

.REM _

IDENTIFICATION

PRODUCT CODE: AC-E794E-MC
PRODUCT NAME: CKAF AEO AFC11 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) 1973,1978 DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT:

AFA IS AN IOMOD THAT EXERCISES AN AFC11 LOW LEVEL MULTIPLEXER. IT PERFORMS A WAS/IS TEST ON CHANNELS ZERO THRU SEVEN, BY LOADING THE CONVERSION RESULTS OF EACH CHANNEL INTO TWO TABLES AND COMPARING THE RESULTS STORED IN THE TABLES. IF THE WAS/IS READINGS ARE NOT WITHIN PLUS OR MINUS 20 IN THE LSB AN ERROR IS REPORTED. THE CONVERSIONS ARE DONE AT A GAIN OF 1000 SO EACH CHANNEL MUST HAVE A CONSTANT DC INPUT LESS THAN 5MV.

2. REQUIREMENTS:

HARDWARE: ONE AFC11 CONVERTER

STORAGE:: AFA REQUIRES:

1. DECIMAL WORDS: 272
2. OCTAL WORDS: 0420
3. OCTAL BYTES: 1040

3. PASS DEFINITION

ONE PASS OF THE AFA MODULE CONSISTS OF READING THE RESULTS OF 3,072. CONVERSIONS WHICH RESULTS IN 3072 WORDS TRANSFERRED ON THE UNIBUS.

4. EXECUTION TIME

AFA RUNNING ALONE ON A PDP11/05 PROCESSOR TAKES APPROXIMATELY ONE MINUTE TO COMPLETE ONE PASS.

5. CONFIGURATION REQUIREMENTS:

DEFAULT PARAMETERS:

DEVADR: 172570, VECTOR: 134, BR1: 5, DEVCNT: 1

REQUIRED PARAMETERS:

NONE

6. DEVICE/OPTION SET-UP:

ALL CHANNELS MUST HAVE A DC INPUT LESS THAN 5 MILLIVOLTS.

7. MODULE OPERATION:

TEST SEQUENCE:

- A. SET UP VECTOR AND ADDRESS POINTERS
- B. MAKE ONE CONVERSION ON EACH CHANNEL (0-7)
AND STORE READINGS IN TABLE 1
- C. REPEAT B BUT STORE READINGS IN TABLE 2
- D. COMPARE DATA IN TABLE 1 WITH TABLE 2 AND REPORT
ERRORS - READINGS MUST BE WITHIN + OR - 20 IN LSB.
- E. REPEAT B-D 192 TIMES
- F. REPORT END OF PASS - RESTART AT B

8. OPERATION OPTIONS:

9. NON STANDARD PRINTOUT:

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

FAFC11 DEC/X11 EXERCISER MODULE

```

000000- IOMOD <AFAP > 172570,134,5,192,51
000000- MODULE 140000,AFAP,172570,134,5,192,51
; DDACOM VFRSION 6 23-MAY-78
-LIST BIN
*****
000000- ;BEGIN: ASCII (AFAP / ;MODULE NAME
000005- 043101 042501 040 XFLAG: .RYTE OPEN ;USED TO KEEP TRACK OF WBUFF USAGE
000006- 000 ADDR: 172570+0 ;1ST DEVICE ADDR.
000010- 000134 VECTOR: 134+0 ;1ST DEVICE VECTOR.
000012- 240 BR1: .RYTE PRTY5+0 ;1ST BR LEVEL.
000014- 000 BR2: .RYTE PRTY+0 ;2ND BR LEVEL.
000016- 000000 DFD1: .RYTE ;DEVICE INDICATOR 1.
000020- 000000 SR1: OPEN ;SWITCH REGISTER 1
000022- 000000 SR2: OPEN ;SWITCH REGISTER 2
000024- 000000 SR3: OPEN ;SWITCH REGISTER 3
000026- 000000 SR4: OPEN ;SWITCH REGISTER 4
*****
000026- 140000 STAT: 140000 ;STATUS WORD.
000030- 000232- INIT: START ;MODULE START ADDR.
000032- 000224- SPOINT: MODSP ;MODULE STACK POINTER.
000034- 000000 PASCNT: 0 ;PASS COUNTER.
000036- 000300 ICOUNT: 0 ;# OF ITERATIONS PER PASS=192.
000040- 000000 SOFCNT: 0 ;LOC TO COUNT ITERATIONS
000042- 000000 HRDCNT: 0 ;LOC TO SAVE TOTAL SOFT ERRORS
000044- 000000 SOFPAS: 0 ;LOC TO SAVE TOTAL HARD ERRORS
000046- 000000 HRDPAS: 0 ;LOC TO SAVE SOFT ERRORS PER PASS
000050- 000000 SVSCNT: 0 ;LOC TO SAVE HARD ERRORS PER PASS
000052- 000000 RANUM: 0 ;# OF SVS ERRORS ACCUMULATED
000054- 000000 CONFCG: 0 ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
000056- 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.
000070- 000000 SVR3: OPEN ;LOC TO SAVE R3.
000072- 000000 SVR4: OPEN ;LOC TO SAVE R4.
000074- 000000 SVR5: OPEN ;LOC TO SAVE R5.
000076- 000000 SVR6: OPEN ;LOC TO SAVE R6.
000100- 000000 CSRA: OPEN ;ADDR OF CURRENT CSR.
000102- 000000 SRAADR: OPEN ;ADDR OF GOOD DATA, OR
000104- 000000 ACSR: OPEN ;CONTENTS OF CSR.
000106- 000000 WASADR: OPEN ;ADDR OF BAD DATA, OR
000108- 000000 ERRSTYP: UPEN ;STATUS REG CONTENTS.
000106- 000000 ASB: OPEN ;EXPECTED DATA.
000110- 000000 AVAS: OPEN ;ACTUAL DATA.
000112- 000246 RSTRT: RSTRT ;RESTART ADDRESS AFTER END OF PASS
000114- 000000 WDTO: OPEN ;WORDS TO MEMORY PER ITERATION
000120- 000000 INTR: OPEN ;WORDS FROM MEMORY PER ITERATION
;# OF INTERRUPTS PER ITERATION
  
```

```

000122- 000051 IDNUM: 51 ;MODULE IDENTIFICATION NUMBER=51
000040 000040 ;MODULE STACK STARTS HERE.
;REPT SPSIZ
;WORD 0
;LIST
;LIST
;ENDR
000224- MODSP:
*****
186 AFCSR: OPEN ;CONTAINS ADDRESS OF THE CSR
187 AFDBR: OPEN ;CONTAINS ADDRESS OF THE DBR
188 APCAR: OPEN ;CONTAINS ADDRESS OF CHAN. ADDR. REG.
189
190 ;MODULE INITIALIZATION
191
192
193 000232- 012767 000020 177654 START: MOV #16, WDTO ;16 WDS TO MEM PER ITERATION
194 000240- 012767 000020 177652 MOV #16, INTR ;16 INTERRUPTS PER ITERATION
195
196 000246- 016705 177534 RSTRT: MOV ADDR, R5 ;GET THE FIRST ADDRESS
197 000252- 010567 177746 MOV R5, AFCSR ;BUILD ALL THE DEVICE ADDRESSES
198 000256- 005725 TST (5)+
199 000260- 010567 177742 MOV R5, AFDBR
200 000264- 005725 TST (5)+
201 000266- 010567 177736 MOV R5, APCAR
202 000272- 016700 177512 MOV VECTOR, R0 ;GET THE VECTOR ADDRESS
203 000276- 012720 000374- MOV #AFC11, (0)+ ;POINT INTERRUPTS TO AFC11
204 000302- 116720 177504 MOV# BR1, (0)+ ;SET UP THE PRIORITY LEVEL
205 000306- 012703 001000- ST1: MOV #AFTAB1, R3
206 000312- 005067 000460 CLR APPLG ;CLEAR TABLE PASS FLAG
207 000316- 012701 177770 SRTPUP: MOV #10, R1 ;SET UP TO CLEAR DATA TABLES
208 000322- 010302 MOV R3, R2
209 000324- 005022 15: CLR (R2)+
210 000326- 005201 INC R1
211 000330- 002775 BLT 15 ;STAY IN LOOP TIL TABLE CLEAR
212 000332- 010302 MOV R3, R2 ;RESET THE TABLE POINTER
213 000334- 112767 177770 MOV# #10, AFCTR1 ;SAMPLE COUNTS FOR STORAGE OF
214 000342- 116767 000422 000421 40V# AFCTR1, AFCTR2 ;INDIVIDUAL SAMPLES AND AVERAGES FOR
215 ;EACH CHANNEL TESTED
216 ;TO ENABLE GENERATING TABLE OF AVGS.
217 000350- 005067 000420 CLR AFSEL ;SELECT CHN. 0 AT GAIN=X1000
218 000354- 012767 000100 MOV #100, AFCSR ;SET INTERRUPT ENABLE
219 000362- 016777 000406 177640 MOV AFSEL, APCAR ;START CONVERSION
220 000370- 104400 000000- EXITS, BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
221
222 ;INTERROPT SERVICE ROUTINES
223
224 AFC11:
225 -----
226 000374- 000004 000000- 000402- PTRQS, BEGIN, 15 ; QUEUE UP TO CONTINUE AT 15 AND RTI
227 -----
228 000402- 016700 177616 15: MOV AFCSR, R0 ; GET CSR ADDRESS
229 000406- 105710 RSTR (0) ; WAS DONE SET ??
230 000410- 100414 BRCH 25 ; BRANCH IF YES
231 000412- 017767 177606 177462 MOV @AFCSR, ACSR ; SAVE THE CONTENTS OF THE CSR IN ACSR
232 000420- 016767 177600 177452 MOV AFCSR, CSRA ; SAVE THE ADDRESS OF THE CSR IN CSRA
233 000426- 012767 000011 177452 MOV #11, ERRSTYP ; ILLEGAL INTERRUPT OR DONE NOT SET
  
```

```

234
235 000434 104405 000000 000000
236
237 000442 067712 177560
238 000446 105267 000316
239 000452 002424
240 000454 006212
241 000456 006212
242 000460 006222
243 000462 005267 000306
244 000466 105267 000277
245 000472 002411
246 000474 005767 000276
247 000500 001016
248 000502 005167 000270
249 000506 012703 001020
250 000512 000167 177600
251 000516 112767 177770
252 000524 016777 000244
253 000532 104400 000000 000244 177476
254
255 000536 012767 001000 000220
256 000544 112767 177770 000216
257 000552 012767 001020 000206
258 000560 017767 000200 000204
259 000566 167767 000174 000176
260 000574 100402
261 000578 005487 000020
262 000582 002022
263 000610 002022
264 000612 017767 000146 177270
265 000620 017767 000142 177260
266 000626 016767 177372 177244
267 000634 017767 177364 177240
268 000642 012767 000026 177236
269
270 000650 104405 000000 000000
271
272 000656 062767 000002 000100
273 000664 062767 000002 000074
274 000672 105267 000072
275 000676 002730
276 000700 012767 001000 000056
277 000706 012767 001020 000052
278 000714 112767 177770 000046
279 000722 017767 000040 000034
280 000730 062767 000002 000026
281 000736 062767 000002 000022
282 000744 105267 000020
283 000750 002764
284 000752 005077 177246
285 000756 104413 000000
286
287 000762 000651
288
289 000764 000000

```

```

*****
HRDERS,REGIN,NULL ;DONE NOT SET
*****
2S: ADD @AFDDBR,@R2 ;GENERATE TOTAL
INCR AFCTR1 ;ALL CHANNELS DONE ??
BLT AFEXIT ;BRANCH IF NOT
ASR @R2 ;DIVIDE THE TOTAL BY 8
ASR (R2)+
INC AFSEL ;ADVANCE TO THE NEXT CHANNEL
INCR AFCTR2 ;ALL CHANNELS DONE ??
BLT AFEXA ;BRANCH IF NOT
TST APPFLG ;YES, FIRST TABLE ?
BNE AFCDM ;NO COMPARE RESULTS
COM APPFLG ;CHANGE THE FLAG FO 2ND TABLE
AFSP2: MOV #AFTAB2,R3
JMP SETUP
AFEXA: MOV# -10,AFCTR1 ;SET UP FOR 8 SAMPLES PER LINE
AFEXIT: MOV #AFSFL,@AFCAR ;START THE MULTIPLEXOR
EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.

AFCDM: MOV #AFTAB1,AFPTR ;SET UP TO COMPARE THE TABLES
MOV# -10,AFCTR1 ;8 CHANNELS
MOV #AFTAB2,AFPTRA ;TO BE COMPARED
1S: MOV @AFPTR,AFTEMP ;GET THE DATA
SUB @AFPTRA,AFTEMP ;COMPARE ??
BWI 2S ;GET THE ABSOLUTE DIFFERENCE
NEG AFTEMP ;COMPARE AGAINST ALLOWED LIMIT
ADD #20,AFTEMP ;BRANCH IF OK
BGE 3S
MOV @AFPTR,AWAS ;AWAS AND ASB CONTAIN THE TWO READINGS
MOV @AFPTRA,ASB ;DIFFERENCE IS THE ERROR
MOV @AFCSR,CSRA ;SAVE THE CSR ADDRESS
MOV @AFCSR,ACSR ;SAVE THE CONTENTS OF THE CSR
MOV #26,ERRTYP ;A-D CONVERSION OUT OF TOLERANCE
*****
HRDERS,REGIN,NULL ;DIFFERENCE OUT OF LIMITS
*****
3S: MOV #2,AFPTR ;UPDATE THE POINTERS
ADD #2,AFPTRA
INCR AFCTR1 ;BUMP THE COUNTER
BLT 1S ;GET THE NEXT TWO READINGS
AFMOV: MOV #AFTAB1,AFPTR ;REPLACE OLD TABLE OF DATA
MOV #AFTAB2,AFPTRA ;WITH NEW DATA
MOV# -10,AFCTR1
1S: MOV @AFPTRA,@AFPTR
ADD #2,AFPTR
ADD #2,AFPTRA
INCR AFCTR1
BLT 1S
CLR @AFCSR ;TURN OFF INTERRUPTS
ENDITS,BEGIN ;SIGNAL END OF ITERATION.
;MONITOR SHALL TEST END OF PASS
;GET SOME MORE DATA
2S: BR AFSP2
AFPTR: OPEN

```

```

290 000766 000000
291 000770 000
292 000771 000
293
294 000772 000000
295 000774 000000
296 000776 000000
297 001000 000000
298 001020
299 001040
300
301 000001

```

```

AFPTRA: OPEN
AFCTR1: .BYTE 0
AFCTR2: .BYTE 0
-EVEN
AFTEMP: OPEN
AFSEL: OPEN
APPFLG: OPEN
AFTAB1: OPEN
--*16
AFTAB2: OPEN
--*16
-END

```


ST1	000306R	205#	
SVR0	000062R	159#	
SVR1	000064R	160#	
SVR2	000066R	161#	
SVR3	000070R	162#	
SVR4	000072R	163#	
SVR5	000074R	164#	
SVR6	000076R	165#	
SVSCNT	000052R	154#	
TRPDFD=	000022	186#	
VECTOR	000010R	135#	202
WASADR	000104R	169#	
WDFR	000116R	176#	
WDTO	000114R	175#	193*
XFLAG	000005R	133#	
.	= 001040R	298#	300#

. ABS. 000000 000
001040 001

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

XAFABO,XAFABO/SOL/CRF:SYM=DDXCOM,XAFABO
RUN-TIME: 1 1 .2 SECONDS
RUN-TIME RATIO: 13/2=4.7
CORE USED: 7K (13 PAGES)