

AD-A044 126

ARMY MISSILE RESEARCH AND DEVELOPMENT COMMAND REDSTO--ETC F/G 17/8
OPTICAL CONTRAST VARIABLE GATE CENTROID TRACKER FOR THE INTEL 3--ETC(U)

UNCLASSIFIED

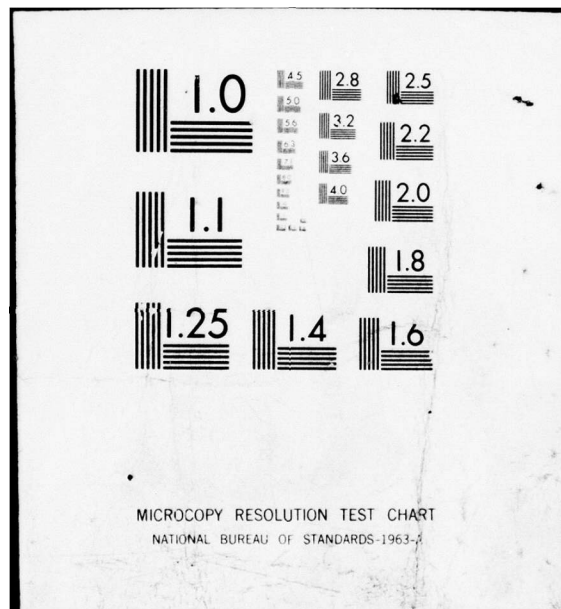
ORDMI-TE-77-13

NL

| OF |
AD
A044 126



END
DATE
FILMED
9-77
DDC



ADA 044 126



**U.S. ARMY
MISSILE
RESEARCH
AND
DEVELOPMENT
COMMAND**



Redstone Arsenal, Alabama 35809

DDC FILE COPY,

DMI FORM 1000, 1 APR 77

2 (2)
TECHNICAL REPORT TE-77-13

OPTICAL CONTRAST VARIABLE GATE
CENTROID TRACKER FOR THE INTEL
3000 MICRO PROCESSOR

Advanced Sensors Directorate
Technology Laboratory

June 1977

Approved for public release; distribution unlimited.

DDDC
RECEIVED
SEP 15 1977
RECEIVED
B

DISPOSITION INSTRUCTIONS

**DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT
RETURN IT TO THE ORIGINATOR.**

DISCLAIMER

**THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN
OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED
BY OTHER AUTHORIZED DOCUMENTS.**

TRADE NAMES

**USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES
NOT CONSTITUTE AN OFFICIAL INDORSEMENT OR APPROVAL OF
THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.**

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER TE-77-13	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER ⑦
4. TITLE (and Subtitle) OPTICAL CONTRAST VARIABLE GATE CENTROID TRACKER FOR THE INTEL 3000 MICRO PROCESSOR.		5. TYPE OF REPORT & PERIOD COVERED Technical Report
6. AUTHOR(s) S./Richard F./Sims		7. PERFORMING ORG. REPORT NUMBER TE-77-13
8. PERFORMING ORGANIZATION NAME AND ADDRESS Commander US Army Missile Research and Development Command Attn: DRDMI-TE Redstone Arsenal, Alabama 35809		9. CONTRACT OR GRANT NUMBER(s)
10. CONTROLLING OFFICE NAME AND ADDRESS Commander US Army Missile Research and Development Command Attn: DRDMI-TI Redstone Arsenal, Alabama 35809		11. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS DA 1W362303A214 ANCMMS 632303.2140111.06
12. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) ⑫ 79p.		13. REPORT DATE 17 June 1977
		14. NUMBER OF PAGES 79
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) MICRO code list MACRO code list EAI PACER 100 Optical contrast variable gate centroid tracker		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report presents a description of the optical contrast variable gate centroid tracker. Tracking flow charts and computer listings are also presented.		

DD FORM 1473
1 JAN 73

EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

410143

JP

OXFORD UNIVERSITY PRESS

5. CLASSIFICATION:

CONTENTS

	Page
I. INTRODUCTION	3
II. MAIN TRACKER LOGIC FLOW DESCRIPTION	3
III. CONCLUSIONS	5
Appendix A. TRACKER MACRO CODE LISTING	23
Appendix B. TRACKER MICRO CODE LISTING	27
Appendix C. TRACKER MICRO CODE LISTING WITH EXPANDED BIT LISTING.	45
Appendix D. EAI PACER 100 EMULATION MICRO CODE LISTING	70

ACCESSION No.	
NTIS	Write Section ☒
DDC	Dist. Section ☐
UNANNOUNCED	☐
JUS. ACTION	
BY	
DISTRIBUTION/AVAILABILITY CODES	
Dist.	TS/Spec
A	

I. INTRODUCTION

The US Army Missile Research and Development Command's Imaging Seeker Algorithm Development Microprocessor System, which employs the Intel 3000 microprocessor, was used in developing this optical contrast variable gate centroid tracker. The tracker processes data taken from a 10K memory which has been "filled" with digitized data from a 100×100 element image produced by a Fairchild Charge Couple Device (CCD) 201 camera. A hardware description of this system is currently being prepared and will be published in a forthcoming report.

The major portion of the tracker program is micro coded and is called through macro-mnemonics which are used like subroutine calls. This macro instruction set is an emulation of the EAI-PACER 100 assembly language with some additions to allow for the specialized tracker mnemonics. The actual program listings for the micro and macro code are presented in the appendices.

II. MAIN TRACKER LOGIC FLOW DESCRIPTION

Figure 1 shows the main flow of the optical contrast variable gate centroid tracker. The macro code presented in Appendix A implements the tracker through the micro coded subroutines presented in Appendix B.

Block No. 1 in Figure 1 corresponds to sequence numbers 19 through 23 in the macro code of Appendix A. The START routine initializes I, J, K, and L for the minimum gate as shown in Figure 2. The octal ten in latch 6 will put out the cross hairs on the gate display. The WINDOW routine will load the window latches with the current I, J, K, and L as shown in Figure 3. The tracker will display its minimum gate with cross hairs when it stops on the pause instruction. This is a "cage" mode where the contrast of the target within the 2 by 2 gate will be tracked upon resumption of execution by single stepping past the pause.

Block No. 2 in Figure 1 corresponds to sequence numbers 24 through 48 in the macro code. The six in latch 6 will enable the status on the 10K memory to be loaded from the 100×100 CCD array and will also enable the frame request. The FRAMEW routine will issue a frame request strobe to fill the 10K memory and loop until the status of the memory is ready as shown in Figure 4. Instructions 27 through 37 calculate the mean value of the data within the minimum gate. When this is compared to the "nominal" value of seven in this case (instructions 38 through 48), various flags are set to track either a "black" or "white" target. This nominal value does not need to be fixed as it is here, but could be set by an automatic gain control in an optimum situation. Instruction 49 reinitializes the I, J, K, and L micro registers destroyed during the mean value and compare calculations (START routine, Figure 5). Instructions 50 through 53 set up the bias and gain latches which decide the amplitude

and bias point in the video that is to be digitized coming from the CCD array. Instructions 55 and 59 set the color of the tracking gate depending upon what contrast target will be tracked.

Block No. 3 in Figure 1 is represented by a FRAMEW instruction (sequence No. 57). Instructions 54 through 61 represent block No. 4 in Figure 1.

Instructions 62 through 73 represent block No. 5 in Figure 1. The MEAN instruction takes the digitized data within the tracking gate and calculates the mean value as shown in Figure 6.

Instructions 74, 75, and 76 represent block No. 6 in Figure 1. The EGATES instruction sets the size of the edge gates shown in Figure 2. It also sets the target flag if the macro code mean value calculation found the mean to be less than seven as shown in Figure 7.

Instructions 77 through 99 represent block No. 7 in Figure 1. The CNTRD instruction calculates the centroid of the target within the tracking gate as shown in Figure 8.

Instructions 100 through 122 represent block No. 8 in Figure 1. NGATEA and NGATEC calculate NX and NY, respectively.

$$NX = TAXG - AXG/2 + 4 YS$$

$$NY = TAYG - AYG/2 + 4 XS$$

TAXG is the target area in X gate

TAYG is the target area in Y gate

AXG is the total area of X gate

AYG is the total area of Y gate

XS is the length of gate in X direction

YS is the length of gate in Y direction

NGATEB and NGATEL calculate I, K and J, L, respectively.

$$DX = NX/(4 YS)$$

$$DY = NY/(4 XS)$$

$$DELXX = - \left(\frac{XS}{2} + DX \right)$$

$I = X \text{ beginning of gate} = \text{DELXX} + CX$

$K = X \text{ end of gate} = CX - \text{DELXX}$

$\text{DELYY} = -\left(\frac{YS}{2} + DY\right)$

$J = Y \text{ beginning of gate} = \text{DELYY} + CY$

$L = Y \text{ end of gate} = CY - \text{DELYY}$

$CX = \text{target centroid in X direction}$

$CY = \text{target centroid in Y direction}$

A detailed calculation of the new gate is shown in Figures 9 through 12.

III. CONCLUSIONS

The current optical contrast variable gate centroid tracker is now undergoing several stages of optimization. This has been implemented and is useful in locking onto and tracking targets at long ranges; however, it does have the same limitations as other contrast trackers, i.e., a well bounded "patch" of contrast is necessary to track properly.

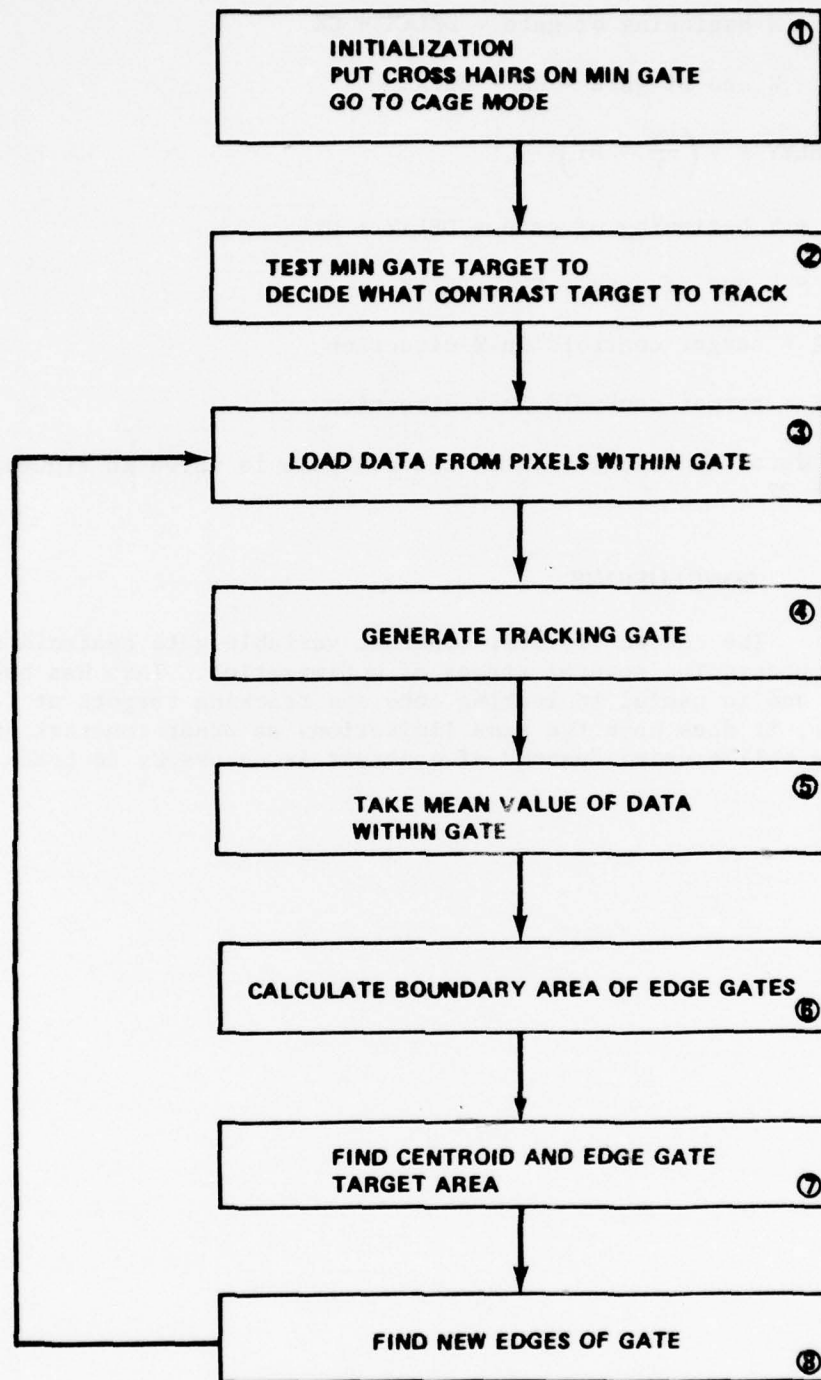


Figure 1. Main tracking loop.

MACRO CALLING INSTRUCTION
START INIT

INITIALIZE I, J, K, AND L

LOAD DATA FROM INIT+0 AND STORE IN I
INIT+1 AND STORE IN J
INIT+2 AND STORE IN K
INIT+3 AND STORE IN L

RETURN

I, J, K, AND L ARE REGISTERS IN THE INTEL 3000
CENTRAL PROCESSING UNIT.

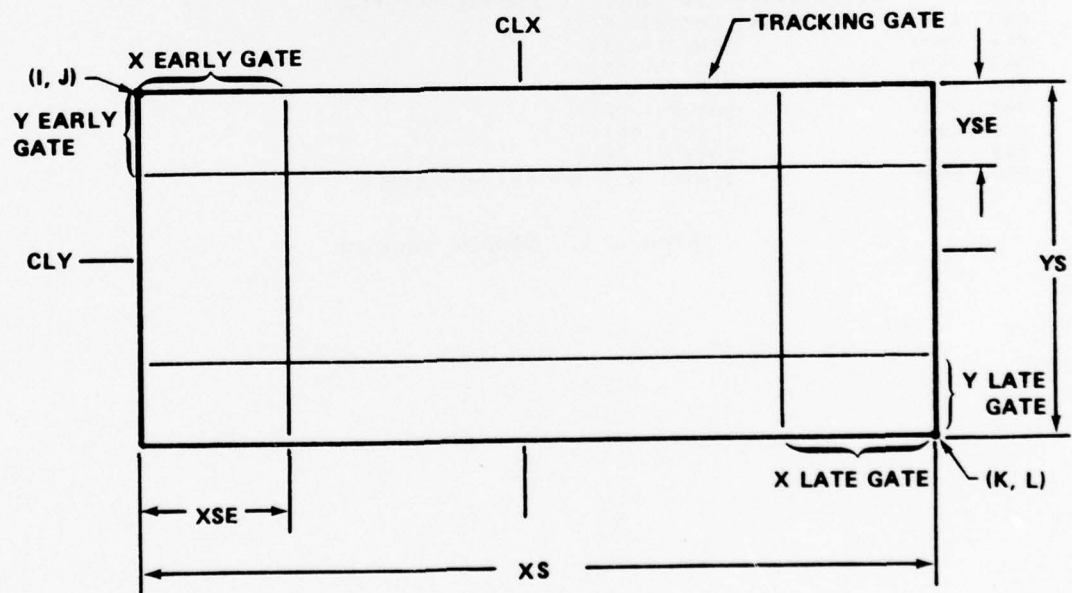


Figure 2. START routine for minimum gate.

MACRO CALLING INSTRUCTION

WINDOW

LOAD WINDOW LATCHES
TO PUT OUT GATE.

LOAD LATCH (0) WITH J (COLUMN LEFT)
LOAD LATCH (1) WITH L (COLUMN RIGHT)
LOAD LATCH (2) WITH I (ROW UP)
LOAD LATCH (3) WITH K (ROW DOWN)

RETURN

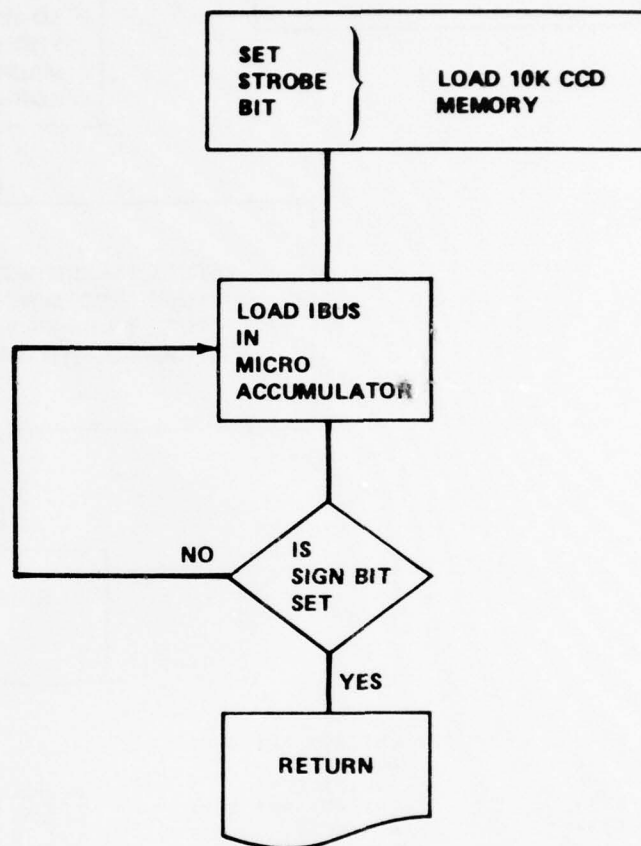
```

242  /* WINDOW----- (VARIABLE WINDOW MICROROUTINE) */
243  12H! WINDOW! LMI(R9) FF1:
244  22H! ILR(J) ROT:
245  32H! LMI(R9) FF1:
246  42H! ILR(L) ROT:
247  52H! LMI(R9) FF1:
248  62H! ILR(I) ROT:
249  6CH! LMI(R9) FF1:
250  6DH! ILR(K) ROT 17R(FETCH) PAGE1:

```

Figure 3. WINDOW routine.

**MACRO CALLING INSTRUCTION
FRAMEW**



**NOTE: THE SIGN BIT OF THE IBUS IS THE
STATUS READY FLAG FOR THE CCD 10K MEMORY**

251	1A4!	FRAME!	NOP(R4) STROBE2:
252	1AA4!	WAT!	NOP(R4) RIN:
253	1AC4!		LDT(AC) FF1:
254	1AD4!		TZP(AC) K80000:
255	1AF4!		NOP(R8) JFL(WAT,RDY):
256	1AB4!	RDY!	NOP(R8) JZR(FETCH) PAGE1:

Figure 4. FRAMEW routine.

MACRO CALLING INSTRUCTION

START INIT

INIT OCT 61
OCT 61
OCT 63
OCT 63

LOAD DATA
FROM
MACRO
ADDRESSES

STORE FIRST WORD OF DATA IN I
STORE SECOND WORD OF DATA IN J
STORE THIRD WORD OF DATA IN K
STORE FOURTH WORD OF DATA IN L

RETURN

```

257 134!      START! LMT(R9) FF1 RRM:
258 344!      ACM(AC):
259 304!      SDR(I) FF1:
260 1004!     LMT(R9) FF1 RRM:
261 1204!     ACM(AC):
262 1304!     SDR(J) FF1:
263 1404!     LMT(R9) FF1 RRM:
264 1504!     ACM(AC):
265 1604!     SDR(K) FF1:
266 1704!     LMT(R9) RRM:
267 1804!     ACM(AC):
268 1904!     SDR(L) FF1 172(FETCH) PAGE1:
    
```

BEST AVAILABLE COPY

Figure 5. Reinitialization of I, J, K, and L.

MACRO CALLING INSTRUCTION
MEAN ECOUNT

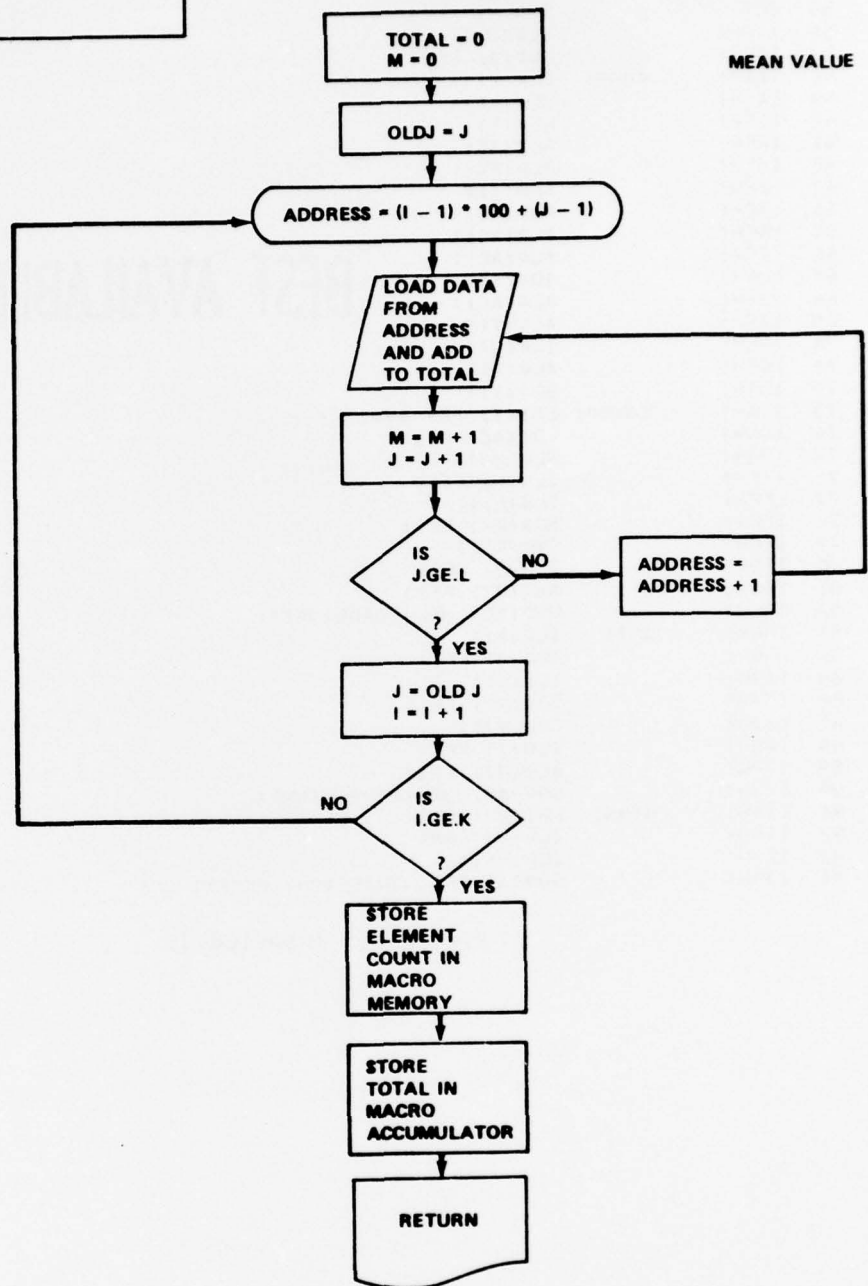


Figure 6. Calculation of mean value.

CALCULATE MEAN VALUE FOR 100 BY 100 ARRAY BEING DIGITIZED */

52	/*	
53	11H!	MEAN! CLA(AC);
54	21H!	SDR(R6) FF1;
55	2CH!	SDR(M) FF1;
56	10CH!	ILP(J);
57	11CH!	SDR(A) FF1;
58	11AH!	ADDR! ILR(I);
59	11DH!	DCA(AC);
60	11FH!	CLA(T);
61	12FH!	ALR(AC);
62	13FH!	ALR(AC);
63	14FH!	ALR(T);
64	15FH!	ALR(AC);
65	16FH!	ALP(AC);
66	17FH!	ALR(AC);
67	18FH!	ADR(T);
68	19FH!	ALP(AC);
69	1AFH!	ALR(T);
70	1BFH!	ILR(J);
71	1CEH!	ALP(T);
72	1DEH!	SDR(T);
73	1DAH!	LOADD! LMT(T) FF1 PDM;
74	1DDH!	LDI(AC) FF1;
75	1FDH!	ALR(R6);
76	1FEH!	ILR(M) FF1;
77	1FFH!	ILR(L);
78	1EFH!	SDR(R7) FF1;
79	1FEH!	CMR(R7);
80	1EDH!	ILR(J) FF1;
81	1ECH!	ALP(R7) FF1;
82	1DCH!	NOP(AC) JFL(LOADD,JGT);
83	1DBH!	JGT! ILR(A);
84	179H!	SDR(J) FF1;
85	179H!	ILP(K);
86	169H!	SDR(R7) FF1;
87	159H!	CMR(R7);
88	149H!	ILP(I) FF1;
89	139H!	ALR(R7) FF1;
90	129H!	NOP(AC) JFL(ADDR,RTNN);
91	11AH!	RTNN! LMT(R9) FF1;
92	11AH!	ILR(M) PDM;
93	12AH!	ILR(R6);
94	13AH!	SDR(A) FF1 IZP(FETCH) PAGE1;

BEST AVAILABLE COPY

Figure 6. (Concluded).

MACRO CALLING INSTRUCTION

EGATES	DELX
ADR	XG

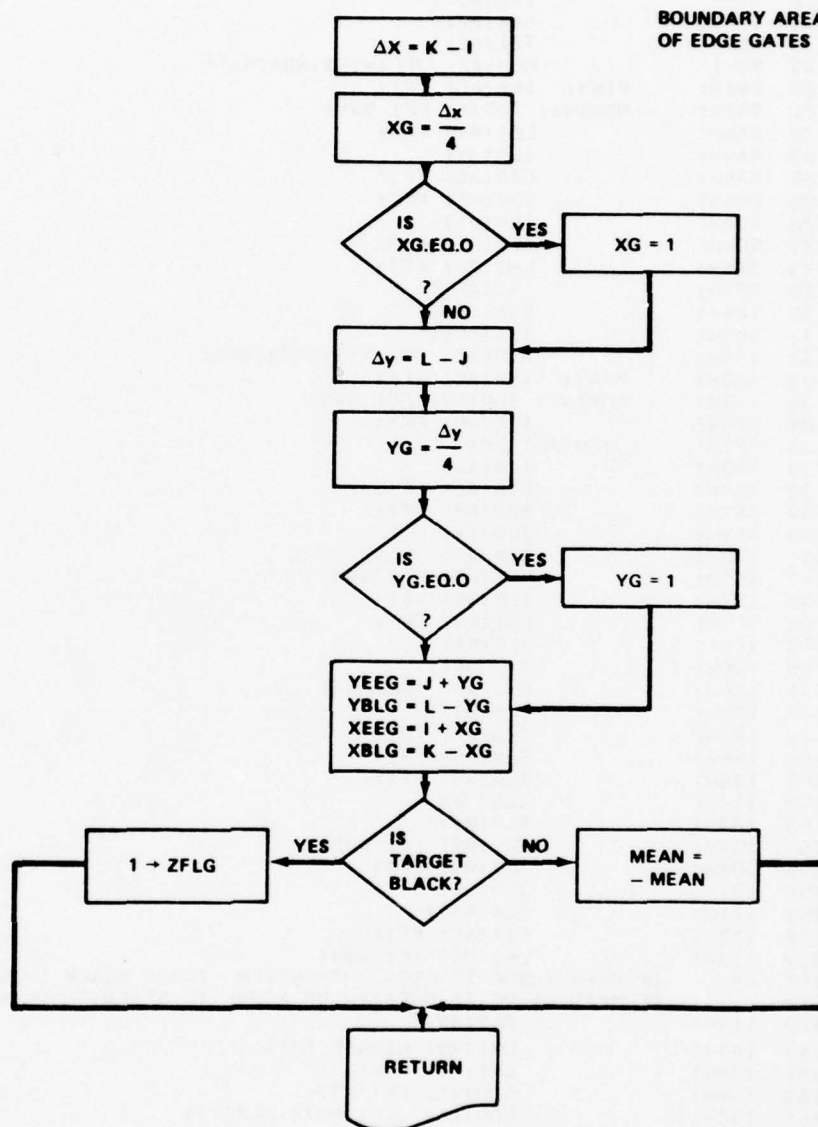


Figure 7. Setting of target flag when macro code mean value is less than seven.

```

105 /*      CALCULATE BOUNDARY AREA OF EDGE GATES      */
106 14H!      EGATFS! LMI(P) FF1 RRM:
107 24H!      ACM(AC):
108 34H!      SDR(R6) FF1:
109 44H!      LMI(R9) FF1:
110 54H!      ILR(I):
111 64H!      CIA(AC) FF1:
112 74H!      SDR(R7) FF1:
113 84H!      ILR(K):
114 85H!      ALR(R7) RWM:
115 86H!      LMI(R6) FF1:
116 87H!      SRA(AC):
117 97H!      SRA(AC):
118 96H!      TZA(AC):
119 95H!      NOP(AC) JFL(MINX,NOMINX):
120 0A2H!     MINX! INA(AC) FF1:
121 0A3H!     NOMINX! SDR(A) FF1 RWM:
122 0A4H!      LMI(R9) FF1:
123 0A5H!      ILR(J):
124 0A6H!      CIA(AC) FF1:
125 0B6H!      SDR(R7) FF1:
126 0C6H!      ILR(L):
127 0D6H!      ALR(R7) RWM:
128 0E6H!      LMI(R6) FF1:
129 0F6H!      SRA(AC):
130 106H!      SRA(AC):
131 105H!      TZA(AC):
132 115H!      NOP(AC) JFL(MINY,NOMINY):
133 122H!     MINY! INA(AC) FF1:
134 123H!     NOMINY! SDR(R7) FF1 RWM:
135 124H!      ILR(R9) FF1:
136 127H!      TLR(R9) FF1:
137 167H!      ILR(A):
138 1A7H!      CIA(AC) FF1:
139 1B7H!      SDR(R8) FF1:
140 1C7H!      ILR(I):
141 1D7H!      ALR(A):
142 1E7H!      LMI(R9) FF1 RWM:
143 1F7H!      ILR(R9) FF1:
144 1F6H!      LMI(R9) FF1:
145 1E6H!      ILR(K):
146 1D6H!      ALR(R8):
147 1C6H!      CIA(AC) FF1 RWM:
148 1B6H!      ILR(R7):
149 1B6H!      CIA(AC) FF1:
150 156H!      SDR(R8) FF1:
151 116H!      ILR(R9) FF1:
152 117H!      ILR(J):
153 1D7H!      ALR(R7):
154 1D3H!      LMI(R9) FF1 RWM:
155 1D2H!      ILR(R9) FF1:
156 1D1H!      ILR(L):
157 111H!      ALR(R8):
158 112H!      CIA(AC) FF1:
159 113H!      LMI(R9) FF1 RWM:
160 /*      IF MEAN VLAUF IS LESS THAN SEVEN TRACK BLACK ON WHITE
161          IF MEAN VALUE IS GREATER OR EQUAL TO SEVEN TRACK WHITE ON BLACK*/
162 119H!      CLR(R9):
163 109H!      LMI(R9) K0000F FF1:
164 109H!      LMI(R9) RRM:
165 148H!      ACM(AC) FF1 STZ:
166 14CH!      NOP(AC) JFL(WHITE,BLACK):
167 124H!      WHITE! CLR(R9):
168 104H!      LMI(R9) K00005 RRM:
169 108H!      LCM(AC):
170 10FH!      INA(AC) FF1 RWM PAGE1 JZR(FETCH):
171 12BH!      BLACK! NOP(AC) PAGE1 JZR(FETCH):

```

Figure 7. (Concluded).


```

273 /* CENTROID CALCULATIONS AND TARGET AREAS OF EDGE GATES
274 10H: CNTDN: CLA(AC);
275 00H: SDR(M) FF1;
276 20H: LMI(R9) FF1 RWM;
277 30H: LMI(R9) FF1;
278 40H: LMI(R9) FF1 RWM;
279 50H: LMI(R9) FF1;
280 60H: LMI(R9) FF1 RWM;
281 70H: LMI(R9) FF1;
282 80H: LMI(R9) FF1 RWM;
283 90H: LMI(R9) FF1 RWM;
284 0A0H: LMI(R9) RWM;
285 0B0H: ILR(J);
286 0C0H: SDR(A) FF1;
287 0C2H: CADDR: ILR(I);
288 0C4H: DCA(AC);
289 0D4H: CLA(T);
290 0D0H: ALR(AC);
291 0E0H: ALR(AC);
292 0F0H: ALR(T);
293 100H: ALR(AC);
294 110H: ALR(AC);
295 120H: ALR(AC);
296 130H: ADR(T);
297 140H: ALR(AC);
298 150H: ALR(T);
299 158H: ILR(J);
300 168H: ALR(T);
301 160H: SDR(T);
302 162H: CLOAD: CLR(R9);
303 166H: T7R(R9) FF7;
304 /* BRANCH TO WHITEE FOR WHITE ON BLACK TARGET
305 BRANCH TO BLACKK TO TRACK ON BLACK TARGET */
306 16CH: LMI(R9) K00005 FF1 RRM JFL(WHITEE,BLACKK);
307 178H: BLACKK: ACM(AC);
308 17CH: SDR(R6) FF1;
309 178H: LMI(T) FF1 RRM;
310 188H: LDI(AC) FF1;
311 198H: CIA(AC) FF1;
312 197H: ALR(R6) JCR(OVER);
313 17AH: WHITEE: ACM(AC);
314 170H: LMI(T) FF1 RRM;
315 180H: AIA(AC);
316 190H: OVER: NOP(AC) JFL(NODATA,DATA);
317 192H: NODATA: ILR(L);
318 191H: SDR(R6) FF1;
319 1A1H: CMR(R6);
320 1A4H: ILR(J) FF1;
321 194H: ALR(R6) FF1;
322 154H: NOP(AC) JFL(CLOAD,JGTL);
323 163H: JGTL: ILR(A);
324 165H: SDR(J) FF1;

```

Figure 8. (Continued).

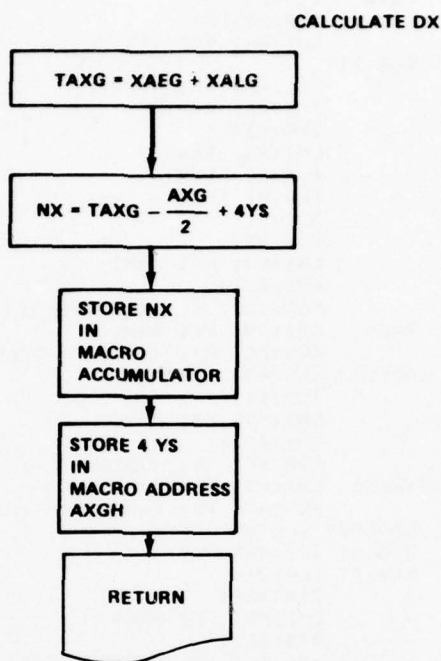

```

325 145H!      ILR(K);
326 144H!      SDR(R6) FF1;
327 104H!      CMR(R6);
328 0F4H!      ILR(I) FF1;
329 0E4H!      ALR(R6) FF1;
330 0B4H!      NOP(AC) JFL(CADDR,ENDC);
331 193H! DATA: ILR(R9);
332 195H!      SDR(R6)FF1;
333 1A5H!      LMI(R6) K0000H;
334 1A0H! ILR(T);
335 1A0H!      LMI(R6) FF1 RRM;
336 1C0H!      AMA(AC) RWM;
337 1D0H!      ILP(J);
338 1E0H!      LMI(R6) RRM;
339 1F0H!      AMA(AC) RWM;
340 1F5H!      ILR(M) FF1;
341 1E5H!      ILR(I);
342 1D5H!      CIA(AC);
343 1D4H!      LMI(R9) FF1 RRM;
344 1C4H!      AMA(AC);
345 1C5H!      NOP(AC) JFL(NXEEG,XEEG);
346 1B3H! XEEG! LMI(R9) FF1 RRM;
347 1B4H!      ACM(AC) FF1 RWM JCC(NXBLG2);
348 1B2H! NXFFG! ILR(R9) FF1;
349 1B1H!      ILR(I);
350 1A1H!      LMI(R9) FF1 RRM;
351 171H!      AMA(AC);
352 161H!      NOP(AC) JFL(NXBLG,XRLG);
353 173H! XRLG! LMI(R9) FF1 RRM;
354 175H!      ACM(AC) FF1 RWM JCR(NXBL);
355 174H! NXRLG2! ILR(R9) FF1;
356 172H! NXRLG! ILR(R9) FF1;
357 177H! NXRL! ILR(J);
358 137H!      CIA(AC);
359 136H!      LMI(R9) FF1 RRM;
360 135H!      AMA(AC);
361 134H!      NOP(AC) JFL(NYEEG,YEEG);
362 143H! YFFG! LMI(R9) FF1 RRM;
363 146H!      ACM(AC) FF1 RWM;
364 196H!      NOP(A) JCR(NODATA);
365 142H! NYFFG! ILR(R9) FF1;
366 147H!      ILR(J);
367 157H!      LMI(R9) FF1 RRM;
368 1A7H!      AMA(AC);
369 1A4H!      NOP(AC) JFL(NODAT,YRLG);
370 1A3H! YRLG! LMI(R9) FF1 RRM;
371 1A5H!      ACM(AC) FF1 RWM JCR(NODAT);
372 0C3H! ENDC! CLR(R9);
373 0C5H!      LMI(R9) K0000H;
374 0E5H!      ILR(M) RWM PAGE1 JZR(FETCH);
375 1A2H! NODAT! NOP(A) JCC(NODATA);

```

Figure 8. (Concluded).

MACRO CALLING INSTRUCTION
NGATEA AXGH



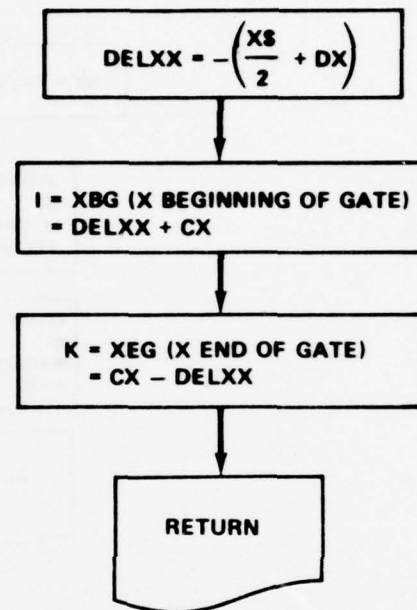
172	/*	NGATEA NGATEB NGATEC AND NGATEL CALCULATE EDGES OF THE GATE
173		AND THE NEW I J K L
174	15H!	*/
175	25H!	NGATEA! LMI(R9) RRM:
176	35H!	ACM(AC):
177	45H!	CIA(AC) FF1:
178	55H!	SDR(R6) FF1:
179	65H!	CLR(R7):
180	75H!	LMI(R7) K00007:
181	085H!	LMI(R7) FF1 RRM:
182	0ACH!	ACM(AC):
183	0DCH!	LMI(R7) FF1:
184	0DDH!	LMI(R7) RRM:
185	0CDH!	AMA(AC):
186	0HDH!	ADR(R6):
187	0ADH!	CLR(R7):
188	9DH!	LMI(R7) K00003:
189	8DH!	LMI(R7) RRM:
190	8CH!	ACM(AC):
191	9CH!	ALR(AC):
192	0ACH!	ALR(AC):
193	0CCH!	LMI(R9) FF1 RWM:
194	0CEH!	ALR(R6):
		SDR(A) FF1 IZR(FETCH) PAGE1:

Figure 9. New gate calculation of DX.

MACRO CALLING INSTRUCTION

NGATEB	DELX
ADR	CX

**CALCULATE
NEW I AND K**



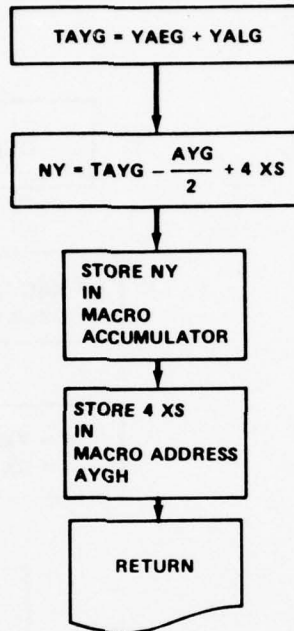
195 16H!	NGATEB! LMI(R9) RRM;
196 26H!	ACM(AC);
197 36H!	SRA(AC);
198 46H!	ALP(A);
199 56H!	SDR(A) FF1;
200 66H!	CIA(AC) FF1;
201 76H!	LMI(P) FF1 RRM;
202 79H!	LMM(T) RRM;
203 89H!	AMA(AC);
204 99H!	SDR(I) FF1;
205 0A7H!	ILR(A) RRM;
206 0B9H!	AMA(AC);
207 0C9H!	SDR(K) FF1 J7R(FETCH) PAGE1;

Figure 10. New gate calculation of I and K.

MACRO CALLING INSTRUCTION

NGATEC AYGH

CALCULATE DY



208	17H!	NGATEC! LMI(R9) RRM:
209	27H!	ACM(AC):
210	37H!	CIA(AC) FF1:
211	47H!	SDR(R6) FF1:
212	57H!	CLR(R7):
213	67H!	LMI(R7) K0000R:
214	77H!	LMI(R7) FF1 RRM:
215	0A7H!	ACM(AC):
216	0A7H!	LMI(R7) FF1:
217	0C7H!	LMI(R7) RRM:
218	0D7H!	AMA(AC):
219	0E7H!	ADR(R6):
220	0F7H!	CLR(R7):
221	0FAH!	LMI(R7) K0000P:
222	0F9H!	LMI(R7) RRM:
223	0FC4!	ACM(AC):
224	0FD4!	ALR(AC):
225	0ED4!	ALR(AC):
226	0EC4!	LMI(R9) FF1 RRM:
227	0F34!	ALR(R6):
228	0E44!	SDR(A) FF1 17R(FETCH) PAGE1:

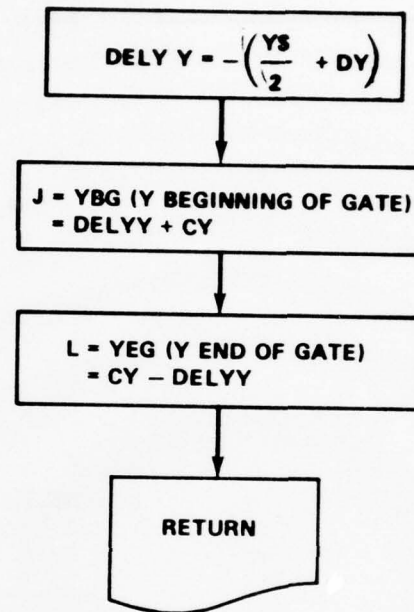
Figure 11. New gate calculation of DY.

MACRO CALLING INSTRUCTION

NGATEL
ADR

DELY
CY

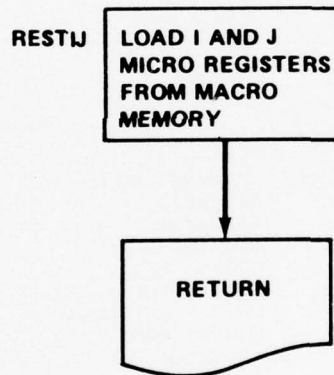
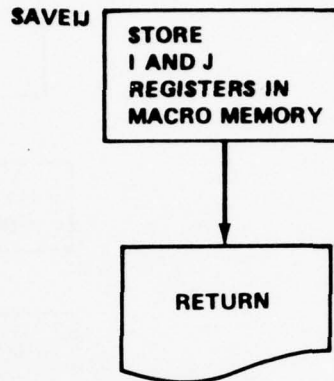
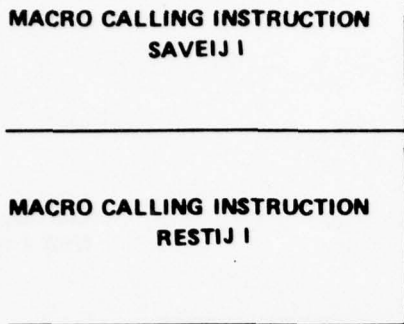
CALCULATE
NEW J AND L



229 14H!
230 24H!
231 34H!
232 44H!
233 54H!
234 64H!
235 74H!
236 84H!
237 94H!
238 0A4H!
239 0B4H!
240 0C4H!
241 0D4H!

NGATEL: LMI(P9) RRM:
ACM(AC):
SPA(AC):
ALP(A):
SDP(A) FF1:
CIA(AC) FF1:
LMI(P) FF1 RRM:
LMM(T) RRM:
AMA(AC):
SDP(J) FF1:
ILR(A) RRM:
AMA(AC):
SDR(L) FF1 IZR(FETCH) PAGE1:

Figure 12. New gate calculation of J and L.



```

95 12H!   SAVEIJ! LMI(R9) FF1:
96 32H!       ILR(I) RWM:
97 31H!       LMI(R9):
98 41H!       ILR(J) RWM 17R(FETCH) PAGE1:
  
```

```

99 13H!   RESTIJ! LMI(R9) FF1 RRM:
100 53H!       ACM(AC):
101 51H!       SDP(I) FF1:
102 61H!       LMI(R9) RRM:
103 71H!       ACM(AC):
104 81H!       SDP(J) FF1 17R(FETCH) PAGE1:
  
```

Appendix A.
TRACKER MACRO CODE LISTING

1:	00000	000001	010000		OCT	21
2:	00001	000004	000000		OCT	198
3:	00002	000001	000002	DELX	BSS	1
4:	00003	000001	000003	DELY	BSS	1
5:	00004	000001	000004	ASUM	BSS	1
6:	00005	000001	000005	MEAN2	BSS	1
7:	00006	000001	000006	XEEG	BSS	1
8:	00007	000001	000007	XAEG	BSS	1
9:	00010	000001	000010	XRLG2	BSS	1
10:	00011	000001	000011	XALG	BSS	1
11:	00012	000001	000012	YEEG	BSS	1
12:	00013	000001	000013	YAEG	BSS	1
13:	00014	000001	000014	YBLG2	BSS	1
14:	00015	000001	000015	YALG	BSS	1
15:	00016	000001	000016	XBAR	BSS	1
16:	00017	000001	000017	YBAR	BSS	1
17:	00020	000001	000020	BWF	BSS	1
19:	00100	125400	000262		ABS	100
20:	00101	000000	000275		START	INIT
21:	00102	176400	060000		LA	TEN
22:	00103	124400	000000		DO	6
23:	00104	037400	000000		WINDOW	
24:	00105	000000	000270		P	
25:	00106	176400	060000		LA	SIX
26:	00107	125400	000000		DO	6
27:	00110	000000	000273		FRAME	
28:	00111	176400	000000		LA	THREE
29:	00112	120400	000254		DO	6
30:	00113	174400	000000		MEAN	ECOUNT
31:	00114	000000	000254		EQ	
32:	00115	032400	070000		LA	ECCOUNT
33:	00116	000400	000254		ALS	7
34:	00117	000000	000000		STA	ECOUNT
35:	00120	174400	000000		CLR	
36:	00121	000400	000254		EQ	
37:	00122	033000	140000		B	ECOUNT
38:	00123	000000	000274		LRS	14
39:	00124	072000	000000		O	SEVEN
40:	00125	010000	000100		SGE	
41:	00126	030000	000000		J	BLACK
42:	00127	000400	000374		CLR	
43:	00130	000400	000000		STA	TARGET
44:	00131	010000	000136		STA	PWF
45:	00132	000000	000277	BLACK	J	WHITE
46:	00133	000400	000000		LA	OGNE
47:	00134	000000	000276		STA	BWF
48:	00135	000400	000374		LA	TWENTY
49:	00136	125400	000262	WHITE	STA	TARGET
50:	00137	000000	000275		START	INIT
51:	00140	176400	000000		LA	731
52:	00141	000000	000271	SET GAIN TO MAX	DO	5
53:	00142	176400	040000	SET OFFSET TO BLANKING LEVEL	LA	PIAS
54:	00143	000000	000270	STORE MODE LATCH FOR VIDEO	DO	4
55:	00144	000400	000374		LA	SIX
56:	00145	176400	060000		OR	TARGET
					DO	6

REQUEST FRAME AND WAIT TILL DATA READY				
57:	00146	125000	000000	FRAMEW
SET MODE LATCH FOR READ MODE				
58:	00147	000000	000266	LA ONE
59:	00150	004400	000374	OR TARGET
60:	00151	176400	000000	DO 6
61:	00152	124400	000000	WINDOW
62:	00153	121000	000255	SAVEIJ 1
63:	00154	120400	000254	MEAN ECOUNT
64:	00155	174400	000000	EQ
65:	00156	000000	000254	LA ECOUNT
66:	00157	032400	070000	ALS 7
67:	00160	000400	000254	STA ECOUNT
68:	00161	030000	000000	CLR
69:	00162	174400	000000	EQ
70:	00163	003400	000254	D ECOUNT
71:	00164	030000	140000	LRS 14
72:	00165	000400	000005	STA MEAN2
73:	00166	121400	000255	RESTIJ 1
74:	00167	122000	000002	EGATES DELX
75:	00170	000000	000256	ADR XG
76:	00171	121000	000255	SAVEIJ 1
77:	00172	120000	000007	CNTRD XAEG
78:	00173	000000	000004	LA ASUM
79:	00174	032400	070000	ALS 7
80:	00175	000400	000004	STA ASUM
81:	00176	030000	000000	CLR
82:	00177	174400	000000	EQ
83:	00200	000000	000010	LA YPAR
84:	00201	003400	000004	D ASUM
85:	00202	030000	140000	LRS 14
86:	00203	000400	000252	STA CX
87:	00204	000000	000017	LA YPAR
88:	00205	003400	000004	D ASUM
89:	00206	030000	140000	LRS 14
90:	00207	000400	000253	STA CY
91:	00210	000000	000003	LA DFLY
92:	00211	003000	000250	M XG
93:	00212	174400	000000	EQ
94:	00213	000400	000257	STA AXGH
95:	00214	000000	000002	LA DELX
96:	00215	003000	000251	M YG
97:	00216	174400	000000	EQ
98:	00217	000400	000260	STA AYGH
99:	00220	121400	000255	RESTIJ 1
100:	00221	122400	000257	NGATEA AXGH
101:	00222	174400	000000	EQ
102:	00223	000000	000257	LA AXGH
103:	00224	032400	040000	ALS 4
104:	00225	000400	000257	STA AXGH
105:	00226	030000	000000	CLR
106:	00227	174400	000000	EQ
107:	00230	003400	000257	D AXGH
108:	00231	032000	100000	ARS 10
109:	00232	120000	000002	NGATEB DELX
110:	00233	000000	000252	ADR CX
111:	00234	120400	000250	NGATEC AYGH
112:	00235	174400	000000	EQ
113:	00236	000000	000260	LA AYGH
114:	00237	032400	040000	ALS 4
115:	00240	000400	000260	STA AYGH
116:	00241	030000	000000	CLR

117:	00242	000000	000000	EQ		
118:	00243	000000	000260	D	4YGH	
119:	00244	000000	000000	ARG	10	
120:	00245	124000	000003	NGATEL	DELY	
121:	00246	000000	000253	ADR	GY	
122:	00247	010000	000143	J	START	
123:	00250	000001	000253	XG	BSS	1
124:	00251	000001	000251	YG	BSS	1
125:	00252	000001	000252	CX	BSS	1
126:	00253	000001	000253	CY	BSS	1
127:	00254	000001	000254	ECOUNT	BSS	1
128:	00255	000001	000255	I	BSS	1
129:	00256	000001	000256	J	BSS	1
130:	00257	000001	000257	AXGH	BSS	1
131:	00260	000001	000260	AYGH	BSS	1
132:	00261	000001	000261	AG2	BSS	1
133:	00262	000003	010000	INIT	OCT	61
134:	00263	000003	010000		OCT	61
135:	00264	000003	030000		OCT	63
136:	00265	000003	030000		OCT	63
137:	00266	000000	010000	ONE	OCT	1
138:	00267	000000	020000	TWO	OCT	2
139:	00270	000000	060000	SIX	OCT	6
140:	00271	000000	060000	BIAS	OCT	6
141:	00272	000001	110000	T31	OCT	31
142:	00273	000000	030000	THREE	OCT	3
143:	00274	000000	070000	SEVEN	OCT	7
144:	00275	000000	100000	TEN	OCT	10
145:	00275	000001	000000	TWENTY	OCT	20
146:	00277	177777	177777	ONE	OCT	377777
147:	00300	000000	000300	XXA	BSS	12
148:	00312	000000	000312	XXB	BSS	12
149:	00324	000000	000324	YYA	BSS	12
150:	00336	000000	000336	YYB	BSS	12
151:	00350	000000	000350	YYC	BSS	12
152:	00362	000000	000362	DMP	BSS	12
153:	00374	000001	000374	TARGET	BSS	1
154:	00375	000000	000375	G	OCT	1400000

Appendix B.
TRACKER MICRO CODE LISTING

RECORD
NUMBER

```

1
2 KRUS FIELD LENGTH=5 DEFAULT=0
3 MICRPS(KFFFF=10101B KFFFA=10101B K7FFFF=10011B
4 K80000=10010B K00FFF=10001B K000A0=10000B
5 K0000F=01111B K0000F=01110B K0000D=01101B
6 K0000C=01100B K0000R=01011B K0000A=01010B
7 K00009=01001B K0000R=01000B K00007=00111B
8 K00006=00110B K00005=00101B K00004=00100B
9 K00003=00011B K00002=00010B K00001=00001B
10 K00000=00000B);
11 KRUS KRUS:
12
13 /* ALL MICROCODE IS ON PAGE2 WITH DEFAULT SET TO 00001 FOR OTHER */
14 OTHER FIELD LENGTH=5 DEFAULT=00001B
15 MICRPS(STROBE=10000B PAGE1=00000B PAGE2=00001B
16 STROBE2=10001B);
17
18 /* DEFINITION OF BUS CONTROL FIELD */
19
20 CBUS FIELD LENGTH=3 DEFAULT=0
21 MICRPS(NB0=000B INH=001B RMW=010B CNB=011B
22 RIN=100B ROT=101B RRM=110B RWM=111B);
23
24 /*
25 NB0 NOBUS OPERATION
26 INH INHIBIT CPE ARRAY
27 RMW READ-MODIFY-WRITE
28 CNB CPU NEEDS BUS
29 RIN REQUEST INPUT
30 ROT REQUEST OUTPUT
31 RDM REQUEST READ MEMORY
32 RWM REQUEST WRITE MEMORY
33
34 SET UP SYMBOLIC REPRESENTATION OF REGISTER DESIGNATIONS */
35 A STRING 'R0';
36 I STRING 'R1';
37 J STRING 'R2';
38 P STRING 'R3';
39 K STRING 'R4';
40 L STRING 'R5';
41 TEMP STRING 'R6';
42 TEMP2 STRING 'R7';
43 M STRING 'R8';
44
45 PAUSE FIELD LENGTH=1 DEFAULT=1
46 MICRPS(PP=0);
47
48
49 OFH! FETCH! NOP(A) JPR(CNTRD MEAN SAVEIJ RESTIJ EGATES NGATEA
50 NGATEB NGATEC NGATEL WINDOW FRAME START NOTC NOTD NOTE
51 NOTF);
52 /* CALCULATE MEAN VALUE FOR 100 BY 100 ARRAY BEING DIGITIZED */
53 11H! MEAN! CLA(AC);
54 21H! SDR(R6) FF1;

```

RECORD
NUMBER

```

55 2CH!          SDR(M) FF1;
56 10CH!         ILR(J);
57 11CH!         SDR(A) FF1;
58 11AH!         ADDR! ILR(I);
59 11DH!         DCA(AC);
60 11EH!         CLA(T);
61 12EH!         ALR(AC);
62 13EH!         ALR(AC);
63 14EH!         ALR(T);
64 15EH!         ALR(AC);
65 16EH!         ALR(AC);
66 17EH!         ALR(AC);
67 18EH!         ADR(T);
68 19EH!         ALR(AC);
69 1AEH!         ALR(T);
70 1BEH!         ILR(J);
71 1CEH!         ALR(T);
72 1DEH!         SDR(T);
73 1DAH!         LOADD! LMI(T) FF1 RRM;
74 1DDH!         LDI(AC) FF1;
75 1FDH!         ALR(R6);
76 1FEH!         ILR(M) FF1;
77 1FFH!         ILR(L);
78 1EFH!         SDR(R7) FF1;
79 1EEH!         CMR(R7);
80 1EDH!         ILR(J) FF1;
81 1ECH!         ALR(R7) FF1;
82 1DCH!         NOP(AC) JFL(LOADD,JGT);
83 1DBH!         JGT!  ILR(A);
84 1D9H!         SDR(J) FF1;
85 179H!         ILR(K);
86 169H!         SDR(R7) FF1;
87 159H!         CMR(R7);
88 149H!         ILR(I) FF1;
89 139H!         ALR(R7) FF1;
90 129H!         NOP(AC) JFL(ADDR,RTNN);
91 118H!         RTNN! LMI(R9) FF1;
92 118H!         ILR(M) RWM;
93 128H!         ILR(R6);
94 138H!         SDR(A) FF1 JZR(FETCH) PAGE1;
95 12H!          SAVEIJI LMI(R9) FF1;
96 32H!          ILR(I) RWM;
97 31H!          LMI(R9);
98 41H!          ILR(J) RWM JZR(FETCH) PAGE1;
99 13H!          RESTIJI LMI(R9) FF1 RRM;
100 53H!         ACM(AC);
101 51H!         SDR(I) FF1;
102 61H!         LMI(R9) RRM;
103 71H!         ACM(AC);
104 81H!         SDR(J) FF1 JZR(FETCH) PAGE1;
105 /*          CALCULATE BOUNDARY AREA OF EDGE GATES */
106 14H!         EGATES! LMI(P) FF1 RRM;
107 24H!         ACM(AC);
108 34H!         SDR(R6) FF1;

```

RECORD
NUMBER

```

109 44H:          LMI(R9) FF1:
110 54H:          ILR(I):
111 64H:          CIA(AC) FF1:
112 74H:          SDR(R7) FF1:
113 84H:          ILR(K):
114 85H:          ALR(R7) RWM:
115 86H:          LMI(R6) FF1:
116 87H:          SRA(AC):
117 97H:          SRA(AC):
118 96H:          TZA(AC):
119 95H:          NOP(AC) JFL(MINX,NOMINX):
120 0A2H: MINX: INA(AC) FF1:
121 0A3H: NOMINX: SDR(A) FF1 RWM:
122 0A4H:          LMI(R9) FF1:
123 0A5H:          ILR(J):
124 0A6H:          CIA(AC) FF1:
125 0B6H:          SDR(R7) FF1:
126 0C6H:          ILR(L):
127 0D6H:          ALR(R7) RWM:
128 0E6H:          LMI(R6) FF1:
129 0F6H:          SRA(AC):
130 106H:          SRA(AC):
131 105H:          TZA(AC):
132 115H:          NOP(AC) JFL(MINY,NOMINY):
133 122H: MINY: INA(AC) FF1:
134 123H: NOMINY: SDR(R7) FF1 RWM:
135 124H:          ILR(R9) FF1:
136 127H:          ILR(R9) FF1:
137 167H:          ILR(A):
138 1A7H:          CIA(AC) FF1:
139 1B7H:          SDR(R8) FF1:
140 1C7H:          ILR(I):
141 1D7H:          ALR(A):
142 1E7H:          LMI(R9) FF1 RWM:
143 1F7H:          ILR(R9) FF1:
144 1F6H:          LMI(R9) FF1:
145 1E6H:          ILR(K):
146 1D6H:          ALR(R8):
147 1C6H:          CIA(AC) FF1 RWM:
148 1B6H:          ILR(R7):
149 1B6H:          CIA(AC) FF1:
150 156H:          SDR(R8) FF1:
151 116H:          ILR(R9) FF1:
152 117H:          ILR(J):
153 107H:          ALR(R7):
154 103H:          LMI(R9) FF1 RWM:
155 102H:          ILR(R9) FF1:
156 101H:          ILR(L):
157 111H:          ALR(R8):
158 112H:          CIA(AC) FF1:
159 113H:          LMI(R9) FF1 RWM:
160 /*          IF MEAN VLAUE IS LESS THAN SEVEN TRACK BLACK ON WHITE
161              IF MEAN VALUE IS GREATER OR EQUAL TO SEVEN TR*CK WHITE ON BLACK*/
162 119H:          CLR(R9):

```

RECORD
NUMBER

```

163 109H!      LMI(R9) K0000F FF1;
164 108H!      LMI(R9) RRM;
165 148H!      ACM(AC) FF1 STZ;
166 14CH!      NOP(AC) JFL(WHITE,BLACK);
167 12AH!      WHITE! CLR(R9);
168 10AH!      LMI(R9) K00005 RRM;
169 10BH!      LCM(AC);
170 10EH!      INA(AC) FF1 RWM PAGE1 JZR(FETCH);
171 12BH!      BLACK! NOP(AC) PAGE1 JZR(FETCH);
172 /*        NGATEA NGATEB NGATEC AND NGATEL CALCULATE EDGES OF THE GATE
173              AND THE NEW I J K L          */
174 15H!      NGATFA! LMI(R9) RRM;
175 25H!      ACM(AC);
176 35H!      CIA(AC) FF1;
177 45H!      SDR(R6) FF1;
178 55H!      CLR(R7);
179 65H!      LMI(R7) K00007;
180 75H!      LMI(R7) FF1 RRM;
181 085H!     ACM(AC);
182 08CH!     LMI(R7) FF1;
183 0DCH!     LMI(R7) RRM;
184 0DDH!     AMA(AC);
185 0CDH!     ADR(R6);
186 0BDH!     CLR(R7);
187 0ADH!     LMI(R7) K00003;
188 9DH!      LMI(R7) RRM;
189 8DH!      ACM(AC);
190 8CH!      ALR(AC);
191 9CH!      ALR(AC);
192 0ACH!     LMI(R9) FF1 RWM;
193 0CCH!     ALR(R6);
194 0CEH!     SDR(A) FF1 IZR(FETCH) PAGE1;
195 16H!      NGATFB! LMI(R9) RRM;
196 26H!      ACM(AC);
197 36H!      SRA(AC);
198 46H!      ALR(A);
199 56H!      SDR(A) FF1;
200 66H!      CIA(AC) FF1;
201 76H!      LMI(P) FF1 RRM;
202 79H!      LMM(T) RRM;
203 89H!      AMA(AC);
204 99H!      SDR(I) FF1;
205 0A9H!     ILR(A) RRM;
206 0B9H!     AMA(AC);
207 0C9H!     SDR(K) FF1 JZR(FETCH) PAGE1;
208 17H!      NGATEC! LMI(R9) RRM;
209 27H!      ACM(AC);
210 37H!      CIA(AC) FF1;
211 47H!      SDR(R6) FF1;
212 57H!      CLR(R7);
213 67H!      LMI(R7) K0000B;
214 77H!      LMI(R7) FF1 RRM;
215 0A7H!     ACM(AC);
216 0B7H!     LMI(R7) FF1;

```


RECORD
NUMBER

```

217 0C74!      LMI(R7) RRM:
218 0D74!      AMA(AC):
219 0E74!      ADR(R6):
220 0F74!      CLR(R7):
221 0F84!      LMI(R7) K00002:
222 0F94!      LMI(R7) RRM:
223 0FCH!      ACM(AC):
224 0FD4!      ALR(AC):
225 0ED4!      ALR(AC):
226 0EC4!      LMI(R9) FF1 RRM:
227 0E94!      ALR(R6):
228 0E84!      SDR(A) FF1 JZR(FETCH) PAGE1:
229 18H!      NGATEL! LMI(R9) RRM:
230 24H!      ACM(AC):
231 34H!      SRA(AC):
232 48H!      ALR(A):
233 58H!      SDR(A) FF1:
234 68H!      CIA(AC) FF1:
235 78H!      LMI(P) FF1 RRM:
236 88H!      LMM(T) RRM:
237 98H!      AMA(AC):
238 0A84!      SDR(J) FF1:
239 0B84!      ILR(A) RRM:
240 0C84!      AMA(AC):
241 0D84!      SDR(L) FF1 JZR(FETCH) PAGE1:
242 /*      WINDOW----- (VARIABLE WINDOW MICROROUTINE) */
243 19H!      WINDOW! LMI(R9) FF1:
244 29H!      ILR(J) ROT:
245 39H!      LMI(R9) FF1:
246 49H!      ILR(L) ROT:
247 59H!      LMI(R9) FF1:
248 69H!      ILR(I) ROT:
249 6CH!      LMI(R9) FF1:
250 6DH!      ILR(K) ROT JZR(FETCH) PAGE1:
251 1AH!      FRAME! NOP(R8) STROBE2:
252 1AA4!      WAT!   NOP(R8) RIN:
253 1ACH!      LDI(AC) FF1:
254 1AD4!      TZR(AC) K80000:
255 1AF4!      NOP(R8) JFL(WAT,RJY):
256 1AB4!      RNY!   NOP(R8) JZR(FETCH) PAGE1:
257 1BH!      START! LMI(R9) FF1 RRM:
258 38H!      ACM(AC):
259 3DH!      SDR(I) FF1:
260 10DH!      LMI(R9) FF1 RRM:
261 12DH!      ACM(AC):
262 13DH!      SDR(J) FF1:
263 14DH!      LMI(R9) FF1 RRM:
264 15DH!      ACM(AC):
265 16DH!      SDR(K) FF1:
266 17DH!      LMI(R9) RRM:
267 18DH!      ACM(AC):
268 19DH!      SDR(L) FF1 JZR(FETCH) PAGE1:
269 1CH!      NOTC!  NOP(A) JZR(FETCH) PAGE1:
270 1DH!      NOTD!  NOP(A) JZR(FETCH) PAGE1:

```

RECORD
NUMBER

```

271 1EH!      NOTF!  NOP(A) JZR(FETCH) PAGE1;
272 1FH!      NOTF!  NOP(A) JZR(FETCH) PAGE1;
273 /*      CENTROID CALCULATIONS AND TARGET AREAS OF EDGE GATES      */
274 10H!      CNTPD! CLA(AC);
275 00H!      SDR(M) FF1;
276 20H!      LMI(R9) FF1 RWM;
277 30H!      LMI(R9) FF1;
278 40H!      LMI(R9) FF1 RWM;
279 50H!      LMI(R9) FF1;
280 60H!      LMI(R9) FF1 RWM;
281 70H!      LMI(R9) FF1;
282 80H!      LMI(R9) FF1 RWM;
283 90H!      LMI(R9) FF1 RWM;
284 0A0H!     LMI(R9) RWM;
285 0B0H!     ILR(J);
286 0C0H!     SDR(A) FF1;
287 0C2H!     CADDR! ILR(I);
288 0C4H!     DCA(AC);
289 0D4H!     CLA(T);
290 0D0H!     ALR(AC);
291 0E0H!     ALR(AC);
292 0F0H!     ALR(T);
293 100H!     ALR(AC);
294 110H!     ALR(AC);
295 120H!     ALR(AC);
296 130H!     ADR(T);
297 140H!     ALR(AC);
298 150H!     ALR(T);
299 158H!     ILR(J);
300 168H!     ALR(T);
301 160H!     SDR(T);
302 162H!     CLOAD! CLR(R9);
303 166H!     TZR(R9) FF1;
304 /*      BRANCH TO WHITEE FOR WHITE ON BLACK TARGET
305          BRANCH TO BLACKK TO TRACK ON BLACK TARGET      */
306 16CH!     LMI(R9) K00005 FF1 RRM JFL(WHITEE,BLACKK);
307 17BH!     BLACKK! ACM(AC);
308 17CH!     SDR(R6) FF1;
309 178H!     LMI(T) FF1 RRM;
310 188H!     LDI(AC) FF1;
311 198H!     CIA(AC) FF1;
312 197H!     ALR(R6) JCR(OVER);
313 17AH!     WHITEE! ACM(AC);
314 170H!     LMI(T) FF1 RRM;
315 180H!     AIA(AC);
316 190H!     OVER! NOP(AC) JFL(NODATA,DATA);
317 192H!     NODATA! ILR(L);
318 191H!     SDR(R6) FF1;
319 1A1H!     CMR(R6);
320 1A4H!     ILR(J) FF1;
321 194H!     ALR(R6) FF1;
322 154H!     NOP(AC) JFL(CLOAD,JGTL);
323 163H!     JGTL! ILR(A);
324 165H!     SDR(J) FF1;

```

RECORD
NUMBER

```

325 145H!      ILR(K);
326 144H!      SDR(R6) FF1;
327 104H!      CMR(R6);
328 0F4H!      ILR(I) FF1;
329 0E4H!      ALR(R6) FF1;
330 084H!      NOP(AC) JFL(CADDR,ENDC);
331 193H!      DATA! ILR(R9);
332 195H!      SDR(R6)FF1;
333 1A5H!      LMI(R6) K00009;
334 1A0H!      ILR(I);
335 1B0H!      LMI(R6) FF1 RRM;
336 1C0H!      AMA(AC) RWM;
337 100H!      ILR(J);
338 1E0H!      LMI(R6) RRM;
339 1F0H!      AMA(AC) RWM;
340 1F5H!      ILR(M) FF1;
341 1E5H!      ILR(I);
342 1D5H!      CIA(AC);
343 1D4H!      LMI(R9) FF1 RRM;
344 1C4H!      AMA(AC);
345 1C5H!      NOP(AC) JFL(NXEEG,XEEG);
346 1B3H!      XEEG! LMI(R9) FF1 RRM;
347 1B4H!      ACM(AC) FF1 RWM JCC(NXBLG2);
348 1B2H!      NXEEG! ILR(R9) FF1;
349 1B1H!      ILR(I);
350 1B1H!      LMI(R9) FF1 RRM;
351 171H!      AMA(AC);
352 161H!      NOP(AC) JFL(NXBLG,XBLG);
353 173H!      XBLG! LMI(R9) FF1 RRM;
354 175H!      ACM(AC) FF1 RWM JCR(NXBL);
355 174H!      NXBLG2! ILR(R9) FF1;
356 172H!      NXBLG! ILR(R9) FF1;
357 177H!      NXBL! ILR(J);
358 137H!      CIA(AC);
359 136H!      LMI(R9) FF1 RRM;
360 135H!      AMA(AC);
361 134H!      NOP(AC) JFL(NYEEG,YEEG);
362 143H!      YFEG! LMI(R9) FF1 RRM;
363 146H!      ACM(AC) FF1 RWM;
364 196H!      NOP(A) JCR(NODATA);
365 142H!      NYEEG! ILR(R9) FF1;
366 147H!      ILR(J);
367 157H!      LMI(R9) FF1 RRM;
368 187H!      AMA(AC);
369 184H!      NOP(AC) JFL(NODAT,YRLG);
370 183H!      YRLG! LMI(R9) FF1 RRM;
371 185H!      ACM(AC) FF1 RWM JCR(NODAT);
372 0C3H!      ENDC! CLR(R9);
373 0C5H!      LMI(R9) K00004;
374 0E5H!      ILR(M) RWM PAGE1 JZR(FETCH);
375 182H!      NODAT! NOP(A) JCC(NODATA);
376 EOF

```

NO PROGRAM ERRORS

XMAS VERS 2.0 OPTICAL CONTRAST VARIABLE GATE CENTROID TRACK ERRORS= 0 PAGE 9

END OF PROGRAM

MICROPROGRAM MEMORY IMAGE

	0H	1H	2H	3H	4H	5H	6H	7H	
=====									
000H	JCC	*	*	*	*	*	*	*	=
	0020H	*	*	*	*	*	*	*	=
	275	*	*	*	*	*	*	*	=
	1	*	*	*	*	*	*	*	=
=====									
001H	JCC	JCC	JCC	JCC	JCC	JCC	JCC	JCC	=
	0000H	0021H	0032H	0053H	0024H	0025H	0026H	0027H	=
	274	53	95	99	106	174	195	208	=
	1	1	1	1	1	1	1	1	=
=====									
002H	JCC	JCR	*	*	JCC	JCC	JCC	JCC	=
	0030H	002CH	*	*	0034H	0035H	0036H	0037H	=
	276	54	*	*	107	175	196	209	=
	1	1	*	*	1	1	1	1	=
=====									
003H	JCC	JCC	JCR	*	JCC	JCC	JCC	JCC	=
	0040H	0041H	0031H	*	0044H	0045H	0046H	0047H	=
	277	97	96	*	108	176	197	210	=
	1	1	1	*	1	1	1	1	=
=====									
004H	JCC	JZR	*	*	JCC	JCC	JCC	JCC	=
	0050H	000FH	*	*	0054H	0055H	0056H	0057H	=
	278	98	*	*	109	177	198	211	=
	1	1	*	*	1	1	1	1	=
=====									
005H	JCC	JCC	*	JCR	JCC	JCC	JCC	JCC	=
	0060H	0061H	*	0051H	0064H	0065H	0066H	0067H	=
	279	101	*	100	110	178	199	212	=
	1	1	*	1	1	1	1	1	=
=====									
006H	JCC	JCC	*	*	JCC	JCC	JCC	JCC	=
	0070H	0071H	*	*	0074H	0075H	0076H	0077H	=
	280	102	*	*	111	179	200	213	=
	1	1	*	*	1	1	1	1	=
=====									
007H	JCC	JCC	*	*	JCC	JCC	JCR	JCC	=
	0080H	0081H	*	*	0084H	0085H	0079H	00A7H	=
	281	103	*	*	112	180	201	214	=
	1	1	*	*	1	1	1	1	=
=====									

MICROPROGRAM MEMORY IMAGE

	0H	1H	2H	3H	4H	5H	6H	7H
008H	JCC 0090H 282 1	JZR 000FH 104 1	*	*	JCR 0085H 113 1	JCR 00A6H 114 1	JCR 0087H 115 1	JCC 0097H 116 1
009H	JCC 00A0H 283 1	*	*	*	*	JFL 00A2H 119 1	JCR 0095H 118 1	JCR 0096H 117 1
00AH	JCC 00B0H 284 1	*	JCR 00A3H 120 1	JCR 00A4H 121 2	JCR 00A5H 122 1	JCR 00A6H 123 1	JCC 00B6H 124 1	JCC 00B7H 215 1
00BH	JCC 00C0H 285 1	*	*	*	JFL 00C2H 330 1	JCR 00A6H 181 1	JCC 00C6H 125 1	JCC 00C7H 216 1
00CH	JCR 00C2H 286 1	*	JCR 00C4H 287 2	JCR 00C5H 372 1	JCC 00D4H 288 1	JCC 00E5H 373 1	JCC 00D6H 126 1	JCC 00D7H 217 1
00DH	JCC 00E0H 290 1	*	*	*	JCR 00D0H 289 1	*	JCC 00E6H 127 1	JCC 00E7H 218 1
00EH	JCC 00F0H 291 1	*	*	*	JCC 00B4H 329 1	JZR 000FH 374 1	JCC 00F6H 128 1	JCC 00F7H 219 1
00FH	JCC 0100H 292 1	*	*	*	JCC 00E4H 328 1	*	JCC 0106H 129 1	JCR 00F8H 220 1

MICROPROGRAM MEMORY IMAGE

	0H	1H	2H	3H	4H	5H	6H	7H
010H	JCC 0110H 293 1	JCC 0111H 156 1	JCR 0101H 155 1	JCR 0102H 154 1	JCC 00F4H 327 1	JCC 0115H 131 1	JCR 0105H 130 1	JCR 0103H 153 1
011H	JCC 0120H 294 1	JCR 0112H 157 1	JCR 0113H 158 1	JCR 0119H 159 1		JFL 0122H 132 1	JCR 0117H 151 1	JCC 0107H 152 1
012H	JCC 0130H 295 1		JCR 0123H 133 1	JCR 0124H 134 2	JCR 0127H 135 1			JCC 0167H 136 1
013H	JCC 0140H 296 1				JFL 0142H 361 1	JCR 0134H 360 1	JCR 0135H 359 1	JCR 0136H 358 1
014H	JCC 0150H 297 1		JCR 0147H 365 1	JCR 0146H 362 1	JCC 0104H 326 1	JCR 0144H 325 1	JCC 0196H 363 1	JCC 0157H 366 1
015H	JCR 0158H 298 1				JFL 0162H 322 1		JCC 0116H 150 1	JCC 0187H 367 1
016H	JCR 0162H 301 1	JFL 0172H 352 1	JCR 0166H 302 2	JCR 0165H 323 1		JCC 0145H 324 1	JCR 0160H 303 1	JCC 01A7H 137 1
017H	JCC 0180H 314 1	JCC 0161H 351 1	JCR 0177H 356 2	JCR 0175H 353 1	JCR 0172H 355 1	JCR 0177H 354 1		JCC 0137H 357 2

MICROPROGRAM MEMORY IMAGE

	0H	1H	2H	3H	4H	5H	6H	7H
018H	JCC 0190H 315 1	JCC 0171H 350 1	JCC 0192H 375 2	JCR 0185H 370 1	JFL 0182H 369 1	JCR 0182H 371 1	JCC 0156H 149 1	JCR 0184H 368 1
019H	JFL 0192H 316 2	JCC 01A1H 318 1	JCR 0191H 317 3	JCR 0195H 331 1	JCC 0154H 321 1	JCC 01A5H 332 1	JCR 0192H 364 1	JCR 0190H 312 1
01AH	JCC 0180H 334 1	JCR 01A4H 319 1			JCC 0194H 320 1	JCR 01A0H 333 1		JCC 0187H 138 1
01BH	JCC 01C0H 335 1	JCC 0181H 349 1	JCR 0181H 348 1	JCR 0184H 346 1	JCC 0174H 347 1		JCC 0186H 148 1	JCC 01C7H 139 1
01CH	JCC 01D0H 336 1				JCR 01C5H 344 1	JFL 01A2H 345 1	JCC 01B6H 147 1	JCC 01D7H 140 1
01DH	JCC 01E0H 337 1				JCC 01C4H 343 1	JCR 01D4H 342 1	JCC 01C6H 146 1	JCC 01E7H 141 1
01EH	JCC 01F0H 338 1					JCC 01D5H 341 1	JCC 01D6H 145 1	JCC 01F7H 142 1
01FH	JCR 01F5H 339 1					JCC 01E5H 340 1	JCC 01E6H 144 1	JCR 01F6H 143 1

MICROPROGRAM MEMORY IMAGE

	8H	9H	AH	BH	CH	DH	EH	FH
=	.	.	*	*	*	.	.	JPR
=	.	.	*	*	*	.	.	0010H
000H	.	.	*	*	*	.	.	.
=	.	.	*	*	*	.	.	49
=	.	.	*	*	*	.	.	17
=	JCC	JCC	JCC	JCC	JZR	JZR	JZR	JZR
=	0028H	0029H	01AAH	0038H	000FH	000FH	000FH	000FH
001H	.	.	*	*	*	.	.	.
=	229	243	251	257	269	270	271	272
=	1	1	1	1	1	1	1	1
=	JCC	JCC	.	.	JCC	.	.	.
=	0038H	0039H	.	.	010CH	.	.	.
002H	.	.	*	*	*	.	.	.
=	230	244	.	.	55	.	.	.
=	1	1	.	.	1	.	.	.
=	JCC	JCC	.	JCR	.	JCC	.	.
=	0048H	0049H	.	003DH	.	010DH	.	.
003H	.	.	*	*	.	.	.	.
=	231	245	.	258	.	259	.	.
=	1	1	.	1	.	1	.	.
=	JCC	JCC	.	.	.	.	.	.
=	0058H	0059H	.	.	.	.	.	.
004H	.	.	*	*	.	.	.	.
=	232	246	.	.	.	.	.	.
=	1	1	.	.	.	.	.	.
=	JCC	JCC	.	.	.	.	.	.
=	0068H	0069H	.	.	.	.	.	.
005H	.	.	*	*	.	.	.	.
=	233	247	.	.	.	.	.	.
=	1	1	.	.	.	.	.	.
=	JCC	JCR	.	.	JCR	JZR	.	.
=	0078H	006CH	.	.	006DH	000FH	.	.
006H	.	.	*	*	*	.	.	.
=	234	248	.	.	249	250	.	.
=	1	1	.	.	1	1	.	.
=	JCC	JCC	.	.	.	.	.	.
=	0088H	0089H	.	.	.	.	.	.
007H	.	.	*	*	.	.	.	.
=	235	202	.	.	.	.	.	.
=	1	1	.	.	.	.	.	.

MICROPROGRAM MEMORY IMAGE

	8H	9H	AH	BH	CH	DH	EH	FH
008H	JCC	JCC	*	*	JCC	JCR	*	*
=	0098H	0099H	*	*	009CH	008CH	*	*
=	236	203	*	*	190	189	*	*
=	1	1	*	*	1	1	*	*
009H	JCC	JCC	*	*	JCC	JCC	*	*
=	00A8H	00A9H	*	*	00ACH	008DH	*	*
=	237	204	*	*	191	188	*	*
=	1	1	*	*	1	1	*	*
00AH	JCC	JCC	*	*	JCC	JCC	*	*
=	00B8H	00B9H	*	*	00CCH	009DH	*	*
=	23A	205	*	*	192	187	*	*
=	1	1	*	*	1	1	*	*
00BH	JCC	JCC	*	*	JCC	JCC	*	*
=	00C8H	00C9H	*	*	00DCH	00ADH	*	*
=	239	206	*	*	182	186	*	*
=	1	1	*	*	1	1	*	*
00CH	JCC	JZR	*	*	JCR	JCC	JZR	*
=	00D8H	000FH	*	*	00CEH	008DH	000FH	*
=	240	207	*	*	193	185	194	*
=	1	1	*	*	1	1	1	*
00DH	JZR	*	*	*	JCR	JCC	*	*
=	000FH	*	*	*	00DDH	00CDH	*	*
=	241	*	*	*	183	184	*	*
=	1	*	*	*	1	1	*	*
00EH	JZR	JCR	*	*	JCR	JCR	*	*
=	000FH	00E8H	*	*	00E9H	00ECH	*	*
=	228	227	*	*	226	225	*	*
=	1	1	*	*	1	1	*	*
00FH	JCR	JCR	*	*	JCR	JCC	*	*
=	00F9H	00FCH	*	*	00FDH	00FDH	*	*
=	221	222	*	*	223	224	*	*
=	1	1	*	*	1	1	*	*

MICROPROGRAM MEMORY IMAGE

	8H	9H	AH	BH	CH	DH	EH	FH
010H	JCC 0148H 164 1	JCR 0108H 163 1	JCR 0108H 168 1	JCR 010EH 169 1	JCC 011CH 56 1	JCC 012DH 260 1	JZR 000FH 170 1	
011H	JCC 0128H 92 1	JCC 0109H 162 1	JCR 011DH 58 2	JCR 0118H 91 1	JCR 011AH 57 1	JCR 011EH 59 1	JCC 012EH 60 1	
012H	JCC 0138H 93 1	JFL 011AH 90 1	JCC 010AH 167 1	JZR 000FH 171 1		JCC 013DH 261 1	JCC 013EH 61 1	
013H	JZR 000FH 94 1	JCC 0129H 89 1				JCC 014DH 262 1	JCC 014EH 62 1	
014H	JCR 014CH 165 1	JCC 0139H 88 1			JFL 012AH 166 1	JCC 015DH 263 1	JCC 015EH 63 1	
015H	JCC 0168H 299 1	JCC 0149H 87 1				JCC 016DH 264 1	JCC 016EH 64 1	
016H	JCR 0160H 300 1	JCC 0159H 86 1			JFL 017AH 306 1	JCC 017DH 265 1	JCC 017EH 65 1	
017H	JCC 0188H 309 1	JCC 0169H 85 1	JCR 0170H 313 1	JCR 017CH 307 1	JCR 0178H 308 1	JCC 018DH 266 1	JCC 018EH 66 1	

MICROPROGRAM MEMORY IMAGE

	8H	9H	AH	BH	CH	DH	EH	FH
=	JCC	.	.	.	.	JCC	JCC	.
=	0198H	.	.	.	.	019DH	019EH	.
018H	.	.	.	.	.	.	.	.
=	310	.	.	.	.	267	67	.
=	1	.	.	.	.	1	1	.
=	JCR	.	.	.	.	JZP	JCC	.
=	0197H	.	.	.	.	000FH	01AEH	.
019H	.	.	.	.	.	.	.	.
=	311	.	.	.	.	268	68	.
=	1	.	.	.	.	1	1	.
=	.	.	JCR	JZR	JCR	JCR	JCC	JFL
=	.	.	01ACH	000FH	01ADH	01AFH	01BEH	01AAH
01AH	.	.	.	.	.	.	.	.
=	.	.	252	256	253	254	69	255
=	.	.	2	1	1	1	1	1
=	.	.	.	.	.	.	JCC	.
=	.	.	.	.	.	.	01CEH	.
01BH	.	.	.	.	.	.	.	.
=	.	.	.	.	.	.	70	.
=	.	.	.	.	.	.	1	.
=	.	.	.	.	.	.	JCC	.
=	.	.	.	.	.	.	01DEH	.
01CH	.	.	.	.	.	.	.	.
=	.	.	.	.	.	.	71	.
=	.	.	.	.	.	.	1	.
=	JCC	JCR	JCR	JFL	JCC	JCR	.	.
=	0179H	01DDH	01D9H	01DAH	01FDH	01DAH	.	.
01DH	.	.	.	.	.	.	.	.
=	84	73	83	82	74	72	.	.
=	1	2	1	1	1	1	.	.
=	.	.	.	JCC	JCR	JCR	JCR	.
=	.	.	.	01DCH	01ECH	01EDH	01EEH	.
01EH	.	.	.	.	.	.	.	.
=	.	.	.	81	80	79	78	.
=	.	.	.	1	1	1	1	.
=	.	.	.	.	JCR	JCR	JCC	.
=	.	.	.	.	01FEH	01FFH	01EFH	.
01FH	.	.	.	.	.	.	.	.
=	.	.	.	.	75	76	77	.
=	.	.	.	.	1	1	1	.

CROSS REFERENCE DIRECTORY

LABEL	REFERENCES
ADDR	(58), 90
BLACK	166, (171)
BLACKK	306, (307)
CADDR	(287), 330
CLOAD	(302), 322
CNTRD	49, (274)
DATA	316, (331)
EGATES	49, (106)
ENDC	330, (372)
FETCH	(49), 94, 98, 104, 170, 171, 194, 207, 228, 241, 250, 256, 268, 269, 270, 271, 272, 374
FRAME	50, (251)
JGT	82, (83)
JGTL	322, (323)
LOADD	(73), 82
MEAN	49, (53)
MINX	119, (120)
MINY	132, (133)
NGATEA	49, (174)
NGATEB	50, (195)
NGATEC	50, (208)
NGATEL	50, (229)
NODAT	369, 371, (375)
NODATA	316, (317), 364, 375
NOMINX	119, (121)
NOMINY	132, (134)
NOTC	50, (269)
NOTD	50, (270)
NOTE	50, (271)
NOTF	51, (272)
NXBL	354, (357)
NXBLG	352, (356)
NXBLG2	347, (355)
NXEEG	345, (348)
NYEEG	361, (365)
OVER	312, (316)
RDY	255, (256)
RESTIJ	49, (99)
RTNN	90, (91)
SAVEIJ	49, (95)
START	50, (257)
WAT	(252), 255
WHITE	166, (167)
WHITEE	306, (313)
WINDOW	50, (243)
XBLG	352, (353)
XEEG	345, (346)
YBLG	369, (370)
YEEG	361, (362)

Appendix C.
TRACKER MICRO CODE LISTING WITH EXPANDED BITS

```

RECORD      CPE  FI  FO  JUMP  KBUSS  OTHER  CBUS  PAUSE
NUMBER      6543210  10  10  6543210  43210  43210  210  0

1
2 KBUSS      FIELD LENGTH=5  DEFAULT=0
3             MICROPS(KFFFFFF=10101B  KFFFFFFA=10101B  K7FFFF=10011B
4             K80000=10010B  K00FFF=10001B  K00060=10000B
5             K0000F=01111B  K0000E=01110B  K0000D=01101B
6             K0000C=01100B  K0000B=01011B  K0000A=01010B
7             K00009=01001B  K00008=01000B  K00007=00111B
8             K00006=00110B  K00005=00101B  K00004=00100B
9             K00003=00011B  K00002=00010B  K00001=00001B
10            K00000=00000B);
11 KBUSS      KBUS:
12
13 /* ALL MICROCODE IS ON PAGE2 WITH DEFAULT SET TO 0001 FOR OTHER */
14 OTHER      FIELD LENGTH=5  DEFAULT=00001B
15             MICROPS(STROBE=10000B  PAGE1=00000B  PAGE2=00001B
16             STROBE2=10001B);
17
18 /* DEFINITION OF BUS CONTROL FIELD          */
19
20 CBUS      FIELD LENGTH=3  DEFAULT=0
21             MICROPS(NB0=000B  INH=001B  RMW=010B  CNB=011B
22             RIN=100B  ROT=101B  RRM=110B  RWM=111B);
23
24 /*      NRO      NOBUS OPERATION
25             INH      INHIBIT CPE ARRAY
26             RMW      READ-MODIFY-WRITE
27             CNB      CPU NEEDS BUS
28             RIN      REQUEST INPUT
29             ROT      REQUEST OUTPUT
30             RDM      REQUEST READ MEMORY
31             RWM      REQUEST WRITE MEMORY
32
33             SFT (IP SYMBOLIC REPRESENTATION OF REGISTER DESIGNATIONS */
34
35 A          STRING 'R0';
36 I          STRING 'R1';
37 J          STRING 'R2';
38 P          STRING 'R3';
39 K          STRING 'R4';
40 L          STRING 'R5';
41 TEMP      STRING 'R6';
42 TEMP2     STRING 'R7';
43 M          STRING 'R8';
44
45 PAUSE      FIELD LENGTH=1  DEFAULT=1
46             MICROPS(PP=0);
47
48
49 OFH:      FETCH: NOP(A) JPR(CNTRD MEAN SAVEIJ RESTIJ EGATES NGATEA
50             NGATER NGATEC NGATEL WINDOW FRAME START NOTC NOTD NOTE
51             NOTF);
52 (000FH) 110000 11 00 1100001 00000 00001 000 1

```


RECORD NUMBER		CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
52	/*	CALCULATE MEAN VALUE FOR 100 BY 100 ARRAY BEING DIGITIZED */							
53	11H; (0011H)	MEAN; 1001011	CLA(AC); 11	00	0000010	00000	00001	000	1
54	21H; (0021H)	SDR(R6) FF1; 0100110	11	11	0111100	11111	00001	000	1
55	2CH; (002CH)	SDR(M) FF1; 0101000	11	11	0010000	11111	00001	000	1
56	10CH; (010CH)	ILR(J); 0000010	11	00	0010001	00000	00001	000	1
57	11CH; (011CH)	SDR(A) FF1; 0100000	11	11	0111010	11111	00001	000	1
58	11AH; (011AH)	ADDR; ILR(I); 0000001	11	00	0111101	00000	00001	000	1
59	11DH; (011DH)	DCA(AC); 0011111	11	00	0111110	11111	00001	000	1
60	11EH; (011EH)	CLA(T); 1001010	11	00	0010010	00000	00001	000	1
61	12EH; (012EH)	ALR(AC); 0001101	11	00	0010011	11111	00001	000	1
62	13EH; (013EH)	ALR(AC); 0001101	11	00	0010100	11111	00001	000	1
63	14EH; (014EH)	ALR(T); 0001100	11	00	0010101	11111	00001	000	1
64	15EH; (015EH)	ALR(AC); 0001101	11	00	0010110	11111	00001	000	1
65	16EH; (016EH)	ALR(AC); 0001101	11	00	0010111	11111	00001	000	1
66	17EH;	ALR(AC);							

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CRUC 210	PAUSE 0
(017EH)	0001101	11	00	0011000	11111	00001	000	1
67 18EH: (018EH)	0111100	ADR(T): 11	00	0011001	11111	00001	000	1
68 19EH: (019EH)	0001101	ALR(AC): 11	00	0011010	11111	00001	000	1
69 1AEH: (01AEH)	0001100	ALR(T): 11	00	0011011	11111	00001	000	1
70 18EH: (018EH)	0000010	ILR(J): 11	00	0011100	00000	00001	000	1
71 1CEH: (01CEH)	0001100	ALR(T): 11	00	0011101	11111	00001	000	1
72 1DEH: (01DEH)	0101100	SDR(T): 11	00	0111010	11111	00001	000	1
73 1DAH: (01DAH)	0011100	LOADD: LMI(T) FF1 RRM: 11	11	0111101	00000	00001	110	1
74 1DDH: (01DDH)	0101111	LDI(AC) FF1: 11	11	0011111	11111	00001	000	1
75 1FDH: (01FDH)	0000110	ALR(R6): 11	00	0111110	11111	00001	000	1
76 1FEH: (01FEH)	0001000	ILR(M) FF1: 11	11	0111111	00000	00001	000	1
77 1FFH: (01FFH)	0000101	ILR(L): 11	00	0011110	00000	00001	000	1
78 1EFH: (01EFH)	0100111	SDR(R7) FF1: 11	11	0111110	11111	00001	000	1
79 1EEH: (01EEH)	1110111	CMR(R7): 11	00	0111101	00000	00001	000	1

RECORD NUMBER	CPE 6543210	F1 10	F0 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CHUC 210	PAUSE 0	
80	1EDH! (01EDH)	0000010	11	11	0111100	00000	00001	000	1
				ILR(J) FF1:					
81	1ECH! (01ECH)	0000111	11	11	0011101	11111	00001	000	1
				ALR(R7) FF1:					
82	1DCH! (01DCH)	1101101	11	00	1001101	00000	00001	000	1
				NOP(AC) JFL(LOADD,JGT):					
83	1DRH! (01DRH)	0000000	11	00	0111001	00000	00001	000	1
				JGT: ILR(A):					
84	1D9H! (01D9H)	0100010	11	11	0010111	11111	00001	000	1
				SDR(J) FF1:					
85	179H! (0179H)	0000100	11	00	0010110	00000	00001	000	1
				ILR(K):					
86	169H! (0169H)	0100111	11	11	0010101	11111	00001	000	1
				SDR(R7) FF1:					
87	159H! (0159H)	1110111	11	00	0010100	00000	00001	000	1
				CMR(R7):					
88	149H! (0149H)	0000001	11	11	0010011	00000	00001	000	1
				ILR(I) FF1:					
89	139H! (0139H)	0000111	11	11	0010010	11111	00001	000	1
				ALR(R7) FF1:					
90	129H! (0129H)	1101101	11	00	1000001	00000	00001	000	1
				NOP(AC) JFL(ADDR,RTNN):					
91	119H! (0119H)	00 1001	11	11	0111000	00000	00001	000	1
				RTNN: LMI(R9) FF1:					
92	119H! (0119H)	0001000	11	00	0010010	00000	00001	111	1
				ILR(M) RWM:					
93	128H! (0128H)	0000110	11	00	0010011	00000	00001	000	1
				ILR(R6):					

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CHUC 210	PAUSE 0	
94	134H! (0134H)	SDR(A) FF1 JZR(FETCH) PAGE1: 0100000	11	11	0101111	11111	00000	000	1
95	12H! (0012H)	SAVEIJ! LMI(R9) FF1: 0011001	11	11	0000011	00000	00001	000	1
96	32H! (0032H)	ILR(I) RWM: 0000001	11	00	0110001	00000	00001	111	1
97	31H! (0031H)	LMI(R9): 0011001	11	00	0000100	00000	00001	000	1
98	41H! (0041H)	ILR(J) RWM JZR(FETCH) PAGE1: 0000010	11	00	0101111	00000	00000	111	1
99	13H! (0013H)	RFTIJ! LMI(R9) FF1 RRM: 0011001	11	11	0000101	00000	00001	110	1
100	53H! (0053H)	ACM(AC): 0001011	11	00	0110001	00000	00001	000	1
101	51H! (0051H)	SDR(I) FF1: 0100001	11	11	0000110	11111	00001	000	1
102	61H! (0061H)	LMI(R9) RRM: 0011001	11	00	0000111	00000	00001	110	1
103	71H! (0071H)	ACM(AC): 0001011	11	00	0001000	00000	00001	000	1
104	81H! (0081H)	SDR(J) FF1 JZR(FETCH) PAGE1: 0100010	11	11	0101111	11111	00000	000	1
105	/*	CALCULATE BOUNDARY AREA OF EDGE GATES						*/	
106	14H! (0014H)	EGATES! LMI(P) FF1 RRM: 0010011	11	11	0000010	00000	00001	110	1
107	24H! (0024H)	ACM(AC): 0001011	11	00	0000011	00000	00001	000	1
108	34H!	SDR(R6) FF1:							

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUS 210	PAUSE 0	
	(0074H)	0100110	11	11	0000100	11111	00001	000	1
109	44H! (0044H)	0011001	LMI(R9) FF1: 11	11	0000101	00000	00001	000	1
110	54H! (0054H)	0000001	ILR(I): 11	00	0000110	00000	00001	000	1
111	64H! (0064H)	0011111	CIA(AC) FF1: 11	11	0000111	00000	00001	000	1
112	74H! (0074H)	0100111	SDR(R7) FF1: 11	11	0001000	11111	00001	000	1
113	84H! (0084H)	0000100	ILR(K): 11	00	0110101	00000	00001	000	1
114	85H! (0085H)	0000111	ALR(R7) RWM: 11	00	0110110	11111	00001	111	1
115	86H! (0086H)	0010110	LMI(R6) FF1: 11	11	0110111	00000	00001	000	1
116	87H! (0087H)	0001111	SRA(AC): 11	00	0001001	00000	00001	000	1
117	97H! (0097H)	0001111	SRA(AC): 11	00	0110110	00000	00001	000	1
118	96H! (0096H)	1011111	TZA(AC): 11	00	0110101	11111	00001	000	1
119	95H! (0095H)	1101101	NOP(AC) JFL(MINX,NOMINX): 11	00	1001010	00000	00001	000	1
120	0A2H! (00A2H)	0111111	MINX! INA(AC) FF1: 11	11	0110011	00000	00001	000	1
121	0A3H! (00A3H)	0100000	NOMINX! SDR(A) FF1 RWM: 11	11	0110100	11111	00001	111	1

RECORD NUMBER	CPF 6543210	F1 10	F0 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
122	0A4H! (00A4H)	0011001	LMI(R9) FF1: 11 11	0110101	00000	00001	000	1
123	0A5H! (00A5H)	0000010	ILR(J): 11 00	0110110	00000	00001	000	1
124	0A6H! (00A6H)	0011111	CIA(AC) FF1: 11 11	0001011	00000	00001	000	1
125	0R6H! (00R6H)	0100111	SDR(R7) FF1: 11 11	0001100	11111	00001	000	1
126	0C6H! (00C6H)	0000101	ILR(L): 11 00	0001101	00000	00001	000	1
127	0D6H! (00D6H)	0000111	ALR(R7) RWM: 11 00	0001110	11111	00001	111	1
128	0E6H! (00E6H)	0010110	LMI(R6) FF1: 11 11	0001111	00000	00001	000	1
129	0F6H! (00F6H)	0001111	SRA(AC): 11 00	0010000	00000	00001	000	1
130	106H! (0106H)	0001111	SRA(AC): 11 00	0110101	00000	00001	000	1
131	105H! (0105H)	1011111	TZA(AC): 11 00	0010001	11111	00001	000	1
132	115H! (0115H)	1101101	NOP(AC) JFL(MINY,NOMINY): 11 00	1000010	00000	00001	000	1
133	122H! (0122H)	0111111	MINY! INA(AC) FF1: 11 11	0110011	00000	00001	000	1
134	123H! (0123H)	0100111	NOMINY! SDR(R7) FF1 RWM: 11 11	0110100	11111	00001	111	1
135	124H! (0124H)	0001001	ILR(R9) FF1: 11 11	0110111	00000	00001	000	1

RECORD NUMBER	CPF 6543210	F1 10	F0 10	JUMP 6543210	KRISS 43210	OTHER 43210	CBUS 210	PAUSE 0
136	127H! (0127H)	ILR(R9) FF1: 0001001	11 11	0010110	00000	00001	000	1
137	167H! (0167H)	ILR(A): 0000000	11 00	0011010	00000	00001	000	1
138	1A7H! (01A7H)	CIA(AC) FF1: 0011111	11 11	0011011	00000	00001	000	1
139	187H! (0187H)	SDR(R8) FF1: 0101000	11 11	0011100	11111	00001	000	1
140	1C7H! (01C7H)	ILR(I): 0000001	11 00	0011101	00000	00001	000	1
141	1D7H! (01D7H)	ALR(A): 0000000	11 00	0011110	11111	00001	000	1
142	1E7H! (01E7H)	LMI(R9) FF1 RWM: 0011001	11 11	0011111	00000	00001	111	1
143	1F7H! (01F7H)	ILR(R9) FF1: 0001001	11 11	0110110	00000	00001	000	1
144	1F6H! (01F6H)	LMI(R9) FF1: 0011001	11 11	0011110	00000	00001	000	1
145	1E6H! (01E6H)	ILR(K): 0000100	11 00	0011101	00000	00001	000	1
146	1D6H! (01D6H)	ALR(R8): 0001000	11 00	0011100	11111	00001	000	1
147	1C6H! (01C6H)	CIA(AC) FF1 RWM: 0011111	11 11	0011011	00000	00001	111	1
148	1B6H! (01B6H)	ILR(R7): 0000111	11 00	0011000	00000	00001	000	1
149	1A6H! (01A6H)	CIA(AC) FF1: 0011111	11 11	0010101	00000	00001	000	1

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUC 210	PAUSE 0
150	156H! (0156H)	0101000	11 11	SDR(R8) FF1: 0010001	11111	00001	000	1
151	116H! (0116H)	0001001	11 11	ILR(R9) FF1: 0110111	00000	00001	000	1
152	117H! (0117H)	0000010	11 00	ILR(J): 0010000	00000	00001	000	1
153	107H! (0107H)	0000111	11 00	ALR(R7): 0110011	11111	00001	000	1
154	103H! (0103H)	0011001	11 11	LMI(R9) FF1 RWM: 0110010	00000	00001	111	1
155	102H! (0102H)	0001001	11 11	ILR(R9) FF1: 0110001	00000	00001	000	1
156	101H! (0101H)	0000101	11 00	ILR(L): 0010001	00000	00001	000	1
157	111H! (0111H)	0001000	11 00	ALR(R8): 0110010	11111	00001	000	1
158	112H! (0112H)	0011111	11 11	CIA(AC) FF1: 0110011	00000	00001	000	1
159	113H! (0113H)	0011001	11 11	LMI(R9) FF1 RWM: 0111001	00000	00001	111	1
160	/* IF MEAN VLAUE IS LESS THAN SEVEN TRACK BLACK ON WHITE							
161	IF MEAN VALUE IS GREATER OR EQUAL TO SEVEN TRACK WHITE ON BLACK*/							
162	119H! (0119H)	1001001	11 00	CLR(R9): 0010000	00000	00001	000	1
163	109H! (0109H)	0011001	11 11	LMI(R9) K0000F FF1: 0111000	01111	00001	000	1
164	108H! (0108H)	0011001	11 00	LMI(R9) RRM: 0010100	00000	00001	110	1

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBIUS 210	PAUSE 0
165	14BH! (014BH)	0001011	01	11	0111100	00000	00001	000
				ACM(AC) FF1 STZ;				1
166	14CH! (014CH)	1101101	11	00	1000010	00000	00001	000
				NOP(AC) JFL(WHITE+BLACK);				1
167	12AH! (012AH)	1001001	11	00	0010000	00000	00001	000
				WHITE! CLR(R9);				1
168	10AH! (010AH)	0011001	11	00	0111011	00101	00001	110
				LMI(R9) K00005 RRM;				1
169	10RH! (010RH)	1111011	11	00	0111110	00000	00001	000
				LCM(AC);				1
170	10EH! (010EH)	0111111	11	11	0101111	00000	00000	111
				INA(AC) FF1 RWM PAGE1 JZR(FETCH);				1
171	12BH! (012BH)	1101101	11	00	0101111	00000	00000	000
				-LACK! NOP(AC) PAGE1 JZR(FETCH);				1
172	/*							
173								
174	15H! (0015H)	0011001	11	00	0000010	00000	00001	110
				NGATEA NGATEB NGATEC AND NGATEL CALCULATE EDGES OF THE GATE AND THE NEW I J K L NGATEF! LMI(R9) RRM;				1
175	25H! (0025H)	0001011	11	00	0000011	00000	00001	000
				ACM(AC);				1
176	35H! (0035H)	0011111	11	11	0000100	00000	00001	000
				CIA(AC) FF1;				1
177	45H! (0045H)	0100110	11	11	0000101	11111	00001	000
				SDR(R6) FF1;				1
178	55H! (0055H)	1000111	11	00	0000110	00000	00001	000
				CLR(R7);				1
179	65H! (0065H)	0010111	11	00	0000111	00111	00001	000
				LMI(R7) K00007;				1

RECORD NUMBER	CPF 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
180	75H! (0075H)	0010111	LMI(R7) 11 11	FF1 RRM: 0001011	00000	00001	110	1
181	085H! (0085H)	0001011	ACM(AC): 11 00	0111100	00000	00001	000	1
182	08CH! (008CH)	0010111	LMI(R7) 11 11	FF1: 0001101	00000	00001	000	1
183	0DCH! (00DCH)	0010111	LMI(R7) 11 00	RRM: 0111101	00000	00001	110	1
184	0DDH! (00DDH)	0001011	AMA(AC): 11 00	0001100	11111	00001	000	1
185	0CDH! (00CDH)	0110110	ADR(R6): 11 00	0001011	11111	00001	000	1
186	0BDH! (00BDH)	1000111	CLR(R7): 11 00	0001010	00000	00001	000	1
187	0ADH! (00ADH)	0010111	LMI(R7) 11 00	K00003: 0001001	00011	00001	000	1
188	9DH! (009DH)	0010111	LMI(R7) 11 00	RRM: 0001000	00000	00001	110	1
189	8DH! (008DH)	0001011	ACM(AC): 11 00	0111100	00000	00001	000	1
190	8CH! (008CH)	0001101	ALR(AC): 11 00	0001001	11111	00001	000	1
191	9CH! (009CH)	0001101	ALR(AC): 11 00	0001010	11111	00001	000	1
192	0ACH! (00ACH)	0011001	LMI(R9) 11 11	FF1 RRM: 0001100	00000	00001	111	1
193	0CCH! (00CCH)	0000110	ALR(R6): 11 00	0111110	11111	00001	000	1

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
194	0CEH! (00CEH)	0100000	11 11	0101111	11111	00000	000	1
SDR(A) FF1 JZR(FETCH) PAGE1:								
195	16H! (0016H)	0011001	11 00	0000010	00000	00001	110	1
NGATEB: LMI(R9) RRM:								
196	26H! (0026H)	0001011	11 00	0000011	00000	00001	000	1
ACM(AC):								
197	36H! (0036H)	0001111	11 00	0000100	00000	00001	000	1
SRA(AC):								
198	46H! (0046H)	0000000	11 00	0000101	11111	00001	000	1
ALR(A):								
199	56H! (0056H)	0100000	11 11	0000110	11111	00001	000	1
SDR(A) FF1:								
200	66H! (0066H)	0011111	11 11	0000111	00000	00001	000	1
CIA(AC) FF1:								
201	76H! (0076H)	0010011	11 11	0111001	00000	00001	110	1
LMI(P) FF1 RRM:								
202	79H! (0079H)	0011010	11 00	0001000	00000	00001	110	1
LMM(T) RRM:								
203	89H! (0089H)	0001011	11 00	0001001	11111	00001	000	1
AMA(AC):								
204	99H! (0099H)	0100001	11 11	0001010	11111	00001	000	1
SDR(I) FF1:								
205	0A9H! (00A9H)	0000000	11 00	0001011	00000	00001	110	1
ILR(A) RRM:								
206	0B9H! (00B9H)	0001011	11 00	0001100	11111	00001	000	1
AMA(AC):								
207	0C9H! (00C9H)	0100100	11 11	0101111	11111	00000	000	1
SDR(K) FF1 JZR(FETCH) PAGE1:								

RECORD NUMBER		CPE 4543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
208	17H! (0017H)	NGATEC!	LMI(R9)	RRM:					
		0011001	11	00	0000010	00000	00001	110	1
209	27H! (0027H)		ACM(AC):						
		0001011	11	00	0000011	00000	00001	000	1
210	37H! (0037H)		CIA(AC)	FF1:					
		0011111	11	11	0000100	00000	00001	000	1
211	47H! (0047H)		SDR(R6)	FF1:					
		0100110	11	11	0000101	11111	00001	000	1
212	57H! (0057H)		CLR(R7):						
		1000111	11	00	0000110	00000	00001	000	1
213	67H! (0067H)		LMI(R7)	K00000:					
		0010111	11	00	0000111	01011	00001	000	1
214	77H! (0077H)		LMI(R7)	FF1 RRM:					
		0010111	11	11	0001010	00000	00001	110	1
215	0A7H! (00A7H)		ACM(AC):						
		0001011	11	00	0001011	00000	00001	000	1
216	0B7H! (00B7H)		LMI(R7)	FF1:					
		0010111	11	11	0001100	00000	00001	000	1
217	0C7H! (00C7H)		LMI(R7)	RRM:					
		0010111	11	00	0001101	00000	00001	110	1
218	0D7H! (00D7H)		AMA(AC):						
		0001011	11	00	0001110	11111	00001	000	1
219	0E7H! (00E7H)		ADR(R6):						
		0110110	11	00	0001111	11111	00001	000	1
220	0F7H! (00F7H)		CLR(R7):						
		1000111	11	00	0111000	00000	00001	000	1
221	0F8H! (00F8H)		LMI(R7)	K00002:					
		0010111	11	00	0111001	00010	00001	000	1

RECORD NUMBER		CPE 4543210	FI 10	FD 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
222	0F9H! (00F9H)	0010111	11	00	0111100	00000	00001	110	1
					LMI(R7) RRM:				
223	0FCH! (00FCH)	0001011	11	00	0111101	00000	00001	000	1
					ACM(AC):				
224	0FDH! (00FDH)	0001101	11	00	0001110	11111	00001	000	1
					ALR(AC):				
225	0EDH! (00EDH)	0001101	11	00	0111100	11111	00001	000	1
					ALR(AC):				
226	0ECH! (00ECH)	0011001	11	11	0111001	00000	00001	111	1
					LMI(P9) FF1 RWM:				
227	0E9H! (00E9H)	0000110	11	00	0111000	11111	00001	000	1
					ALR(R6):				
228	0E8H! (00E8H)	0100000	11	11	0101111	11111	00000	000	1
					SDR(A) FF1 JZR(FETCH) PAGE1:				
229	18H! (0018H)	0011001	11	00	0000010	00000	00001	110	1
					NGATEL: LMI(R9) RRM:				
230	29H! (0029H)	0001011	11	00	0000011	00000	00001	000	1
					ACM(AC):				
231	38H! (0038H)	0001111	11	00	0000100	00000	00001	000	1
					SRA(AC):				
232	48H! (0048H)	0000000	11	00	0000101	11111	00001	000	1
					ALR(A):				
233	58H! (0058H)	0100000	11	11	0000110	11111	00001	000	1
					SDR(A) FF1:				
234	68H! (0068H)	0011111	11	11	0000111	00000	00001	000	1
					CIA(AC) FF1:				
235	78H! (0078H)	0010011	11	11	0001000	00000	00001	110	1
					LMI(P) FF1 RRM:				

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
236	88H! (0088H)	0011010	11 00	LMM(T) RRM: 0001001	00000	00001	110	1
237	98H! (0098H)	0001011	11 00	AMA(AC): 0001010	11111	00001	000	1
238	0A8H! (00A8H)	0100010	11 11	SDR(J) FF1: 0001011	11111	00001	000	1
239	0B8H! (00B8H)	0000000	11 00	ILR(A) RRM: 0001100	00000	00001	110	1
240	0C8H! (00C8H)	0001011	11 00	AMA(AC): 0001101	11111	00001	000	1
241	0D8H! (00D8H)	0100101	11 11	SDR(L) FF1 JZR(FETCH) PAGE1: 0101111	11111	00000	000	1
242	/* WINDOW----- (VARIABLE WINDOW MICROROUTINE) */							
243	19H! (0019H)	0011001	11 11	WINDOW! LMI(R9) FF1: 0000010	00000	00001	000	1
244	29H! (0029H)	0000010	11 00	ILR(J) ROT: 0000011	00000	00001	101	1
245	39H! (0039H)	0011001	11 11	LMI(R9) FF1: 0000100	00000	00001	000	1
246	49H! (0049H)	0000101	11 00	ILR(L) ROT: 0000101	00000	00001	101	1
247	59H! (0059H)	0011001	11 11	LMI(R9) FF1: 0000110	00000	00001	000	1
248	69H! (0069H)	0000001	11 00	ILR(I) ROT: 0111100	00000	00001	101	1
249	6CH! (006CH)	0011001	11 11	LMI(R9) FF1: 0111101	00000	00001	000	1
250	6DH!			ILR(K) ROT JZR(FETCH) PAGE1:				

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
(006DH)	0000100	11	00	0101111	00000	00000	101	1
251	1AH! (001AH)	FRAME! 11 1000	NOP(R8) 11 00	STROBE2: 0011010	00000	10001	000	1
252	1AAH! (01AAH)	WAT! 1101000	NOP(R8) 11 00	RIN: 0111100	00000	00001	100	1
253	1ACH! (01ACH)		LDI(AC) 11 11	FF1: 0111101	11111	00001	000	1
254	1ADH! (01ADH)		TZR(AC) 11 00	K80000: 0111111	10010	00001	000	1
255	1AFH! (01AFH)		NOP(R8) 11 00	JFL(WAT,RDY): 1001010	00000	00001	000	1
256	1ABH! (01ABH)	RDY! 1101000	NOP(R8) 11 00	JZR(FETCH) PAGE1: 0101111	00000	00000	000	1
257	1BH! (001BH)	START! 001001	LMI(R9) 11 11	FF1 RRM: 0000011	00000	00001	110	1
258	3BH! (003BH)		ACM(AC): 11 00	0111101	00000	00001	000	1
259	3DH! (003DH)		SDR(I) 11 11	FF1: 0010000	11111	00001	000	1
260	10DH! (010DH)		LMI(R9) 11 11	FF1 RRM: 0010010	00000	00001	110	1
261	12DH! (012DH)		ACM(AC): 11 00	0010011	00000	00001	000	1
262	13DH! (013DH)		SDR(J) 11 11	FF1: 0010100	11111	00001	000	1
263	14DH! (014DH)		LMI(R9) 11 11	FF1 RRM: 0010101	00000	00001	110	1

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
264	15DH! (015DH)	0001011	11 00	0010110	00000	00001	000	1
		ACM(AC):						
265	16DH! (016DH)	0100100	11 11	0010111	11111	00001	000	1
		SDR(K) FF1:						
266	17DH! (017DH)	0011001	11 00	0011000	00000	00001	110	1
		LMI(R9) RRM:						
267	18DH! (018DH)	0001011	11 00	0011001	00000	00001	000	1
		ACM(AC):						
268	19DH! (019DH)	0100101	11 11	0101111	11111	00000	000	1
		SDR(L) FF1 JZR(FETCH) PAGE1:						
269	1CH! (001CH)	1100000	11 00	0101111	00000	00000	000	1
		NOTC! NOP(A) JZR(FETCH) PAGE1:						
270	1DH! (001DH)	1100000	11 00	0101111	00000	00000	000	1
		NOTD! NOP(A) JZR(FETCH) PAGE1:						
271	1EH! (001EH)	1100000	11 00	0101111	00000	00000	000	1
		NOTE! NOP(A) JZR(FETCH) PAGE1:						
272	1FH! (001FH)	1100000	11 00	0101111	00000	00000	000	1
		NOTF! NOP(A) JZR(FETCH) PAGE1:						
273	/* CENTROID CALCULATIONS AND TARGET AREAS OF EDGE GATES							
274	10H! (0010H)	1001011	11 00	0000000	00000	00001	000	1
		CENTRD! CLA(AC):						
275	00H! (0000H)	0101000	11 11	0000010	11111	00001	000	1
		SDR(M) FF1:						
276	20H! (0020H)	0011001	11 11	0000011	00000	00001	111	1
		LMI(R9) FF1 RWM:						
277	30H! (0030H)	0011001	11 11	0000100	00000	00001	000	1
		LMI(R9) FF1:						
278	40H!							
		LMI(R9) FF1 RWM:						

*/

RECORD NUMBER	CPF	F1	F0	JUMP	KRUSS	OTHER	CBUS	PAUSE
	6543210	10	10	6543210	43210	43210	210	0
(0040H)	0011001	11	11	0000101	00000	00001	111	1
279	50H! (0050H)	LMI(R9)	FF1:					
	0011001	11	11	0000110	00000	00001	000	1
280	60H! (0060H)	LMI(R9)	FF1 RWM:					
	0011001	11	11	0000111	00000	00001	111	1
281	70H! (0070H)	LMI(R9)	FF1:					
	0011001	11	11	0001000	00000	00001	000	1
282	80H! (0080H)	LMI(R9)	FF1 RWM:					
	0011001	11	11	0001001	00000	00001	111	1
283	90H! (0090H)	LMI(R9)	FF1 RWM:					
	0011001	11	11	0001010	00000	00001	111	1
284	0A0H! (00A0H)	LMI(R9)	RWM:					
	0011001	11	00	0001011	00000	00001	111	1
285	0B0H! (00B0H)	ILR(J):						
	0000010	11	00	0001100	00000	00001	000	1
286	0C0H! (00C0H)	SDR(A)	FF1:					
	0100000	11	11	0110010	11111	00001	000	1
287	0C2H! (00C2H)	CADDR:	ILR(I):					
	0000001	11	00	0110100	00000	00001	000	1
288	0C4H! (00C4H)	DCA(AC):						
	0011111	11	00	0001101	11111	00001	000	1
289	0D4H! (00D4H)	CLA(T):						
	1001010	11	00	0110000	00000	00001	000	1
290	0D0H! (00D0H)	ALR(AC):						
	0001101	11	00	0001110	11111	00001	000	1
291	0E0H! (00E0H)	ALR(AC):						
	0001101	11	00	0001111	11111	00001	000	1

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBIUS 210	PAUSE 0
292	0F0H! (00F0H)	ALR(T)!	0001100	11	00	0010000	11111	00001 000 1
293	100H! (0100H)	ALR(AC)!	0001101	11	00	0010001	11111	00001 000 1
294	110H! (0110H)	ALR(AC)!	0001101	11	00	0010010	11111	00001 000 1
295	120H! (0120H)	ALR(AC)!	0001101	11	00	0010011	11111	00001 000 1
296	130H! (0130H)	ADR(T)!	0111100	11	00	0010100	11111	00001 000 1
297	140H! (0140H)	ALR(AC)!	0001101	11	00	0010101	11111	00001 000 1
298	150H! (0150H)	ALR(T)!	0001100	11	00	0111000	11111	00001 000 1
299	158H! (0158H)	ILR(J)!	0000010	11	00	0010110	00000	00001 000 1
300	168H! (0168H)	ALR(T)!	0001100	11	00	0110000	11111	00001 000 1
301	160H! (0160H)	SDR(T)!	0101100	11	00	0110010	11111	00001 000 1
302	162H! (0162H)	CLOAD! CLR(R9)!	1001001	11	00	0110110	00000	00001 000 1
303	166H! (0166H)	TZR(R9) FFZ!	1011001	11	10	0111100	11111	00001 000 1
304	/*	BRANCH TO WHITEE FOR WHITE ON BLACK TARGET						
305	/*	BRANCH TO BLACKK TO TRACK ON BLACK TARGET						
306	16CH! (016CH)	LMI(R9) K00005 FF1 RRM JFL(WHITEE, BLACKK)!	0011001	11	11	1000111	00101	00001 110 1

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
307	17BH: (017BH)	BLACKK1	ACM(AC):	0001011	11 00 0111100	00000	00001	000 1
308	17CH: (017CH)	0100110	SDR(R6) FF1:	11 11 0111000	11111	00001	000	1
309	17AH: (017AH)	0011100	LMI(T) FF1 RRM:	11 11 0011000	00000	00001	110	1
310	18BH: (018BH)	0101111	LDI(AC) FF1:	11 11 0011001	11111	00001	000	1
311	19BH: (019BH)	0011111	CIA(AC) FF1:	11 11 0110111	00000	00001	000	1
312	197H: (0197H)	0000110	ALR(R6) JCR(OVER):	11 00 0110000	11111	00001	000	1
313	17AH: (017AH)	0011011	WHITEE1 ACM(AC):	11 00 0110000	00000	00001	000	1
314	170H: (0170H)	0011100	LMI(T) FF1 RRM:	11 11 0011000	00000	00001	110	1
315	180H: (0180H)	0111111	AIA(AC):	11 00 0011001	11111	00001	000	1
316	190H: (0190H)	1101101	OVER1 NOP(AC) JFL(NODATA,DATA):	11 00 1001001	00000	00001	000	1
317	192H: (0192H)	0000101	NODATA1 ILR(L):	11 00 0110001	00000	00001	000	1
318	191H: (0191H)	0100110	SDR(R6) FF1:	11 11 0011010	11111	00001	000	1
319	1A1H: (01A1H)	1110110	CMR(R6):	11 00 0110100	00000	00001	000	1
320	1A4H: (01A4H)	0000010	ILR(J) FF1:	11 11 0011001	00000	00001	000	1

RECORD NUMBER	CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
321	194H! (0194H)	0000110	11 11	0010101	11111	00001	000	1
			ALR(R6) FF1:					
322	154H! (0154H)	1101101	11 00	1000110	00000	00001	000	1
			NOP(AC) JFL(CLOAD,JGTL):					
323	163H! (0163H)	0000000	11 00	0110101	00000	00001	000	1
			JGTL! ILR(A):					
324	165H! (0165H)	0100010	11 11	0010100	11111	00001	000	1
			SDR(J) FF1:					
325	145H! (0145H)	0000100	11 00	0110100	00000	00001	000	1
			ILR(K):					
326	144H! (0144H)	0100110	11 11	0010000	11111	00001	000	1
			SDR(R6) FF1:					
327	104H! (0104H)	1110110	11 00	0001111	00000	00001	000	1
			CMR(R6):					
328	0F4H! (00F4H)	0000001	11 11	0001110	00000	00001	000	1
			ILR(I) FF1:					
329	0E4H! (00E4H)	0000110	11 11	0001011	11111	00001	000	1
			ALR(R6) FF1:					
330	0B4H! (00B4H)	1101101	11 00	1001100	00000	00001	000	1
			NOP(AC) JFL(CADDR,ENDC):					
331	193H! (0193H)	0001001	11 00	0110101	00000	00001	000	1
			DATA! ILR(R9):					
332	195H! (0195H)	0100110	11 11	0011010	11111	00001	000	1
			SDR(R6) FF1:					
333	1A5H! (01A5H)	0010110	11 00	0110000	01000	00001	000	1
			LMI(R6) K00008:					
334	1A0H! (01A0H)	0000001	11 00	0011011	00000	00001	000	1
			ILR(I):					

RECORD NUMBER	CPF 6543210	FI 10	FO 10	JUMP 6543210	KRUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
335	180H! (0180H)	0010110	LMI(R6) FF1 RRM: 11 11	0011100	00000	00001	110	1
336	1C0H! (01C0H)	0001011	AMA(AC) RWM: 11 00	0011101	11111	00001	111	1
337	1D0H! (01D0H)	0000010	ILR(J): 11 00	0011110	00000	00001	000	1
338	1E0H! (01E0H)	0010110	LMI(R6) RRM: 11 00	0011111	00000	00001	110	1
339	1F0H! (01F0H)	0001011	AMA(AC) RWM: 11 00	0110101	11111	00001	111	1
340	1F5H! (01F5H)	0001000	ILR(M) FF1: 11 11	0011110	00000	00001	000	1
341	1E5H! (01E5H)	0000001	ILR(I): 11 00	0011101	00000	00001	000	1
342	1D5H! (01D5H)	0011111	CIA(AC): 11 00	0110100	00000	00001	000	1
343	1D4H! (01D4H)	0011001	LMI(R9) FF1 RRM: 11 11	0011100	00000	00001	110	1
344	1C4H! (01C4H)	0001011	AMA(AC): 11 00	0110101	11111	00001	000	1
345	1C5H! (01C5H)	1101101	NOP(AC) JFL(NXEEG,XEEG): 11 00	1001011	00000	00001	000	1
346	1B3H! (01B3H)	0011001	XEEG! LMI(R9) FF1 RRM: 11 11	0110100	00000	00001	110	1
347	1B4H! (01B4H)	0001011	ACM(AC) FF1 RWM JCC(NXBLG2): 11 11	0010111	00000	00001	111	1
348	1B2H! (01B2H)	0001001	NXEEG! ILR(R9) FF1: 11 11	0110001	00000	00001	000	1

RECORD NUMBER		CPE 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CRUS 210	PAUSE 0	
349	181H! (0181H)	0000001	11	00	0011000	00000	00001	000	1	
			ILR(I):							
350	181H! (0181H)	0011001	11	11	0010111	00000	00001	110	1	
			LMI(R9) FF1 RRM:							
351	171H! (0171H)	0001011	11	00	0010110	11111	00001	000	1	
			AMA(AC):							
352	161H! (0161H)	1101101	11	00	1000111	00000	00001	000	1	
			NOP(AC) JFL(NXBLG,XBLG):							
353	173H! (0173H)	0011001	11	11	0110101	00000	00001	110	1	
			XRLG! LMI(R9) FF1 RRM:							
354	175H! (0175H)	0001011	11	11	0110111	00000	00001	111	1	
			ACM(AC) FF1 RWM JCR(NXBL):							
355	174H! (0174H)	0001001	11	11	0110010	00000	00001	000	1	
			NXBLG2! ILR(R9) FF1:							
356	172H! (0172H)	001001	11	11	0110111	00000	00001	000	1	
			NXBLG! ILR(R9) FF1:							
357	177H! (0177H)	0000010	11	00	0010011	00000	00001	000	1	
			NXBL! ILR(J):							
358	137H! (0137H)	0011111	11	00	0110110	00000	00001	000	1	
			CIA(AC):							
359	136H! (0136H)	0011001	11	11	0110101	00000	00001	110	1	
			LMI(R9) FF1 RRM:							
360	135H! (0135H)	0001011	11	00	0110100	11111	00001	000	1	
			AMA(AC):							
361	134H! (0134H)	1101101	11	00	1000100	00000	00001	000	1	
			NOP(AC) JFL(NYEEG,YEEG):							
362	143H! (0143H)	0011001	11	11	0110110	00000	00001	110	1	
			YFEG! LMI(R9) FF1 RRM:							

RECORD NUMBER	CPF 6543210	FI 10	FO 10	JUMP 6543210	KBUSS 43210	OTHER 43210	CBUS 210	PAUSE 0
363	146H! (0146H)	0001011	11 11	ACM(AC) FF1 RWM; 0011001	00000	00001	111	1
364	196H! (0196H)	1100000	11 00	NOP(A) JCR(NODATA); 0110010	00000	00001	000	1
365	142H! (0142H)	00 1001	11 11	NYEEG! ILR(R9) FF1; 0110111	00000	00001	000	1
366	147H! (0147H)	0000010	11 00	ILR(J); 0010101	00000	00001	000	1
367	157H! (0157H)	0011001	11 11	LMI(R9) FF1 RRM; 0011000	00000	00001	110	1
368	187H! (0187H)	0001011	11 00	AMA(AC); 0110100	11111	00001	000	1
369	184H! (0184H)	1101101	11 00	NOP(AC) JFL(NODAT.YBLG); 1001000	00000	00001	000	1
370	183H! (0183H)	0011001	11 11	YRLG! LMI(R9) FF1 RRM; 0110101	00000	00001	110	1
371	185H! (0185H)	0001011	11 11	ACM(AC) FF1 RWM JCR(NODAT); 0110010	00000	00001	111	1
372	0C3H! (00C3H)	1001001	11 00	ENDC! CLR(R9); 0110101	00000	00001	000	1
373	0C5H! (00C5H)	0011001	11 00	LMI(R9) K00004; 0001110	00100	00001	000	1
374	0E5H! (00E5H)	0001000	11 00	ILR(M) RWM PAGE1 JZR(FETCH); 0101111	00000	00000	111	1
375	182H! (0182H)	1100000	11 00	NODAT! NOP(A) JCC(NODATA); 0011001	00000	00001	000	1
376	EOF							

Appendix D.
EAI PACER 100 EMULATION MICRO CODE LISTING

XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 3000

RECORD
NUMBER

```

1
2 KRUSS FIELD LENGTH=5 DEFAULT=0
3 MICROPS(KFFFFF=10101R KFFFFA=10101B K7FFFF=10011B
4 K80000=10010R K01FFF=10001R K00040=10000R
5 K0000F=01111R K0000E=01110R K0000D=01101R
6 K0000C=01100R K0000B=01011B K0000A=01010R
7 K00009=01001R K00008=01000R K00007=00111R
8 K00006=00110R K00005=00101R K00004=00100R
9 K00003=00011R K00002=00010R K00001=00001R
10 K00000=00000R);
11 KRUSS KRUSS;
12
13 OTHER FIELD LENGTH=5 DEFAULT=0
14 MICROPS(SFT1=11111R PAGE1=00000R PAGE2=00001R);
15
16 /* DEFINITION OF BUS CONTROL FIELD */
17
18 CBUS FIELD LENGTH=3 DEFAULT=0
19 MICROPS(NRO=000B INH=001R PMW=010R CNB=011B
20 RIN=100R ROT=101R RRM=110R RWM=111B);
21
22 /* NRO NOBUS OPERATION
23 INH INHIBIT CPE ARRAY
24 RMW READ-MODIFY-WRITE
25 CNB CPU NEEDS BUS
26 RIN REQUEST INPUT
27 ROT REQUEST OUTPUT
28 RRM REQUEST READ MEMORY
29 RWM REQUEST WRITE MEMORY
30
31 SFT UP SYMBOLIC REPRESENTATION OF REGISTER DESIGNATIONS */
32
33 A STRING 'R0';
34 X STRING 'R1';
35 Q STRING 'R2';
36 P STRING 'R3';
37 S STRING 'R4';
38 B STRING 'R5';
39 E STRING 'R6';
40 W STRING 'R7';
41
42 PAUSE FIELD LENGTH=1 DEFAULT=1
43 MICROPS(SP1=1 SP0=0);
44
45 00H! INIT! CLR(A);
46 10H! CLR(X);
47 20H! CLR(W) JCC(INIT1);
48 30H! DIR! LMI(AC) RRM JPR(LA STA LX STX A1 S1 M D AOM ORI XORI
49 ANDD C EE FF GG);
50 40H! INIT1! CLR(T) JCC(INIT2);
51 60H! SE! NOP(A) JCF(NTEQL,EQL);
52 70H! LAI ACM(AC);
53 80H! SDR(A) FF1 JZR(FETCH);
54 90H! INIT2! LMI(T);

```


XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 7000

RECORD
NUMBER

55	0A0H!	ILR(W) ROT;
56	0B0H!	LMI(T) FF1 RRM;
57	0C0H!	ACM(AC);
58	0D0H!	SDR(S) FF1;
59	0E0H!	LMI(T) RRM;
60	0F0H!	ACM(AC);
61	100H!	SDR(P) FF1 JZR(FETCH);
62	110H!	CLR1! CLR(A) JZR(FETCH);
63	1E0H!	N1! NOP(A) JZR(FETCH);
64	31H!	J! SDR(P) FF1 JZP(FETCH);
65	61H!	SG! NOP(A) JCC(SGC);
66	71H!	SYA! ILR(A) RWM JZR(FETCH);
67	91H!	SGC! NOP(A) JCF(NTEQL2,EQL2);
68	111H!	AOA! ILR(A) FF1 STC JZR(FETCH);
69	131H!	YY! SDR(A) FF1 JZR(FETCH);
70	151H!	WW! SDR(A) FF1 JZR(FETCH);
71	1E1H!	N2! NOP(A) JZR(FETCH);
72	02H!	N EQ3! NOP(A) JZF(LT,NTLT);
73	12H!	LT! ILR(P) FF1 JZR(FETCH);
74	22H!	NTEQL! NOP(A) JZR(FETCH);
75	32H!	L! LMI(S) FF1;
76	42H!	ILR(P) RWM;
77	52H!	ILR(R9) JCR(CL);
78	62H!	SL! NOP(A) JCF(NTEQ3,EQ3);
79	72H!	LX! ACM(AC);
80	82H!	SDR(X) FF1 JZR(FETCH);
81	92H!	NTEQL2! NOP(A) JZF(NTGT,GT);
82	0A2H!	NTGT! NOP(A) JZR(FETCH);
83	0D2H!	NTEQL4! ILR(P) FF1 JZR(FETCH);
84	0E2H!	NTEQ5! NOP(A) JZF(NTGT5,GT5);
85	0F2H!	NTGT5! NOP(A) JZR(FETCH);
86	112H!	CA0! CLR(A);
87	122H!	ILR(A) FF1 JZR(FETCH);
88	132H!	OUT2! ALR(AC) FF2 JCR(YY);
89	142H!	OUT3! NOP(A) JZR(FETCH);
90	162H!	NTEQ6! NOP(A) JZF(NTGT6,GT6);
91	172H!	NTGT6! ILR(P) FF1 JZR(FETCH);
92	1E2H!	N3! NOP(A) JZR(FETCH);
93	03H!	E 3! NOP(A) JZR(FETCH);
94	13H!	NTLT! NOP(A) JZR(FETCH);
95	23H!	EQL! ILR(P) FF1 JZR(FETCH);
96	33H!	RFG! ILR(A) JCC(REGC);
97	63H!	SNE! NOP(A) JCC(SNEC);
98	73H!	S X! ILR(X) RWM JZR(FETCH);
99	83H!	SNEC! NOP(A) JCF(NTEQL4,EQL4);
100	93H!	EQL2! NOP(A) JZR(FETCH);
101	0A3H!	GT! ILR(P) FF1 JZR(FETCH);
102	0D3H!	EQL4! NOP(A) JZR(FETCH);
103	0E3H!	EQ5! ILR(P) FF1 JZR(FETCH);
104	0F3H!	GT5! ILR(P) FF1 JZR(FETCH);
105	103H!	REGC! NOP(A) JPR(CLR1 AOA CAO TCA ARS ALS LRS SSP SSN EX EP ES ICX DCX NOPD PZ);
106		
107	113H!	TCA! CMR(A);
108	123H!	ILR(A) FF1 STC JZR(FETCH);

XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 3000

RECORD
NUMBER

```

109 133H! AGN2! DSM(R9) JCR(XX);
110 143H! AGN3! DSM(R9) JCR(BACK);
111 163H! EQ6! ILR(P) FF1 JZP(FETCH);
112 173H! GT6! NOP(A) JZR(FETCH);
113 1E3H! N4! NOP(A) JZR(FETCH);
114 34H! I! LMI(AC) RRM:
115 44H! ACM(AC);
116 54H! LMI(AC) RRM JPR(LA STA LX STX A1 S1 M D AOM ORI XORI
117 ANDD C EE FF GG);
118 64H! SGE! NOP(A) JCC(SGEC);
119 74H! A1! ACM(AC);
120 84H! ALP(A) STC JZP(FETCH);
121 0E4H! SGE! NOP(A) JCF(NTEQ5,EQ5);
122 114H! ARS! TZP(A) STC K80000 INH;
123 124H! DSM(R9);
124 134H! XX! SRA(AC) STZ FFC JFL(OUT2,AGN2);
125 104H! DWN1! LMI(R9) FF1 JCR(PUSA4);
126 1E4H! N5! NOP(A) JZR(FETCH);
127 35H! J-! LMI(AC) RRM:
128 45H! ACM(AC);
129 55H! CLI SDR(P) FF1 JZR(FETCH);
130 65H! SLE! NOP(A) JCC(SLEC);
131 75H! SI! ACM(AC);
132 85H! CIA(AC) FF1;
133 95H! ALR(A) STC JZR(FETCH);
134 115H! ALS! DSM(R9);
135 125H! ILR(A);
136 135H! DSM(R9);
137 145H! BACK! ALP(A) STC JFL(OUT3,AGN3);
138 155H! SLEC! NOP(A) JCF(NTEQ6,EQ6);
139 105H! DWN2! ILR(R8) RWM JCR(DWN1);
140 1E5H! TRAP! NOP(A) JZR(FETCH);
141 36H! LI! LMI(S) FF1;
142 46H! ILR(P) RWM;
143 56H! LMI(R9) RRM JCR(CLI);
144 66H! PUSA! LMI(R9) FF1 JCC(PUSAC);
145 /* LOAD MULTIPLIER ---MLT---%T */
146 76H! M! ACM(T);
147 /* SAVE SIGN BIT IN C FLAG */
148 86H! ILR(T);
149 8EH! SDR(R9) FF1;
150 9EH! TZP(A) K80000 INH STC;
151 /* WIPE OUT SIGN BIT */
152 96H! TZP(T) K7FFFF;
153 /* SET UP LOOP COUNTER */
154 0A6H! CLP(AC);
155 0A5H! LMI(AC) K0000F;
156 0A4H! LMI(AC) K00004;
157 0B4H! CIA(AC) FF1;
158 0B5H! SDR(R8) FF1;
159 /* CLEAR PARTIAL PRODUCT */
160 0C5H! CLP(AC);
161 /* FETCH AND TEST MULTIPLIER LSR */
162 0C4H! SRA(T);

```

XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 3000

RECORD
NUMBER

```

163 0C2H!    MLP!    LMI(R8) FF1 STZ JFL(MRZ,MR1);
164 /*      ADD SEQUENCE  */
165 0B3H!    MR1!    SDR(R7) FF1 JCR(OV1);
166 0BFH!    OV1!    ILR(A) FF0;
167 0BEH!    ALR(R7) FF0 JCR(MR7);
168 /*      SHIFT RIGHT FILL WITH ONES  */
169 0B2H!    MRZ!    SRA(AC) FFC STZ;
170 0B1H!    SRA(T) FFZ JZF(MLP,MEX);
171 /*      APPLY CORRECTION  */
172 0C3H!    MEX!    TZR(R9) K80000 INH JCR(XXX);
173 0C9H!    XXX!    NOP(A) JFL(POS,NEG);
174 0CBH!    NEG!    CMR(A) JCR(OV2);
175 0CDH!    OV2!    ALR(A) FF1;
176 0CAH!    P S!    SDR(A) FF1 JCR(OV3);
177 0CEH!    OV3!    ILR(T);
178 0C1H!    SRA(AC) FF0;
179 0CFH!    SDR(Q) FF1 JZR(FETCH);
180 0B6H!    PUSAC!   ILR(R0) RWM;
181 0C6H!    LMI(R9) FF1;
182 0D6H!    ILR(R1) RWM;
183 0E6H!    LMI(R9) FF1;
184 0F6H!    ILR(R2) RWM;
185 106H!    LMI(R9) FF1 JCC(PUSA2);
186 116H!    LRS!    ILP(A);
187 126H!    DSM(R9) JCC(Z7);
188 136H!    PUSA2!   ILR(R3) RWM;
189 146H!    LMI(R9) FF1 JCC(PUSA3);
190 156H!    Z7!    SRA(AC) STZ JFL(OUT,AGN);
191 152H!    OUT!    ALR(AC) FFZ JCR(WW);
192 153H!    AGN!    DSM(R9) JCR(ZZ);
193 166H!    PUSA3!   ILR(R4) RWM;
194 176H!    LMI(R9) FF1;
195 186H!    ILR(R5) RWM;
196 196H!    LMI(R9) FF1;
197 1A6H!    ILR(R6) RWM;
198 1B6H!    LMI(R9) FF1;
199 1C6H!    ILR(R7) RWM;
200 026H!    LMI(R9) FF1;
201 27H!    ILR(R8) RWM;
202 28H!    LMI(R9) FF1;
203 29H!    ILR(T) RWM JZR(FETCH);
204 1E6H!    N7!    NOP(A) JZR(FETCH);
205 106H!    LMI(R9) FF1 JCR(DWN2);
206 37H!    MSC!    NOP(A) JPR(SE SG SL SNE SGE SLE PUSA PUSX POPX
RTN SKN SKP SO SNO SAE DN);
207
208 57H!    CLI!    LTM(AC) JCR(CL);
209 67H!    PUSX!    LMI(S) FF1 ICC(PUSX1);
210 /*      DIVIDE .....  */
211 77H!    D!    CLR(R6);
212 0F7H!    CLR(AC);
213 107H!    LMI(AC) K0000F;
214 127H!    LMI(AC) K00005;
215 137H!    CIA(AC) FF1;
216 147H!    SDR(R8) FF1;

```

XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 3000

RECORD
NUMBER

```

217 157H!      LMI(R9) RRM:
218 167H!      ACM(AC):
219 177H!      SDR(R9) FF1:
220 187H!      SDR(T) FF1:
221 185H!      ILP(A):
222 184H!      SDR(R2) FF1:
223 /* START MAIN LOOP, COMPARE SIGN Y WITH SIGN RI */
224 192H!      DIV!  XNR(R9):
225 181H!      TZR(R9) FF0 K80000 INH:
226 180H!      ILR(R6) JFL(SZERO, SONE):
227 /* DIFFERENT SIGNS */
228 192H!      SZERO! ALR(R6) FF0:
229 191H!      ILR(R2):
230 190H!      ALR(R2) FF0:
231 1A0H!      ILR(T):
232 180H!      SDR(R9) FF1:
233 1C0H!      ALR(R2) JCR(OVER):
234 /* SAME SIGNS */
235 193H!      SONE!  ALR(R6) FF1:
236 194H!      ILR(R2):
237 195H!      ALP(R2):
238 197H!      ILR(T):
239 1A7H!      SDR(R9):
240 187H!      CMR(AC):
241 181H!      ALR(R2) FF1:
242 /* CHECK FOR I=0 */
243 1C1H!      OVER!  LMI(R8) FF1:
244 1C7H!      NOP(A) JFL(DIV+DONE):
245 /* APPLY CORRECTION */
246 183H!      DONE!  CLR(AC):
247 188H!      LMI(AC) K80000 FF1:
248 198H!      ALR(R6):
249 1A9H!      SDR(A) FF1 JZR(FETCH):
250 87H!      PLUSX! ILR(A) RWM:
251 97H!      LMI(S) FF1:
252 0A7H!      ILR(X) RWM:
253 087H!      LMI(S) FF1:
254 0C7H!      ILR(P) RWM:
255 0D7H!      LMI(S) FF1:
256 0E7H!      ILR(W) RWM JZR(FETCH):
257 117H!      SSP!  TZR(A) K7FFFF JZR(FETCH):
258 107H!      PLUSA6! ILR(T) RWM JZR(FETCH):
259 1E7H!      R I!  NOP(A) JZR(FETCH):
260 38H!      XK!   ALR(X):
261 48H!      LMI(AC) RRM JPR(LA STA LX STX AI S1 M D AOM ORI XOR)
262                                     ANDD C EE FF GG):
263 68H!      POPX!  DSM(S) JCC(POPX):
264 78H!      AOM!  ACM(AC) FF1 RWM JZR(FETCH):
265 88H!      POPX1! LMI(S) RRM:
266 98H!      ACM(AC):
267 0A8H!      SDR(W) FF1:
268 0B8H!      DSM(S):
269 0C8H!      LMI(S) RRM:
270 0D8H!      ACM(AC):

```

XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 3000

RECORD
NUMBER

```

271 0E8H!      SDR(P)FF1;
272 0F8H!      DSM(S);
273 108H!      LMI(S) RRM JCC(POPX2);
274 118H!      SSN! CSP(AC);
275 128H!      ORR(A) K80000 JZR(FETCH);
276 138H!      POPX2! SDR(X) FF1;
277 148H!      DSM(S);
278 158H!      LMI(S) RRM;
279 168H!      SDR(A) FF1 JZR(FETCH);
280 1D8H!      PIISA5! LMI(R9) FF1 JCR(PUSA6);
281 1E8H!      SMI! NOP(A) JZR(FETCH);
282 39H!       JJ! NOP(A) JZR(FETCH);
283 49H!       RTN2! ACM(AC) JCR(RTN3);
284 59H!       RTN1! LMI(S) RRM JCC(RTN2);
285 69H!       RTN! DSM(S) JCC(RTN1);
286 79H!       OR1! ACM(AC);
287 89H!       ORR(A) JZR(FETCH);
288 119H!      Ex! SDR(T) FF1;
289 129H!      ILR(X);
290 139H!      SDR(A) FF1;
291 149H!      ILR(T);
292 159H!      SDR(X) FF1 JZR(FETCH);
293 189H!      DN1! SDR(T) FF1;
294 199H!      ILR(Q);
295 1A9H!      SDR(A) FF1;
296 1B9H!      ILR(T);
297 1C9H!      SDR(Q) FF1 JZR(FETCH);
298 1D9H!      PIISA4! ILR(R9) RWM JCR(PUSA5);
299 1E9H!      EQ! ILR(A) JCC(DN1);
300 1AH!       POS1! ILR(P) FF1 JZR(FETCH);
301 2AH!       POSS! NOP(A) JZR(FETCH);
302 /* ENTRY FOR PAGE 2 CENTROID TRACKER */
303 3AH!       CTRACK! NOP(A) JZR(FETCH) PAGE2;
304 4AH!       RTN3! SDR(P) FF1 JZR(FETCH);
305 5AH!       SKN1! NOP(A) JFL(POSS,NEGG);
306 6AH!       SKN! TZR(A) K80000 INH JCC(SKN1);
307 7AH!       XOR1! ACM(AC);
308 8AH!       XNR(A) JCR(DWN5);
309 /* IF A .LT. MEM CF=0 ZF=0. IF A .GT. MEM CF=0 ZF=1
310 IF A .EQ. MEM CF=1 ZF=0 */
311 0AAH!      TSAME! NOP(A) JFL(APOS2,ANEG2);
312 0BAH!      NOCRY! NOP(A) JZR(FETCH);
313 0DAH!      NOCRY2! NOP(A) FF1 STC JZ0(FETCH);
314 0FAH!      APOS2! NOP(A) FF1 STZ JZR(FETCH);
315 10AH!      SP! ILR(P) FF1 JZR(FETCH);
316 11AH!      EP! SDR(T) FF1;
317 12AH!      ILR(P);
318 13AH!      SDR(A) FF1;
319 14AH!      ILR(T);
320 15AH!      SDR(P) FF1 JZR(FETCH);
321 16AH!      NSKIP! NOP(A) JZR(FETCH);
322 17AH!      EVEN! ILR(P) FF1 JZR(FETCH);
323 18AH!      NC! NOP(A) JFL(NOVEF1,OVEF1);
324 19AH!      NOVEF1! NOP(A) JZR(FETCH);

```


XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 3000

RECORD
NUMBER

```

325 1AAH! OVEF2! ILR(P) FF1 JZR(FETCH);
326 1BAH! NC2! NOP(A) JFL(NOVEF3,OVEF3);
327 1CAH! NOVEF3! ILR(P) FF1 JZR(FETCH);
328 1DAH! OVEF4! ILR(P) FF1 JZR(FETCH);
329 1EAH! N11! NOP(A) JZR(FETCH);
330 1BH! NFG1! NOP(A) JZR(FETCH);
331 2BH! NFGG! ILR(P) FF1 JZR(FETCH);
332 3BH! LL! NOP(A) JZR(FETCH);
333 5BH! SKP1! NOP(A) JFL(POS1,NEG1);
334 6BH! SKP! TZR(A) K80000 INH JCC(SKPI);
335 7BH! ANDD! ACM(AC);
336 8BH! ANP(A) JZR(FETCH);
337 0AAH! SAME! CMP(R8) JCR(DWN7);
338 0BBH! CRY! NOP(A) JFL(NOCRY2,CRY2);
339 0DH! CRY2! NOP(A) FF1 STZ JZR(FETCH);
340 0FBH! ANEG2! NOP(A) JZR(FETCH);
341 10BH! NSP! NOP(A) JZR(FETCH);
342 11BH! ES! SDR(T) FF1;
343 12BH! ILR(W);
344 13BH! SDR(A) FF1;
345 14BH! ILR(T);
346 15BH! SDR(W) FF1 JZR(FETCH);
347 16BH! SKIP! ILR(P) FF1 JZR(FETCH);
348 17BH! ODD! NOP(A) JZR(FETCH);
349 18BH! CR! NOP(A) JFL(OVEF2,NOVEF2);
350 19BH! OVEF1! ILR(P) FF1 JZR(FETCH);
351 1ABH! NOVEF2! NOP(A) JZR(FETCH);
352 1BBH! CR22! NOP(A) JFL(OVEF4,NOVEF4);
353 1CBH! OVEF3! NOP(A) JZR(FETCH);
354 1DBH! NOVEF4! NOP(A) JZR(FETCH);
355 1EBH! N12! NOP(A) JZR(FETCH);
356 3CH! Iv! LMI(R9) RRM;
357 4CH! ACM(AC);
358 5CH! ALR(X) JCR(CIX);
359 6CH! SO! NOP(A) JCC(SO1);
360 /* REGISTER EIGHT IS SET TO STACK POINTER WHEN SIGNS OF MEMORY LOCAT
361 AND ACCUMULATOR ARE DIFFERENT
362
363 REGISTER EIGHT IS SET TO THE DIFFERENCE OF THE MEMORY LOCATION AND THE
364 ACCUMULATOR WHEN THE SIGNS OF MEMORY LOCATION AND ACCUMULATOR ARE DIFF
365 ERENT */
366 7CH! C! ACM(AC);
367 8CH! SDR(R8) FF1;
368 9CH! SDR(T) FF1 K80000;
369 0ACH! ILR(A) STC;
370 0BCH! SDR(R5) FF1 K80000;
371 0CCH! ILR(R5) FF0 STZ;
372 0DCH! XNR(T) JCC(CX2);
373 0ECH! CX3! TZR(T) INH;
374 0FCH! TZR(R5) INH JFL(NTSAME,SAME);
375 10CH! CX2! ILR(A) FF0 JCC(CX3);
376 11CH! IFX! ILR(R9);
377 12CH! ALR(X);
378 13CH! TZA(AC) K80000 INH;

```

BEST AVAILABLE COPY

XMAS VERS 2.0 PACER EMULATOR MICROCODE FOR INTEL 3000

RECORD
NUMBER

```

379 14CH!      NOP(A) JFL(NSKIP,SKIP);
380 16CH!      DWN7! LTM(AC);
381 17CH!      SDR(A) FF1 JZR(FETCH);
382 18CH!      S01!  TZR(A) K80000 INH JCF(NC,CR);
383 1ECH!      D1!   LMI(R9) RIN JCC(DWN7);
384 3DH!       AA!   NOP(A) JZR(FETCH);
385 5DH!       CIX!  LMI(AC) RRM JPR(LA STA LX STX A1 S) M D ADM ORI XOM)
386           ADDD C FE FF GG);
387 6DH!       SN0!  NOP(A) JCC(SN01);
388 7DH!       EE!   NOP(A) JZR(FETCH);
389 8DH!       DWN5! CMP(A) JZR(FETCH);
390 0ADH!      DWN7! ALR(R8) FF1;
391 0BDH!      C XXX! TZR(R8) JFL(NOCRY,CRY);
392 11DH!      DFX!  ILR(R9);
393 12DH!      CMA(AC);
394 13DH!      ALR(X) FF1;
395 14DH!      NOP(A) JFL(SP,NSP);
396 19DH!      SN01! TZR(A) K80000 INH JCF(NC2 CR22);
397 1DDH!      D N4! LMI(R9) ROT JZR(FETCH);
398 1EDH!      D0!   ILR(A) JCC(DWN4);
399 3FH!       C0R!  NOP(A) PAGE2 JCC(DWN8);
400 5EH!       DWN8! NOP(A) JCC(SAE);
401 6EH!       SAE!  TZR(A) K00001 INH JCC(SAE1);
402 7FH!       FF!   NOP(A) JZR(FETCH);
403 11EH!      NOPP! NOP(A) JZR(FETCH);
404 16EH!      SAE1! NOP(A) JFL(EVEN,ODD);
405 1EEH!      HH!   NOP(A) JZR(FETCH);
406 0FH!       FETCH! LMI(P) FF1 RRM;
407 1FH!       LTM(AC) K01FFF;
408 2FH!       SDR(R9) FF1 JPX(DIR J L REG I JI LI MSC XK JJ
409           CTRACK LL IX AA COR IO);
410 3FH!       I0!   NOP(A) JCC(I01);
411 6FH!       D0!   NOP(A) JZR(FETCH);
412 7FH!       GG!   NOP(A) JZR(FETCH);
413 11FH!      P7!   NOP(A) SP0 JZR(FETCH);
414 14FH!      I01!  NOP(A) JCC(JK);
415 1DFH!      JK!   NOP(A) JPR(N1,N2,N3,N4,N5,TRAP,N7,RX1,SM1,EO,N11,N12
416           DI DO HH II);
417 1EFH!      II!   NOP(A) JZR(FETCH);
418 EOF

```

NO PROGRAM ERRORS
END OF PROGRAM

BEST AVAILABLE COPY

DISTRIBUTION

	No. of Copies
Defense Documentation Center Cameron Station Alexandria, Virginia 23144	2
Commander US Army Materiel and Readiness Command Attn: DRCRD	1
DRCDL 5001 Eisenhower Avenue Alexandria, Virginia 22333	1
Superior Technical Services, Inc. Attn: D. Creel 4308 Governors Drive Huntsville, Alabama 35805	1
DRSMI-LP, Mr. Voigt	1
DRDMI-X, Dr. J. McDaniel	1
-T, Dr. Kobler	1
-TE, Mr. Lindberg	1
Mr. Pittman	1
-TD, Dr. McCorkle	1
Dr. Grider	1
-TGN, Mr. Kelly	1
-TGG, Mr. Reed	1
-TEO, Mr. Ducote	1
Mr. Farmer	1
Dr. Minor	1
Mr. Helton	1
Mr. Sims	10
Mr. Bates	1
Dr. Kulas	1
Mr. Wylie	1
-ESL, Mr. Schrenk	1
-NT, Mr. Norman	1
-TBD	3
-TI (Record Set)	1
(Reference Copy)	1

FI

9.

D