO NOT MODIFY ANY LOCATIONS BELOW THIS POINT.
366                                     ; THE MONITOR MUST KNOW EXACTLY WHERE EACH WORD IS.
367
368
369 001206 000000  KW11L: .EVEN
370 001210 000000  TIME: OPEN          ; HOLDS STARTING ADDRESS OF MODULE
371 001214 000000  TIMEXT: OPEN        ; ELAPSED TIME IN SECONDS
372 001216 000000  MODTIM: OPEN        ; HOLDS EXTENDED BITS FOR THE TIME WORD
373 001218 000170  MODEND: .BLKW 120.  ; TABLE FOR MONITOR, TWO WORDS FOR EACH MODULE
374
375
376
377
378
379 ;+
380 ; SET ASIDE R5 STACK STORAGE
381 ;-
382
383 001644 000000  R5STK: .WORD 0      ;R5 STACK
384 000001          .END
```


SR2	000020R	152#				
SR3	000022R	153#				
SR4	000024R	154#				
START	000224R	155#	202#	295		
STAT	000026R	156#				
SVR0	000062R	171#				
SVR1	000064R	172#				
SVR2	000066R	173#				
SVR3	000070R	174#				
SVR4	000072R	175#				
SVR5	000074R	176#				
SVR6	000076R	177#				
SYSCNT	000052R	166#				
TACTR	001134R	231#	241*	246*	355#	
TIME	001210R	243#	249	265	315	
TINEXT	001212R	244#	313	339*	343#	338*
INPTIN	001136R	249#	250*	251	356#	372#
TRPDFD	000022	147#				
VECTDR	000010R	148#				
WASADR	000104R	181#	223			
WDFR	000116R	188#				
WDT0	000114R	187#				
XFLAG	000005R	145#				
.	= 001646R	374#				

- ABS. 000000 000
001646 001

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0
XKWAGO XKWAGO/SOL/CRF:SYN=DDXCON,XKWAGO
RUN-TIME: 1 1 2 SECONDS
RUN-TIME RATIO: 13/2=4.5
CORE USED: 7K (13 PAGES)