

IDENTIFICATION

PRODUCT CODE: MAINDEC-X8-DIKEA-B-D
PRODUCT NAME: DEC/X8 MODULE "EAEALL"
EAE EXERCISE OF MUY, DVI, SWL, LSR,
ASR AND NMI INSTRUCTIONS
DATE CREATED: SEPTEMBER 7, 1972
MAINTAINER: DIAGNOSTIC GROUP
AUTHOR: ED FORTMILLER

COPYRIGHT (C) 1972

DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASS. 01754

1. MODULE DESCRIPTION

"EAEALL" IS A DEC/X8 SOFTWARE MODULE WHICH EXERCISES THE MUY, DVI, SHL, ASR, LSR AND NMI INSTRUCTIONS IN ALL FAMILY-OF-8 EAE'S. IN THE KE8-E EAE BOTH MODES "A" AND "B" ARE UTILIZED. REFER TO PARAGRAPH 4.3 FOR INITIALIZING INFORMATION.

"EAEALL" IS DIVIDED INTO FIVE TEST SECTIONS, TEST X000 THROUGH X004. TEST X000 EXERCISES MUY AND DVI BY SOLVING THE PROBLEM

$$A*B/B*B/A*A/B*B/A=B$$

TEST X001 THROUGH X003 EXERCISE SHL, LSR, AND ASR RESPECTIVELY. THE NUMBER OF SHIFTS RANGE FROM 1 TO 40 (37 IN A KE8-E EAE MODE "B"). THE MAXIMUM NUMBER OF SHIFTS ALLOWED MAY BE CHANGED BY THE USER.

TEST X004 EXERCISES THE NMI INSTRUCTION. THE RESULT OF THE NORMALIZE IS CHECKED BY THE USE OF THE ASR INSTRUCTION.

SINCE TESTS X001 THROUGH X004 MAY CAUSE "DATA REQUEST LATE" OR "DATA RATE" ERRORS ON SOME HIGH SPEED DIRECT MEMORY ACCESS (DATA BREAK) DEVICES, THE USER HAS THE ABILITY TO BYPASS THESE TESTS AND RUN JUST TEST X000. HOWEVER, TEST X000 MAY ALSO CAUSE SIMILAR ERRORS TO OCCUR.

2. REQUIREMENTS

A. PROCESSORS: PDP-8, 8/I, 8/E, 8/M OR PDP-12,

B. OPTIONS: EXTENDED ARITHMETIC ELEMENT (EAE)

C. SPECIAL: NONE

3. RESTRICTIONS

NONE

4. OPERATING INFORMATION

4.1 SPECIAL CONSIDERATIONS

THIS MODULE REQUIRES A NON-VOLATILE STEP COUNTER AND GT FLAG. SR5 SHOULD BE SET TO 0 WHEN THIS MODULE IS RUNNING SINCE THE MQ IS UTILIZED.

"DATA REQUEST LATE" OR "DATA RATE" ERRORS MAY OCCUR IF THIS MODULE IS RUN WHILE EXERCISING A HIGH SPEED DIRECT MEMORY ACCESS (DATA BREAK) DEVICE.

4.2 BUILDING

A. JOB TYPE: BACKGROUND

B. PRIORITY: MUST HAVE LOWER PRIORITY THAN ALL INTERRUPT JOBS; OTHERWISE UNIMPORTANT.

C. JOB SLOTS: JOB SLOTS JF1 AND JF2 ONLY; 4 PAGES REQUIRED;

4.3 INITIALIZING

AFTER "EAEALL" IS PRINTED THE USER SHOULD TYPE THE NUMERICAL CODE NN INDICATED TO ACHIEVE THE DESIRED RESULTS.

CODE -----	RESULTS -----
00	SPECIFIES A PDP-8, 8/I OR PDP-12 EAE, OR "A" MODE ONLY IN A KE8=E EAE. SHIFT AND NORMALIZE TESTS ARE BYPASSED.
01	SAME AS "00" EXCEPT ALL TESTS ARE RUN.
10	"B" MODE ONLY IN A KE8=E EAE. SHIFT AND NORMALIZE TESTS ARE BYPASSED.
11	SAME AS "10" EXCEPT ALL TESTS ARE RUN.
20	BOTH "A" AND "B" MODES ARE UTILIZED IN A KE8=E EAE. SHIFT AND NORMALIZE TESTS ARE BYPASSED.
21	SAME AS "20" EXCEPT ALL TESTS ARE RUN.

NOTE: THE PRESET CONDITION IS "21".

IN ADDITION, THE MAXIMUM NUMBER OF SHIFTS CAN BE CONTROLLED BY SETTING LOCATION "KXXXX" (0364) TO THE MAXIMUM NUMBER OF SHIFTS VIA THE *O COMMAND, WHEN IN KE8-E EAE "A" MODE OR IF A PDP-8, 8/I OR PDP-12 EAE, THE MAXIMUM NUMBER OF SHIFTS WILL BE THE SAME AS THAT SPECIFIED IN KXXXX, HOWEVER, IN KE8-E "B" MODE THE MAXIMUM NUMBER OF SHIFTS WILL BE ONE LESS THAN SPECIFIED IN KXXXX, THIS LOCATION IS PRESET TO 0040,

4.4 DEVICE SETUP

NONE

4.5 RUNNING

A. CNTR: UPDATED AFTER ONE PASS THROUGH ALL TESTS,

B. SR10: NO EFFECT.

C. SR11: NO EFFECT.

5. ERROR INFORMATION

ALL ERRORS RESULT IN A "STAT ERR" REPORT, THE MEANINGS OF THE VARIOUS STATUS WORDS, SA, SB, ETC, VARY ACCORDING TO THE CONTENTS OF "CODE",

5.1 "CODE" DEFINITIONS

BIT 0 OF THE "CODE" WORD INDICATES THE EAE MODE WHILE BITS 9-11 INDICATE THE FAILING TEST (0-4).

CODE WORD 000X INDICATES A FAILURE IN TEST X WITH THE EAE IN MODE "A" OR IN A PDP-8, 8/I OR PDP-12,

CODE WORD 400X INDICATES A FAILURE IN TEST X WITH THE EAE IN MODE "B" (KE8-E EAE ONLY),

12
STATUS WORD DEFINITIONS

A. IF CODE=X000 (MUY,DVI ERROR), THEN:

SA: OPERAND A
SB: OPERAND B
SC: FINAL AC (REMAINDER AFTER FINAL DVI)
SD: FINAL MQ (QUOTIENT)
SE: FINAL LINK (BIT 0), GT FLAG (BIT 1) AND STEP COUNTER
(BITS 7-11)
SF: FINAL AC SHOULD BE (SIMULATED)
SG: FINAL MQ SHOULD BE (SIMULATED)
SH: FINAL LINK, GT FLAG AND STEP COUNTER SHOULD BE (SIMULATED)

B. IF CODE=X001 (SHL ERROR), X002 (LSR ERROR), OR X003 (ASR ERROR), THEN:

SA: STARTING AC
SB: STARTING MQ
SC: FINAL AC
SD: FINAL MQ
SE: FINAL LINK (BIT 0), GT FLAG (BIT 1) AND STEP COUNTER
(BITS 7-11).
SF: FINAL AC SHOULD BE (SIMULATED)
SG: FINAL MQ SHOULD BE (SIMULATED)
SH: FINAL LINK, GT FLAG AND STEP COUNTER SHOULD BE (SIMULATED)
SI: NUMBER OF SHIFTS IN BITS 7-11

NOTE: THE ACTUAL NUMBER OF SHIFTS IN A PDP-8, 8/I OR PDP-12
OR IN "A" MODE IS ONE GREATER THAN INDICATED.

C. IF CODE=X004 (NMI ERROR), THEN:

SA: STARTING AC
SB: STARTING MQ
SC: CONTENTS OF AC AFTER NMI
SD: CONTENTS OF MQ AFTER NMI
SE: CONTENTS OF LINK (BIT 0), GT FLAG (BIT 1) AND STEP COUNTER
(BITS 7-11) AFTER NMI.
SF: CONTENTS OF AC AFTER ASR CHECK (MAY BE SIMULATED)
SG: CONTENTS OF MQ AFTER ASR CHECK (MAY BE SIMULATED)

6. LISTING (ATTACHED)

1
2
3
4
5
6
7
8
9
10
11
12

/DEC/X8 EXTERNAL SYMBOL TABLE "EXTSYH"
/FOR USE IN ASSEMBLING DEC/X8 SOFTWARE MODULES;
/COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS,
XLIST
PAUSE

13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61

/MAINDEC-X8-DIKEA-B-L "DEC/X8" EAEALL
/EAE TEST FOR DEC/X8 EXERCISER WHICH EXERCISES MULTIPLY,
/DIVIDE, SHIFTS, AND NORMALISE IN EITHER "A" AND/OR "B" MODE'S
/FOR THE KEB-E EAE OR IN PDP-8, 8/I OR PDP-12 EAE'S,
/COPYRIGHT 1972, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS; 01794,
/THIS MODULE OPERATIONAL ON A PDP-8, 8/I, 8/E, 8/M OR PDP-12 WITH EAE;
/PRG: ED FORTMILLER (X3341)

/BUILDER INSTRUCTIONS:

- /1. PRIORITY: BACKGROUND JOB - MUST BE PLACED AFTER ALL INTERRUPT
- / JOBS, OTHERWISE PLACEMENT IS NOT IMPORTANT;
- /2. JOB SLOT: 4 PAGES REQUIRED, SLOTS JF1 AND JF2 ONLY.

/INITIALIZER INSTRUCTIONS:

- /1. FOLLOWING THE "EAEALL" PRINTOUT, TYPE IN A 8 FOR "A" (8-8I-12) MODE,
- / 1 FOR "B" (8E) MODE, OR 2 FOR BOTH "A" AND "B" MODES.
- /2. AFTER THE MODE INITIALIZATION, TYPE IN A 8 FOR MULTIPLY AND
- / DIVIDE OR "1" FOR MULTIPLY, DIVIDE, SHIFTS, AND NORMALISE;
- /3. NO INITIALIZATION WILL RESULT IN TEST OF MULTIPLY,
- / DIVIDE, SHIFTS, AND NORMALISE IN BOTH "A" AND "B" MODES.
- /4. THE MAXIMUM NUMBER OF SHIFTS CAN BE CONTROLLED BY SETTING
- / LOCATION "KXXXX" TO THE MAXIMUM NUMBER OF DESIRED SHIFTS*4.
- / WHEN IN "A" MODE THE NUMBER OF SHIFTS WILL BE THE
- / SAME AS THE NUMBER CONTAINED IN LOCATION "KXXXX"; WHEN
- / IN "B" MODE THE MAXIMUM NUMBER OF SHIFTS WILL BE ONE LESS
- / THAN THE NUMBER CONTAINED IN LOCATION "KXXXX"; NO
- / INITIALIZATION WILL RESULT IN A MAXIMUM OF 48 SHIFTS IN
- / "A" MODE AND 37 SHIFTS IN "B" MODE.

```

52      0200      *0200
53
54      0200 0000      0
55      0201 0501      TEXT1, TEXT 'EAEALL'
56      0202 0501
57      0203 1414
58      0204 0000
59      0205 0411      TEXT 'DIKEA-B'
60      0206 1305
61      0207 0155
62      0210 0200
63      0211 0000      HOMEDF, 0
64      0212 7402      HLT
65      0213 5611      JMP I HOMEDF
66      0214 6202      INTACK, CIF 00
67      0215 4426      JMS I IHRET

```

```

62 0216 7777      =-1
63 0217 7777      KILL,  -1
64 0220 7777      KILLED, -1
65 0221 0000      CNTR,   0
66 0222 0000      ERROR,   0
67 0223 3234      DCA      ,+11
68 0224 7604      LAS
69 0225 0073      AND Z   K4
70 0226 7440      SEA
71 0227 3217      DCA      KILL
72 0230 4211      JMS     HOMEOP
73 0231 6002      IOF
74 0232 6202      CIF     00
75 0233 4461      JMS I   ERRP
76 0234 0000      0
77 0235 5622      JMP I   ERROR
78 0236 0000      CODE,   0
79 0237 0000      0
80 0240 7777      A,      7777
81
82 0241 7776      B,      7776
83 0242 0000      BAC,   0
84 0243 0000      BMO,   0
85 0244 0000      BLGTSC, 0
86 0245 0000      SINAC,  0
87 0246 0000      SINMO,  0
88 0247 0000      SLGTSC, 0
89 0250 0000      STSC,  0
90 0251 0002      SMODE,  2
91 0252 0000      TEMPA,  0
92 0253 0000      MODE,   0
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116

```

/XXXX="A"; 4XXX="B";
/OPERAND "A" FOR MUY+DIV; STARTING AC FOR SHIFTS+NN1;
/OPERAND "B" FOR MUY+DIV; STARTING MO FOR SHIFTS+NN1
/BAD AC
/BAD MO
/BAD LINK, GT, AND SC (BIT 0=LINK, BIT 1=GT, BITS 7=11=SC);
/SIMULATED AC
/SIMULATED MO
/SIMULATED LINK, GT, AND SC (BIT 0=LINK, BIT 1=GT, BITS 7=11=SC);
/NUMBER OF SHIFTS IN BITS 9=11
/SELECTED MODE OF OPERATION, 00=1 1=0 2=00TH
/B="A"; 7777="B"
/EQUATE STATEMENTS:
MOB=7501 /INCLUSIVE OR THE MO WITH THE AC;
MQL=7421 /LOAD MO FROM AC THEN CLEAR THE AC;
CAM=7621 /CLEAR AC AND MO;
SWP=7521 /SWAP AC AND MO;
MOCL=7601 /EAC CLEAR AC;
MNOP=7401 /EAC NOP;
SCA=7441 /INCLUSIVE OR THE STEP COUNTER WITH THE AC;
SWAB=7431 /SET EAC TO "B" MODE;
SWA=7447 /SET EAC TO "A" MODE;
MUY=7403 /MULTIPLY;
DVI=7407 /DIVIDE;
SHL=7413 /SHIFT LEFT;
ASR=7419 /ARITHMETIC SHIFT RIGHT;
LSR=7417 /LOGICAL SHIFT RIGHT;
NHI=7411 /NORMALIZE;
GTF=6004 /GET INTERRUPT FLAG; (PDR=07E ONLY);
SGT=6006 /SKIP ON GREATER THAN FLAG; (PDR=07E ONLY);
OPEN=0000 /PROGRAM MODIFIABLE;
BSC=0017 /ASSIGN BAD STEP COUNTER TO AUTO INDEX REGISTER;

```

117
118 0254 4444      INIT,  MESSAGE
119 0255 0201      TEXT1
120 0256 4455      SPACE2
121 0257 4442      ONEOCT
122 0260 0256      JMP     ,=2
123 0261 3251      DCA     SMODE
124 0262 1251      TAD     SMODE
125 0263 7041      CMA IAC
126 0264 3253      DCA     MODE
127 0265 7326      CLA CLL OML RTL
128 0266 1253      TAD     MODE
129 0267 7650      SNA CLA
130 0270 3253      DCA     MODE
131 0271 0201      CDF     00
132 0272 1765      TAD I  A1432
133 0273 4211      JMS     HOMEOP
134 0274 3777      DCA     XINST
135 0275 4442      ONEOCT
136 0276 0256      JMP     INIT+B
137 0277 3776      DCA     TEST
138 0300 0020      INITEX
139
140
141 0301 3221      RUN,   DCA     CNTR
142 0302 3346      DCA     SHIFTS
143 0303 1136      TAD Z   M7
144 0304 3292      DCA     TEMPA
145 0305 1217      TAD     KILL
146 0306 7400      SNA
147 0307 5312      JMP     ,+3
148 0310 3220      DCA     KILLED
149 0311 0004      SERVEX
150 0312 2221      ISB   CNTR
151 0313 2292      ISB   TEMPA
152 0314 5321      JMP     ,+5
153 0315 1136      TAD Z   M7
154 0316 3292      DCA     TEMPA
155 0317 1375      TAD     (+2
156 0320 0004      SERVEX
157 0321 2240      ISB   A
158 0322 5774      JMP     MOSEQ
159 0323 2241      ISB   B
160 0324 5774      JMP     MOSEQ
161 0325 5774      JMP     MOSEQ
162
163
164
165
166
167 0326 0000      SHIFT, OPEN
168 0327 1346      TAD     SHIFTS
169 0330 1253      TAD     MODE
170 0331 7101      IAC CLL
171 0332 7600      SNA CLA

```

/GET SELECTED MODES
/AND NEGATE IT;
/B="A"; 7777="B";
/CHECK FOR BOTH SELECTED;
/HAS BOTH MODES SELECTED?
/YES, SET MODE TO 0000;
/ROUTINE TO PERFORM EAE SHIFT,
/GET NUMBER OF SHIFTS
/ADD MODE
/ADD 1
/"B" MODE AND 6 SHIFTS)

172	0333	2346	ISE	SHIFTS	/SET SHIFTS TO 1;
173	0334	1726	TAD I	SHIFTS	/OBTAIN THE SHIFT INSTRUCTION
174	0335	3345	DCA	SHIFTS-1	/AND STORE IT FOR EXECUTION
175	0336	2326	ISE	SHIFTS	/+1 FOR EXIT.
176	0337	1346	TAD	SHIFTS	/SET PRINTOUT
177	0340	3250	DCA	STSC	/LOCATION
178	0341	1241	TAD	B	/LOAD STARTING MO
179	0342	7421	HQL		
180	0343	1240	TAD	A	/LOAD STARTING AC
181	0344	7000	NOF		/DUMMY
182	0345	0000	OPEN		/SHIFT INSTRUCTION;
183	0346	0000	SHIFTS, OPEN		/NUMBER OF SHIFTS;
184	0347	4773	JMS	SAVREG	/SAVE THE REGISTERS,
185	0350	1366	TAD	K37	/SET UP
186	0351	0253	AND	MODE	/THE SIMULATED
187	0352	3772	DCA	SHMSC	/STEP COUNTER
188	0353	4771	JMS	CHECK	/CHECK ACTUAL AGAINST SIMULATED
189	0354	2346	ISE	SHIFTS	/SET UP FOR NEXT SHIFT
190	0355	1364	TAD	KXXXX	/
191	0356	7041	CMA IAC		/
192	0357	1346	TAD	SHIFTS	/
193	0360	7640	SEA CLA		/FINISH SHIFTING;
194	0361	9726	JMP I	SHIFT	/NO,
195	0362	2326	ISE	SHIFT	/+1 FOR EXIT
196	0363	9726	JMP I	SHIFT	/TEST COMPLETE
197					
198	0364	0040	KXXXX, 40		/MAXIMUM NUMBER OF SHIFTS=4;
199	0365	1432	A1432, 1432		
200	0366	0037	K37, 37		
201					
202		0367	EOP1=,		
203		0255	*INIT=1		
204	0255	0201	TEXT1		
205		0367	*EOP1		
206					
207					
208	0371	0616			
209	0372	0755			
210	0373	0600			
211	0374	0400			
212	0375	0321			
213	0376	0555			
214	0377	0457			
215		0400	PAGE		
216	0400	1351	MOSEQ, TAD	K7770	/
217	0401	3777	DCA	CODE+1	/
218	0402	4240	JMS	PROB	/ A + B
219	0403	1776	TAD	B	
220	0404	7040	CMA	B	
221	0405	3776	DCA	B	/ A + NOT B
222	0406	4240	JMS	PROB	
223	0407	1775	TAD	A	
224	0410	7040	CMA	A	
225	0411	3775	DCA	A	

226	0412	4240	JMS	PROB	/ NOT A + NOT B
227	0413	1776	TAD	B	
228	0414	7040	CMA	B	
229	0415	3776	DCA	B	
230	0416	4240	JMS	PROB	/ NOT A + B
231	0417	1775	TAD	A	
232	0420	7040	CMA	A	
233	0421	3775	DCA	A	/ A + B
234	0422	1355	TAD	TEST	/
235	0423	7640	SEA CLA		/
236	0424	5774	JMP	SHLST	/PERFORM SHIFT TEST,
237	0425	7344	CLA CLL	CMA RAL	=B
238	0426	1773	TAD	SHODE	/ADD VALUE OF SHODE AND
239	0427	7640	SEA CLA		/SEE IF BOTH MODES ARE SELECTED.
240	0430	5772	JMP	RUN+4	/ONLY ONE MODE SELECTED;
241	0431	1771	TAD	MODE	/GET PRESENT MODE
242	0432	7040	CMA	MODE	/AND COMPLEMENT IT
243	0433	3771	DCA	MODE	/SO A PASS IS MADE IN OPPOSITE MODE;
244	0434	1771	TAD	MODE	/GET MODE AND SEE IF IT'S A MODE
245	0435	7640	SEA CLA		/INDICATING EXECUTION OF BOTH MODES;
246	0436	9200	JMP	MOSEQ	/EXECUTE IN "B"
247	0437	5772	JMP	RUN+4	/
248					
249	0440	0000	PROB, OPEN		/ A+B/B+B/A+1/B+B/X+B;
250	0441	3770	DCA	CODE	/
251	0442	1775	TAD	A	/CHECK A FOR B;
252	0443	7650	SNA	CLA	/A IS OK;
253	0444	0640	JMP I	PROB	/A IS B SO EXIT
254	0445	1776	TAD	B	/CHECK B FOR B;
255	0446	0650	SNA CLA		/B IS OK;
256	0447	0640	JMP I	PROB	/B IS B SO EXIT
257	0450	1771	TAD	MODE	/FIND OUT MODE TO BE EXECUTED;
258	0451	7640	SEA CLA		/TEST CONDITION
259	0452	0661	JMP	+7	/SET UP FOR B MODE;
260	0453	1775	TAD	A	/GET A OPERAND
261	0454	3353	DCA	ADPER	/
262	0455	3776	TAD	B	/GET B OPERAND
263	0456	3354	DCA	ROPER	/
264	0457	4447	XINST, SHSA		/GO TO MODE A, IF IN A NON BC THIS WILL BE A "NOP";
265	0460	0267	JMP		/
266	0461	1375	TAD	(A	/GET A OPERAND ADDRESS
267	0462	3353	DCA	ADPER	/
268	0463	1376	TAD	(B	/GET B OPERAND ADDRESS
269	0464	3354	DCA	ROPER	/
270	0465	7447	SHSA		/
271	0466	7431	SHAB		/ZERO OF FLAG;
272	0467	7621	CAM		/GO TO MODE B;
273	0470	1353	TAD	ADPER	/CLEAR AC AND NO;
274	0471	3320	DCA	A2	
275	0472	1353	TAD	ADPER	
276	0473	3322	DCA	A3	
277	0474	1353	TAD	ADPER	
278	0475	3330	DCA	A4	
279	0476	1354	TAD	ROPER	
280	0477	3312	DCA	B1	


```

281 0500 1354 TAD BOPER
282 0501 3314 DCA B2
283 0502 1354 TAD BOPER
284 0503 3316 DCA B3
285 0504 1354 TAD BOPER
286 0505 3324 DCA B4
287 0506 1354 TAD BOPER
288 0507 3326 DCA B5
289 0510 1775' A1. TAD A
290 0511 7425 MQL MUY
291 0512 0241 B1. B
292 0513 7407 DVI
293 0514 0241 B2. B
294 0515 7405 MUY
295 0516 0241 B3. B
296 0517 7407 DVI
297 0520 0240 A2. A
298 0521 7405 MUY
299 0522 0240 A3. A
300 0523 7407 DVI
301 0524 0241 B4. B
302 0525 7405 MUY
303 0526 0241 B5. B
304 0527 7407 DVI
305 0530 0240 A4. A
306 0531 4767' JMS SAVREG
307 0532 1352 TAD M15 /SAVE REGISTERS;
308 0533 7041 CMA IAC M15 /SET UP THE-
309 0534 3766' DCA SIMC /SIMULATED-
310 0535 3765' DCA SIMGT /STEP COUNTER
311 0536 3764' DCA SIML /B TO SIMULATED GT;
312 0537 3763' DCA SIMAC /B TO SIMULATED LINK;
313 0540 1776' TAD B /B TO SIMULATED AG;
314 0541 3762' DCA SIMMO /B TO SIMULATED MQ;
315 0542 4761' JMS CHECK /CHECK SIMULATED AGAINST ACTUAL.
316 0543 5640 JMP I PROG /EXIT.
317
318 /ROUTINE TO LOAD AC+MQ FOR NMI TEST.
319
320 0544 0000 LMOAC, OPEN
321 0545 1776' TAD B /
322 0546 7421 MQL A /LOAD MQ;
323 0547 1775' TAD A /LOAD AC;
324 0550 3744 JMP I LMOAC /EXIT.
325
326 0551 7770 K7770, 7770
327 0552 7763 M15, =15
328 0553 0000 AOPER, OPEN
329 0554 0000 BOPER, OPEN
330 0555 0001 TEST, 1 /B=MUY+DVI; S=MUY;DVI; SHIFTS, AND NORMALISE.
331
332 0556 .
333
334
335 0561 0616

```

```

336 0562 0246
337 0563 0245
338 0564 0756
339 0565 0757
340 0566 0755
341 0567 0600
342 0570 0236
343 0571 0233
344 0572 0305
345 0573 0251
346 0574 0663
347 0575 0240
348 0576 0241
349 0577 0237
350
351 PAGE
352 /ROUTINE TO SAVE REGISTERS;
353 0600 0000 SAVREG, OPEN
354 0601 3777' DCA BAC /SAVE THE AC;
355 0602 7001 MQA /MQ=>BAC;
356 0603 3776' DCA BMO /SAVE THE MQ;
357 0604 7441 SCA /SQ=>BAC;
358 0605 3017 AUA, DCA BSC /SAVE THE STEP COUNTER;
359 0606 7010 RAR /LINK=SPACE;
360 0607 3354 DCA BLINK /SAVE THE LINK;
361 0610 1775' TAD MODE /CHECK THE MODE AND
362 0611 7640 SEA CLA /IF "A" MODE SKIP NEXT INSTRUCTION;
363 0612 0004 GTF /FLAG=>BAC;
364 0613 0122 AND Z K0000 /SAVE ONLY THE GT FLAG;
365 0614 3353 DCA BGT /SAVE THE BT FLAG;
366 0615 5600 JMP I SAVREG /EXIT. AQ AND LINK=0;
367
368 /ROUTINE TO CHECK AC, MQ, L, GT, SQ.
369
370
371 0616 0000 CHECK, OPEN
372 0617 4242 JMS COMERR /COMBINE DATA
373 0620 7701 CLA MQA //
374 0621 7041 CIA //
375 0622 1774' TAD SIMMO //
376 0623 7640 SEA CLA /MQ CHECK OK?
377 0624 9240 JMP DATERR /NO
378 0625 1777' TAD BAC //
379 0626 7041 CIA //
380 0627 1773' TAD SIMAC //
381 0630 7640 SEA CLA /SIMULATED AGAINST ACTUAL AQ?
382 0631 9240 JMP DATERR /NO
383 0632 1772' TAD SLOTSG //
384 0633 7041 CIA //
385 0634 1771' TAD SLOTSG //
386 0635 7640 SEA CLA /SIMULATED L.GT,SC AGAINST ACTUAL L.GT,SC?
387 0636 9240 JMP DATERR /NO
388 0637 5616 JMP I CHECK /CHECK COMPLETE
389 0640 4770' DATERR, JMS ERROR

```

```

390 0641 0616      JMP I CHECK /
391
392 /ROUTINE TO COMBINE ERROR DATA;
393
394 0642 0000 COMERR, OPEN
395 0643 1767' TAD CODE /SET CODE ACCORDING TO MODE AND TEST;
396 0644 0075 AND Z K7 //
397 0645 3767' DCA CODE //
398 0646 7330 CLA CLL CML RAR //
399 0647 0775' AND MODE //
400 0650 1767' TAD CODE //
401 0651 3767' DCA CODE //
402 0652 1354 TAD BLINK //COMBINE BAD LINK, GT, AND STEP COUNTER;
403 0653 1817' TAD BSC //
404 0654 1383 TAD BGT //
405 0655 3771' DCA BLGTSC //
406 0656 1356 TAD SIML //COMBINE SIMULATED LINK, GT, AND STEP COUNTER;
407 0657 1355 TAD SIMSC //
408 0660 1357 TAD SIMGT //
409 0661 3772' DCA BLGTSC //
410 0662 0642 JMP I COMERR /EXIT.
411
412 /TEST OF THE SHIFT LEFT INSTRUCTION
413 /ERROR CODE=X001.
414
415 0663 1360 SHLTST, TAD K7767 //
416 0664 3766' DCA CODE+1 //
417 0665 4321 JMS SETSIM //
418 0666 1774' TAD SIMMO //
419 0667 7104 RAL CLL //
420 0670 3774' DCA SIMMO //
421 0671 1773' TAD SIMAC //
422 0672 7804 RAL //
423 0673 3773' DCA SIMAC //
424 0674 7810 RAR //
425 0675 3386 DCA SIML //
426 0676 3397 DCA SIMGT //
427 0677 4765' JMS SHIFT //
428 0700 7413 SHL //
429 0701 0266 JMP SHLTST+3 //
430
431 /TEST OF THE LOGICAL SHIFT RIGHT INSTRUCTION
432 /ERROR CODE=X002.
433
434 0702 4321 LSRTST, JMS SETSIM //
435 0703 1773' TAD SIMAC //
436 0704 7110 CLL RAR //
437 0705 3773' DCA SIMAC //
438 0706 1774' TAD SIMMO //
439 0707 7810 RAR //
440 0710 3774' DCA SIMMO //
441 0711 7812 RTR //
442 0712 0775' AND MODE //
443 0713 3397 DCA SIMGT //
444 0714 3386 DCA SIML //

```

```

445 0715 4765' JMS SHIFT //
446 0716 7417 LSR //
447 0717 0303 JMP LSRTST+1 //
448 0720 0764' JMP ASRTST //
449
450 /ROUTINE TO SET SIMULATED.
451
452 0721 0000 SETSIM, OPEN
453 0722 2767' ISR CODE //
454 0723 3763' DCA SHIFTS //B TO NUMBER OF SHIFTS;
455 0724 1762' TAD A //
456 0725 3773' DCA SIMAC //SET SIMULATED AC;
457 0726 1761' TAD B //
458 0727 3774' DCA SIMMO //SET SIMULATED MO;
459 0730 1775' TAD MODE //
460 0731 7600 SNA CLA //IF MODE "0"/ B->BT;
461 0732 0721 JMP I SETSIM //
462 0733 7447 SWA //B GT FLAG;
463 0734 7431 SWAB //
464 0735 0721 JMP I SETSIM //
465
466 /ROUTINE TO SET "SIMAC AND "SIMMO" FOR THE NMI TEST;
467
468 0736 0000 SENSIM, B
469 0737 7901 MCA //
470 0740 1352 TAD M4000 //
471 0741 7640 SNA CLA //
472 0742 0345 JMP ,+3 //
473 0743 1775' TAD MODE //
474 0744 7650 SNA CLA //
475 0745 1762' TAD A //
476 0746 3773' DCA SIMAC //SET SIMULATED AC;
477 0747 1761' TAD B //
478 0750 3774' DCA SIMMO //SET SIMULATED MO;
479 0751 0736 JMP I SENSIM //
480
481 0752 4000 M4000, 4000
482 0753 0000 BGT, OPEN
483 0754 0000 BLINK, OPEN
484 0755 0000 SIMSC, OPEN
485 0756 0000 SIML, OPEN
486 0757 0000 SIMGT, OPEN
487 0760 7767 K7767, 7767
488
489 0761 *
490
491
492 0761 0241
493 0762 0240
494 0763 0346
495 0764 1000
496 0765 0326
497 0766 0237
498 0767 0236
499 0770 0222

```

500 0771 0244
 501 0772 0247
 502 0773 0245
 503 0774 0246
 504 0775 0253
 505 0776 0243
 506 0777 0242
 1000

PAGE

507
 508 /TEST OF THE ARITHMETIC SHIFT RIGHT INSTRUCTION
 509 /ERROR CODE=X003,
 510 ASRTST, JMS SETSIM //
 511 1000 4777' CLL //
 512 1001 7100 TAD SIMAC //
 513 1002 1776' SPA //
 514 1003 7510 CML //
 515 1004 7020 RAR //
 516 1005 7010 DCA SIMAC //
 517 1006 3776' TAD SIMMQ //
 518 1007 1775' RAR //
 519 1010 7010 DCA SIMMQ //
 520 1011 3775' RTR //
 521 1012 7012 AND MODE //
 522 1013 0774' DCA SIMGT //
 523 1014 3773' TAD SIMAC //
 524 1015 1776' AND K4000 //
 525 1016 0354 DCA SIML //
 526 1017 3772' JMS SHFT //
 527 1020 4771' ASR //
 528 1021 7415 JMP ASRTST+1 //
 529 1022 5201

530 /TEST OF THE NORMALIZE INSTRUCTION
 531 /ERROR CODE=X004,
 532 NM11, ISB CODE //
 533 1023 2770' JMS LMGAC //LOAD MQ AND AC,
 534 1024 4767' JMS SENSIM //SET SIMAC AND SIMMQ, REGARDLESS,
 535 1025 4766' TAD A //LOAD AC,
 536 1026 1765' NMI //NORMALIZE,
 537 1027 7411 JMS SAVREG //SAVE REGISTERS,
 538 1030 4764' TAD BAC
 539 1031 1763' CLL RAL
 540 1032 7104 SEL //AC=1?
 541 1033 7430 JMP NMI2 //AC=1,
 542 1034 5202 SPA //AC1=1?
 543 1035 7510 JMP NMI4 //AC=0 AND AC1=1,
 544 1036 5306 MQA //AC=AC1=0,
 545 1037 7501 SEA CLA //AC AND MQ 0?
 546 1038 7640 JMP NMIERR //AC AND/OR MQ NOT 0,
 547 1041 5347 SCA
 548 1042 7441 SEA CLA //SC=0?
 549 1043 7640 JMP NMIERR //SC NOT 0,
 550 1044 5347 JMS LMGAC //LOAD MQ AND AC,
 551 1045 4767' MQA //OR" MQ TO AC,
 552 1046 7501 SNA CLA //STARTING AC AND MQ=0?
 553 1047 7650 JMP SHFRET //EXIT

554 1051 1765' TAD A //GET STARTING AC,
 555 1052 7104 CLL RAL
 556 1053 7501 MQA
 557 1054 7640 SEA CLA //STARTING AC AND MQ 4000 0000?
 558 1055 5347 JMP NMIERR //
 559 1056 5347 TAD MODE //
 560 1057 7650 SNA CLA //MODE "B"?
 561 1060 5347 JMP NMIERR //
 562 1061 5762' JMP SHFRET //EXIT
 563 1062 7700 NM12, SMA CLA //AC1=0?
 564 1063 5272 JMP NMI3 //YES, AC=1 AND AC1=0,
 565 1064 7332 CLA CLL CML RTR //2000
 566 1065 1763' TAD BAC
 567 1066 7501 MQA
 568 1067 7640 SEA CLA //AC AND MQ=0000 0000?
 569 1070 5347 JMP NMIERR //
 570 1071 5306 JMP NMI4 //YES,
 571 1072 4767' NMI3, JMS LMGAC //LOAD MQ AND AC,
 572 1073 7104 CLL RAL
 573 1074 7501 MQA //
 574 1075 7640 SEA CLA //STARTING AC AND MQ=4000 0000?
 575 1076 5306 JMP NMI4 //
 576 1077 1774' TAD MODE //
 577 1100 7640 SEA CLA //MODE "A"?
 578 1101 5347 JMP NMIERR //NO,
 579 1102 7441 SCA //SC=0?
 580 1103 7640 SEA CLA //NO,
 581 1104 5347 JMP NMIERR //YES,
 582 1105 5340 JMP NMI3 //YES,
 583 1106 7641 NMI4, SCA CLA //
 584 1107 7650 SNA CLA //
 585 1110 5340 JMP NMI5 //
 586 1111 1774' TAD MODE //
 587 1112 7140 CMA CLL //
 588 1113 1017 AUC, TAD SSC //
 589 1114 3321 DCA NHISHP //
 590 1115 1761' TAD BMQ //
 591 1116 7421 MQL //
 592 1117 1763' TAD BAC //
 593 1120 7415 ASR //ARITHMETIC SHIFT RIGHT;
 594 1121 0000 NMISHP, B //
 595 1122 3776 DCA SIMAC //SAVE SHIFTED AC
 596 1123 7501 MQA //SHIFTED MQ->AC
 597 1124 3775' DCA SIMMQ //SAVE SHIFTED MQ
 598 1125 1776' TAD SIMAC //RELOAD THE AC,
 599 1126 7041 CMA IAC //
 600 1127 1763' TAD A //
 601 1130 7640 SEA CLA //AC CHECK?
 602 1131 5347 JMP NMIERR //
 603 1132 7501 MQA //
 604 1133 7041 CMA IAC //
 605 1134 1760' TAD B //MQ CHECK?
 606 1135 7640 SEA CLA //
 607 1136 5347 JMP NMIERR //
 608 1137 5762' JMP SHFRET //EXIT

609	1140	1763'	NHIS,	TAD	BAC	/
610	1141	7841		CMA	IAC	
611	1142	1765'		TAD	A	
612	1143	7640		SEA	CLA	
613	1144	5347		JMP	NMIERR	/
614	1145	1761'		TAD	BMO	/
615	1146	5333		JMP	,=13	/
616						
617	1147	1136	NMIERR,	TAD	Z	K7771
618	1150	3757'		DCA		CODE+1
619	1151	4756'		JMS		COMERR
620	1152	4755'		JMS		ERROR
621	1153	5762'		JMP		SHFRET
622						
623	1154	4000	K4000,	4000		
624						
625		1155		*		
626						
627	1155	0222				
628	1156	0642				
629	1157	0237				
630	1160	0241				
631	1161	0243				
632	1162	0425				
633	1163	0242				
634	1164	0600				
635	1165	0240				
636	1166	0736				
637	1167	0544				
638	1170	0236				
639	1171	0326				
640	1172	0756				
641	1173	0757				
642	1174	0253				
643	1175	0246				
644	1176	0245				
645	1177	0721				

```

0000
0100

0200 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0300 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

0400 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0500 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

0600 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
0700 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

1000 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
1100 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111

1200
1300

1400
1500

1600
1700

2000
2100

2200
2300

2400
2500

2600
2700

3000
3100

3200
3300

3400
3500

3600
3700
    
```

4000
4100
4200
4300
4400
4500
4600
4700

5000
5100
5200
5300
5400
5500
5600
5700

6000
6100
6200
6300

6400
6500

6600
6700

7000
7100

7200
7300

7400
7500

7600
7700

646	0001	FIELD	1
647		/LOADER CALL	
648			
649	1200	0254	INIT
650	1201	0301	RUN
651	1202	0000	0
652	1203	7775	=3
653	1204	0605	AUAJAU8JAU8
654	1205	0653	
655	1206	1113	
656	1207	0000	0
657	1210	0000	0
658	1211	0000	0
659			
660			S

0000
0100
0200
0300
0400
0500
0600
0700

1000
1100

1200 11111111 11000000 00000000 00000000 00000000 00000000 00000000 00000000
1300 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000

1400
1500
1600
1700

2000
2100
2200
2300

2400
2500
2600
2700

3000
3100
3200
3300
3400
3500
3600
3700

4000
4100
4200
4300
4400
4500
4600
4700

5000
5100
5200
5300
5400
5500
5600
5700

6000
6100
6200
6300
6400
6500
6600
6700

7000
7100
7200
7300
7400
7500
7600
7700

A	0240	K0	0066	KILL	0217	RLBUFF	0087
A1	0510	K10	0076	KILLED	0200	RUN	0301
A1432	0365	K100	0107	KIOP	0004	SAVREC	0600
A2	0520	K11	0077	KXXX	0364	SCA	7441
A3	0522	K116	0071	LISN	4440	SENSIM	0736
A4	0530	K13	0100	LISNP	0040	SERVEX	5004
ADPER	0553	K17	0101	LMOAC	0544	SETSIN	0721
ASBUFF	4460	K177	0130	LSR	7417	SOI	6006
ASBUFF	0060	K20	0102	LSRTST	0702	SHPRET	0485
ASR	7415	K200	0110	M15	0552	SHIFT	0326
ASRTST	1000	K2000	0122	M20	0135	SHIFTS	0346
AUA	0605	K212	0111	M200	0127	SHL	7413
AUB	0653	K215	0112	M240	0126	SHLTST	0043
AUC	1113	K240	0113	M260	0125	SIMAC	0240
AUTO	0017	K240	0114	M270	0125	SIMC	0797
B	0241	K272	0115	M3	0141	SIMG	0796
B1	0512	K277	0116	M30	0134	SIMHG	0246
B2	0514	K3	0072	M4	0140	SIMHC	0705
B3	0516	K30	0103	M40	0133	SLOTRO	0247
B4	0524	K301	0117	M4000	0792	SMODE	0291
B5	0526	K32	0067	M43	0132	SPACE2	4495
BAC	0242	K323	0120	M5	0137	SPACEP	0055
BGT	0753	K37	0166	M7	0136	STOC	0250
BLOTSC	0244	K4	0073	MDBCO	0400	SWAB	7431
BLINK	0754	K40	0104	MESSAGE	4444	SWBA	7447
BMO	0243	K400	0121	MESSAGEP	0004	SWP	7921
BOPER	0554	K4000	1154	MODE	0003	TEMPA	0292
BSC	0017	K5	0074	MGA	7901	TEST	0095
CAM	7621	K5000	0123	MGLA	7601	TEXT1	0201
CHECK	0616	K540	0124	MGL	7401	THOODP	0041
CNTR	0221	K5402	0003	MGNOP	7401	THOODT	4441
CODE	0236	K64	0070	MUL20P	0005	TYPE	4400
COMERR	0642	K7	0075	MUY	7405	TYPEP	0000
CRLF	4494	K70	0105	NMI	7411	XINST	0487
CRLFPP	0054	K7510	0125	NMI1	1003		
CATERR	0640	K7520	0126	NMI2	1002		
CVI	7407	K7540	0127	NMI3	1072		
POP1	0367	K7600	0131	NMI4	1106		
ERROR	0222	K77	0106	NMI5	1140		
ERRP	0061	K7735	0132	NMIERR	1147		
EXINIT	0020	K7740	0133	NMISHF	1121		
EXSERV	0004	K7750	0134	ONEOCP	0042		
EXMTEM	0141	K7760	0135	ONEOCT	4442		
FOROCP	0043	K7767	0760	OPEN	0000		
FOROCT	4443	K7770	0551	PRNT1	4401		
GTF	6004	K7771	0136	PRNT1P	0001		
HOMEDF	0211	K7773	0137	PRNT2	4402		
INRETP	0026	K7774	0140	PRNT2P	0002		
INIT	0254	K7775	0141	PRNT4	4403		
INITEX	0020	KCOF	0064	PRNT4P	0003		
INTACK	0214	KCIF	0005	PROB	0440		
IOFMS	0056	KCIFDF	0020	RLBUFF	4497		

ERRORS DETECTED: 0
 LINKS GENERATED: 122
 RUN-TIME: 7 SECONDS
 3K CORE USED