```
WVS*VORTEX(1).620
      1       *       620/F EMULATION MICROPROGRAM                                    001
      2       *                                                                       002
      3       *       REGISTERS USED;                                                 003
      4       *           0 = 620 A REGISTER                                          004
      5       *           1 = 620 B REGISTER                                          005
      6       *           2 = 620 X REGISTER                                          006
      7       *           3 = I/O DEVICE SELECTION REGISTER (LOW ORDER 6 BITS)        007
      8       *           4 = USED BY HALT LOOP TO HOLD INSTRUCTION REGISTER.  THE    008
      9       *               CONSOLE IS WIRED TO REQUEST DISPLAY OF REGISTER 4 WHEN  009
     10       *               THE INSTRUCTION REGISTER KEY IS PRESSED.                010
     11       *           E = USED FREELY AS WORKING SPACE, AND REQUIRED BY MULTIPLY   011
     12       *               AND DIVIDE INSTRUCTIONS.                                012
     13       *           F = USED FREELY AS WORKING SPACE, AND REQUIRED BY MULTIPLY   013
     14       *               AND DIVIDE INSTRUCTIONS.                                014
     15       *                                                                       015
     16       *       ENTER AT X'1D FOR TESTING PURPOSES IN PAGE 1.                   016
     17       *                                                                       017
     18       *       HEXADECIMAL CONSTANTS.                                          018
     19       *                                                                       019
     20       A       EQU     X'A                                                     020
     21       B       EQU     X'B                                                     021
     22       C       EQU     X'C                                                     022
     23       D       EQU     X'D                                                     023
     24       E       EQU     X'E                                                     024
     25       F       EQU     X'F                                                     025
     26       *                                                                       026
     27       *       SPECIAL REGISTERS.                                              027
     28       *                                                                       028
     29       DR      EQU     3                       I/O DEVICE REGISTER             029
     30       IH      EQU     4                       INSTRUCTION REGISTER (HALT LOOP) 030
     31       S1      EQU     E                                                       031
     32       S2      EQU     F                                                       032
     33       *                                                                       033
     34       *       INTERRUPT MASKS.                                                034
     35       *                                                                       035
     36       M       EQU     E                               ALLOW I/O ONLY IF M/P   036
     37       I       EQU     F                               ALLOW I/O ALWAYS        037
     38       N       EQU     C                               DO NOT ALLOW I/O        038
     39       *                                                                       039
     40       *                                                                       040
     41       *       MACROS                                                          041
     42       *                                                                       042
     43       *       ORIGIN AT NEXT MULTIPLE OF P(1).                                043
     44       *                                                                       044
     45       MORG    MAC                                                             045
     46               ORG     *+P(1)-1                                                046
     47               ORG     */P(1)*P(1)                                             047
     48               EMAC                                                            048
     49       *                                                                       049
     50       *       ILLEGAL INSTRUCTIONS                                            050
     51       *                                                                       051
     52       ILLAD   EQU     2                       ILLEGAL INSTRUCTION INTERRUPT   052
     53       ILLOP   MAC                                                             053
     54               GMSK    /N(ILLOP1),LA1,LB3,RF1,FF9,MK1      P-2->P              054
     55               EMAC                                                            055
     56               EJEC                                                            056
```

```
57        *       SINGLE WORD ADDRESSING INSTRUCTIONS.                                                    057
58        *       PART OF DOUBLE-WORD EXTENDED INSTRUCTIONS.                                              058
59        *                                                                                               059
60                MORG    32  REF BY FSEL(15-13)                                                        060
61        LDA1    GEN     /N(LDB),SF1,AB3,IM8,RF4,BB1,AA0     INCP,IF(P),SET AA&BB                     061
62        ELDA1   GEN     /N(IWAIT),1(M),GF5,LB1,FFA,MF1,      IBR->I,DEC,INTRPT,        FREE           062
63                      CWR1,BB1,AA0                         MIL->A                                     063
64                MORG    2   LDB,LDX REF BY FSEL(12)                                                   064
65        LDB     GEN     /N(IWAIT),1(I),GF5,IM6,LB1,FFA,      IBR->I,DEC,INTRPT,                       065
66                      CMF1,WR1,BB1,AA1                     SEL&RESET CINTF,MIL->B                     066
67        ELDB1   GEN     /N(IWAIT),1(M),GF5,LB1,FFA,MF1,      IBR->I,DEC,INTRPT,        FREE           067
68                      CWR1,BB1,AA1                         MIL->B                                     068
69                ORG     LDA1+4                                                                        069
70        SWA31   GEN     /F(LDB),2(4),FSE,SF1,IM8,RF4         FSEL(12),IF(P),INCP                      070
71        ELDX1   GEN     /N(IWAIT),1(M),GF5,LB1,FFA,MF1,      IBR->I,DEC,INTRPT,        FREE           071
72                      CWR1,BB1,AA2                         MIL->X                                     072
73                ORG     LDB+4                                                                         073
74        LDX     GEN     /N(IWAIT),1(I),GF5,IM6,LB1,FFA,      IBR->I,DEC,INTRPT,                       074
75                      CMF1,WR1,BB1,AA2                     SEL&RESET CINTF,MIL->X                     075
76        EINR1   GEN     /N(INR2),GF8,LB1,FF9,CF3,SH1,BB1     MIL+1->ALU,SAMPLE OVFL    FREE           076
77                ORG     SWA31+4                                                                       077
78        INR1    GEN     /N(EINR1),SF1,IM6,LB1,FFA,MF1        OS(ALU),DOR->ALU                         078
79        INR2    GEN     /F(SS3M),2(8),FSF,SF1,IM8,RF4,       FSEL(14),MIL+1->ALU,      FREE           079
80                      CLB1,FF9,CF3,SH1,BB1                 INCP,IF(P)                                 080
81                MORG    2   SUB,ANA REF BY FSEL(12)                                                   081
82        SUB     GEN     /N(IWAIT),1(I),GFD,IM6,LB1,FF6,      IBR->I,DECODE,EN INT,                    082
83                      CCF3,WR1,BB1,AA0                     SAMPLE OVFL,A-MIL->A                       083
84        ESUB1   GEN     /N(IWAIT),1(M),GFD,LB1,FF6,CF3,      IBR->I,DEC,INTRPT,        FREE           084
85                      CWR1,BB1,AA0                         A-MIL->A,SAMPLE OVFL                       085
86                ORG     SUB+2                                                                         086
87        ANA     GEN     /N(IWAIT),1(I),GF5,IM6,LB1,FFB,      IBR->I,DECODE,EN INT,                    087
88                      CMF1,WR1,BB1,AA0                     A&MIL->A                                   088
89        EANA1   GEN     /N(IWAIT),1(M),GF5,LB1,FFB,MF1,      IBR->I,DEC,INTRPT,        FREE           089
90                      CWR1,BB1,AA0                         A&MIL->A                                   090
91                MORG    2   /S(LDA1,SWA26)                                                           091
92        SWA26   GEN     /N(SWA22),SF1,IMD,RF5                OF(MIL),INCS                             092
93        EORA1   GEN     /N(IWAIT),1(M),GF5,LB1,FFE,MF1,      IBR->I,DEC,INTRPT,        FREE           093
94                      CWR1,BB1,AA0                         A V MIL -> A                               094
95                ORG     INR1+8                                                                        095
96        ORA1    GEN     /N(ORA2),SF1,IM8,RF4                 IF(P),INCP                               096
97        ORA2    GEN     /N(IWAIT),1(I),GF5,IM6,LB1,FFE,      IBR->I,DEC,INTRPT,        FREE           097
98                      CMF1,WR1,BB1,AA0                     A V MIL -> A                               098
99                MORG    2   ADD, ERA REF BY F.SEL(12)                                                 099
100       ADD     GEN     /N(IWAIT),1(I),GFD,IM6,LB1,FF9,      IBR->I,DECODE,EN INT,                    100
101                     CWR1,BB1,AA0                         SAMPLE OVFL,A+MIL->A                       101
102       EADD1   GEN     /N(IWAIT),1(M),GFD,LB1,FF9,WR1,      IBR->I,DEC,INTRPT,        FREE           102
103                     CBB1,AA0                             A+MIL->A,SAMPLE OVFL                       103
104               ORG     ORA1+4                                                                        104
105       SWA35   GEN     /F(ADD),2(4),FSE,SF1,IM8,RF4         FSEL(12),IF(P),INCP                      105
106       SWA22   GEN     /S(LDA1,SWA26),2(C),MT1,FSF,TF3,     C.FSEL(15-13)(MIL15-0),FREE              106
107                     CGFB,LB1,RF3,FFA,MF1,BB1             MIL->OPR                                   107
108               ORG     ADD+4   MORG 16 WITH ADD                                                      108
109       ERA     GEN     /N(IWAIT),1(I),GF5,IM6,LB1,FF6,      IBR->I,DECODE,EN INT,                    109
110                     CMF1,WR1,BB1,AA0                     A XOR MIL -> A                             110
111       EERA1   GEN     /N(IWAIT),1(M),GF5,LB1,FF6,MF1,      IBR->I,DEC,INTRPT,        FREE           111
112                     CWR1,BB1,AA0                         A XOR MIL -> A                             112
113               ORG     SWA35+4                                                                       113
```

```
114       SWA36   GEN     /F(SUB),2(2),FSF,SF1,IM8,RF4        FSEL(12),IF(P),INCP          114
115       SWA21   GEN     /N(SWA22),SF1,IMD                   OF(MIL)                FREE  115
116               MORG     2    REF BY FSEL(12)                                          116
117       MULO    GEN     /N(MUL),LB1,FFA,MF1,WR1,BB1,AAF     MIL=>S2                      117
118       MUL     GMSK    /T(MUL4,MUL1),TF2,GFD,LB3,RF2,      ROSIGN=>MUL4,15=>SC    FREE  118
119                       CFFA,MK0006                                                      119
120               ORG     SWA36+4                                                          120
121       SWA37   GEN     /F(MULO),2(4),FSE,IM1               FSEL(12),WAIT(M)             121
122       ENTER   GEN     /N(SS3MI),SF1,IM8,RF4               IF(P),INCP                   122
123               ORG     MULO+4   MORG 16 WITH MULO                                       123
124       DIVO    GEN     /N(DIV),LB1,FFA,MF1,WR1,WF1,BB1,    MIL=>S2,MIL15=>QS            124
125                       C24(S2)                                                          125
126       DIV     GEN     /T(DIV2,DIV1),TF3,GFF,VF1,AA0       QS=>DIV1,R0(15)=>DSB   FREE  126
127               EJEC                                                                     127
128       *       DOUBLE-WORD EXTENDED INSTRUCTIONS.  NOTE THAT THESE INSTRUCTIONS         128
129       *       ARE DIFFERENT FROM VARIAN'S SINCE AN M-FIELD EQUAL TO ZERO OR 8          129
130       *       DOES NOT CAUSE UNPREDICTABLE RESULTS.                                    130
131       *                                                                                131
132               MORG    32  ELDA ETC REF BY CND FSEL, FAIL TO DEIND                      132
133       EXTBAS  EQU     *                                                                133
134       DL      GEN     /N(DL1),SF1,IM5,LB1,RF4,FF9,CF3,    OF(ALU), DOR+1=>ALU,         134
135                       CSH1,BB0                            INCP                         135
136       ELDA    GEN     /N(ELDA1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                136
137       ELDB    GEN     /N(ELDB1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                137
138       ELDX    GEN     /N(ELDX1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                138
139       EINR    GEN     /N(EINR1),SF1,IM6,LB1,RF4,FFA,MF1   OS(A),DOR=>ALU,INCP          139
140       ESTA    GEN     /N(SS2M),SF1,IM2,AB3,RF4,AA0        OVR(OS),A=>AA,INCP           140
141       ESTB    GEN     /N(SS2M),SF1,IM2,AB3,RF4,AA1        OVR(OS),B=>AA,INCP           141
142       ESTX    GEN     /N(SS2M),SF1,IM2,AB3,RF4,AA2        OVR(OS),X=>AA,INCP           142
143       DS      GEN     /N(DS1),SF1,IM2,AA0                 OVR(OS), A=>ALU              143
144       EORA    GEN     /N(EORA1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                144
145       EADD    GEN     /N(EADD1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                145
146       EERA    GEN     /N(EERA1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                146
147       ESUB    GEN     /N(ESUB1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                147
148       EANA    GEN     /N(EANA1),SF1,IM4,RF1,FF0,CF3,AAE   IF(A),S1+1=>P                148
149       EMUL    GEN     /N(MULO),IM1,RF4                    WAIT(M),INCP                 149
150       EDIV    GEN     /N(DIVO),IM1,RF4                    WAIT(M),INCP                 150
151               MORG    16  REF BY CND FSEL, FAIL TO DEIND                               151
152       DERL    GEN     /S(EXTBAS,DEIND),2(F),FS7,TF3,      C.FSEL(6-3) (QS=0),          152
153                       CSF1,GFF,IM5,LA1,LB1,RF3,FF9,BB0    OF(A),P+DOR=>DOR             153
154       DEIX    GEN     /S(EXTBAS,DEIND),2(F),FS7,TF3,      C.FSEL(6-3) (QS=0),          154
155                       CSF1,GFF,IM5,LB1,RF3,FF9,BB0,AA2    OF(A),X+DOR->DOR             155
156       DEIB    GEN     /S(EXTBAS,DEIND),2(F),FS7,TF3,      C.FSEL(6-3) (QS=0),          156
157                       CSF1,GFF,IM5,LB1,RF3,FF9,BB0,AA1    OF(A),B+DOR->DOR             157
158       DENI    GEN     /S(EXTBAS,DEIND),2(F),FS7,TF3,      C.FSEL(6-3) (QS=0),          158
159                       CSF1,GFF,IM5,LB1,FFA,MF1,BB0        OF(A),DOR=>ALU               159
160               MORG     2   REF BY FSEL(7)                                             160
161       DEPRE   GEN     /F(DERL),2(3),FS4,LB1,RF3,          FSEL(1,0),                   161
162                       CFFA,MF1,WF1,BB1                    MIL=>DOR,MIL15->WS           162
163       DEPOST  GEN     /S(DERL,DEPST1),2(3),FS4,TF3,       C.FSEL(1,0) (MIL15=0),       163
164                       CGFB,LB1,RF3,FFA,MF1,BB1            MIL=>DOR                     164
165               MORG     2   /T(DEPSTI,DEIND1)                                          165
166       DEIND1  GEN     /S(EXTBAS,DEIND),2(F),FS7,TF3,      C.FSEL(6-3) (MIL15=0),       166
167                       CSF1,GFB,IM5,LB1,RF3,FFA,MF1,BB1    OF(A),MIL=>DOR               167
168       DEPSTI  GEN     /S(DERL,DEIND),2(3),FS4,TF3,SF3,    C.FSEL(1,0) (MIL15=0),       168
169                       CGFB,IM5,LB1,RF3,FFA,MF1,BB1        CND(MIL15=1) OF(ALU),        169
170       *                                                   MIL=>DOR                     170
```

```
171                MORG    2   /S(EXTBAS,DEIND)                                                    171
172     DEIND      GEN     /T(DEPSTI,DEIND1),TF3,GFF,IM1,RF5 WAIT(M),QS=>DEIND1,INCS              172
173     DL1        GEN     /N(ELDB1),SF1,IM8,LB1,RF4,FFA,     IF(P),INCP,MIL->A        FREE    173
174                       CMF1,WR1,BB1,AA0                                                        174
175                MORG    2   /S(DERL,DEPST1)                                                   175
176     DEPST1     GEN     /N(DEIND),SF1,IM5,LB1,FFA,MF1,BB0 OF(ALU),DOR=>ALU                    176
177                EJEC                                                                          177
178     *          SKIP IF REGISTER EQUAL.  THIS CODE IS SOMEWHAT DIFFERENT FROM                 178
179     *          VARIAN'S SINCE SETTING OF MORE THAN ONE OF THE A, B OR X BITS                 179
180     *          DOES NOT CAUSE UNPREDICTABLE RESULTS.                                         180
181     *                                                                                        181
182     SRE5X      GEN     /N(SRE60),IM1,LA1,RF1,FF1,BB0     WAIT(M),P V A => P       FREE    182
183                MORG    4   REF BY FSEL(1-0)                                                 183
184     SRE20      GEN     /S(SRE50,SRE30),2(7),FS7,TF3,SF2, CND(MIL15=0) FSEL(5-3),           184
185                       CGFB,IM5,LB1,LA1,FF9,BB1           CND(MIL15=0) OF(ALU),             185
186     *                                                    P+MIL->ALU                        186
187     SRE21      GEN     /S(SRE50,SRE30),2(7),FS7,TF3,SF2, CND(MIL15=0) FSEL(5-3),           187
188                       CGFB,IM5,LB1,FF9,BB1,AA2           CND(MIL15=0) OF(ALU),             188
189     *                                                    X+MIL->ALU                        189
190     SRE22      GEN     /S(SRE50,SRE30),2(7),FS7,TF3,SF2, CND(MIL15=0) FSEL(5-3),           190
191                       CGFB,IM5,LB1,FF9,BB1,AA1           CND(MIL15=0) OF(ALU),             191
192     *                                                    B+MIL->ALU                        192
193     SRE23      GEN     /S(SRE50,SRE30),2(7),FS7,TF3,SF1, CND(MIL15=0) FSEL(5-3),           193
194                       CGFB,IMD                           OF(MIL)                           194
195                MORG    16  SRE5+ REF BY CND FSEL                                           195
196     SRE50      GEN     /N(SRE60),IM1,RF1,SH1             WAIT(M),0->P                      196
197     SRE51      GEN     /N(SRE60),IM1,RF1,AA0             WAIT(M),A=>P                      197
198     SRE52      GEN     /N(SRE60),IM1,RF1,AA1             WAIT(M),B=>P                      198
199     SRE53      GEN     /N(SRE60),IM1,RF1,FF1,BB1,AA0     WAIT(M),A V B -> P                199
200     SRE54      GEN     /N(SRE60),IM1,RF1,AA2             WAIT(M),X=>P                      200
201     SRE55      GEN     /N(SRE60),IM1,RF1,FF1,BB2,AA0     WAIT(M),A V X -> P                201
202     SRE56      GEN     /N(SRE60),IM1,RF1,FF1,BB2,AA1     WAIT(M),B V X -> P                202
203     SRE57      GEN     /N(SRE5X),RF1,FF1,BB2,AA1         B V X -> P                        203
204                MORG    2   MORG 32 WITH SRE50                                              204
205     SRE30      GEN     /N(SRE40),SF1,IMD                 OF(MIL)                           205
206     SRE60      GEN     /N(SRE70),GF2,LB1,LA1,FF6,MF1,BB1 MIL XOR P, SAMPLF        FREE    206
207                MORG    2   MORG 32 WITH SRE50                                              207
208     SRE40      GEN     /F(SRE20),2(3),FS4,IM1,RF5        FSEL(1-0),WAIT(M),INCS            208
209     SRE70      GEN     /T(SRE80,SS2M),TF2,SF2,GF9,IM4,   CND(NO ALUZ) IF(ALU),     FREE    209
210                       CLB1,RF1,BB0                       DOR->P                            210
211                EJEC                                                                        211
212     *          REGISTER - TO - REGISTER TRANSFER INSTRUCTIONS.                              212
213     *                                                                                      213
214     REG30      GEN     /F(REG20),2(7),FS4,SF1,IM8,LB1,RF3,FFE,MF1                          214
215     REG41      GEN     /N(REG20),SF1,GF6,IM8,LB1,FF9,WR1,SH5                      FREE     215
216     REG42      GEN     /N(REG20),SF1,GF6,IM8,LB1,FF9,WR1,SH5,AA1                  FREE     216
217     REG43      GEN     /N(REG21),SF1,GF6,IM8,LB1,FF9,WR1,SH5,AA1                  FREE     217
218                MORG    16  REF BY CND FSEL(2-0)                                            218
219     REG20      GEN     /N(IWAIT),1(1),GF5,IM6                                              219
220     REG21      GEN     /N(IWAIT),1(1),GFD,IM6,LB1,FF9,WR1,SH5                              220
221     REG22      GEN     /N(IWAIT),1(1),GFD,IM6,LB1,FF9,WR1,SH5,AA1                          221
222     REG23      GEN     /N(REG41),SF1,GF6,IM1,LB1,FF9,WR1,SH5,AA1                           222
223     REG24      GEN     /N(IWAIT),1(1),GFD,IM6,LB1,FF9,WR1,SH5,AA2                          223
224     REG25      GEN     /N(REG41),SF1,GF6,IM1,LB1,FF9,WR1,SH5,AA2                           224
225     REG26      GEN     /N(REG42),SF1,GF6,IM1,LB1,FF9,WR1,SH5,AA2                           225
226     REG27      GEN     /N(REG43),SF1,GF6,IM1,LB1,FF9,WR1,SH5,AA2                           226
227     ********************   D E C O D E R   ********************                             227
```

```
228                ORG     X'58     DECODER REFERENCES NEXT 8 INSTRUCTIONS          *      228
229       REG18    GEN      /S(REG20,REG20),2(7),FS4,TF2,SF1,IM8,RF7,SH1           *      229
230       REG1A    GEN      /S(REG20,REG20),2(7),FS4,TF2,SF1,IM8,RF7,AA1           *      230
231       REG1C    GEN      /S(REG20,REG20),2(7),FS4,TF2,SF1,IM8,RF7,AA2           *      231
232       REG1E    GEN      /S(REG20,REG20),2(7),FS4,TF2,SF1,IM8,RF7,FFE,          *      232
233                        CMF1,BB2,AA1                                            *      233
234       REG19    GEN      /S(REG20,REG20),2(7),FS4,TF2,SF1,IM8,RF7               *      234
235       REG1B    GEN      /S(REG20,REG20),2(7),FS4,TF2,SF1,IM8,RF7,FFE,          *      235
236                        CMF1,BB1                                                *      236
237       REG1D    GEN      /S(REG20,REG20),2(7),FS4,TF2,SF1,IM8,RF7,FFE,          *      237
238                        CMF1,BB2                                                *      238
239       REG1F    GEN      /T(REG30,REG20),TF2,SF2,IM8,RF7,FFE,MF1,BB2,AA1        *      239
240       ********************************************************************************240
241                EJEC                                                                   241
242       *        HALT LOOP PROCESSING.  CONTROL GETS HERE VIA HALT OR ILLEGAL           242
243       *        INSTRUCTIONS, VIA THE RESET KEY AND VIA THE STEP KEY.                  243
244       *                                                                               244
245                MORG     16  REF BY FSEL(5-4), (5-4)=11 IMPOSSIBLE, REG FLD SELECT     245
246       HREN1    GEN      /N(HMEM)                          NOP (CAN'T CHANGE STAT)      246
247       HREN2    GEN      /N(HMEM),LB1,RF1,FFA,MF1          DOR->P                       247
248       HREN3    GEN      /N(HMEM),LB1,FFA,MF1,WR1          DOR->R**                     248
249       *        HMDP1 IS NOT REFERENCED BY FSEL.                                       249
250       HMDP1    GEN      /N(HRDP1),IM1,RF4,AB3,BB1         WAIT(M),INCP,"MIL"->BB FREE  250
251                MORG     4   REF BY FSEL(7-6), (7-6)=10 IMPOSSIBLE                     251
252       HREN     GEN      /F(HREN1),1(0),2(3),FS8,AB2,MR1   FSEL(5-4),(3-0)->AA          252
253       HMEMA    GEN      /N(HLT1),1(4),GF4,IMC             SAMPLE INTERRUPTS            253
254       *        HMEN4 IS NOT REFERENCED BY FSEL                                        254
255       HMEN4    GEN      /N(HMEM),LB1,RF4,FFA,MF1          WAIT(M),INCP,DOR->ALU  FREE  255
256       HMEN3    GEN      /N(HMEN4),SF1,IMA                 OS(P)                        256
257       ********************  D E C O D E R  ********************************************257
258                ORG      X'68     DECODER REFERENCES NEXT 8 INSTRUCTIONS.        *      258
259       IMMREL   GEN      /F(IMPRE),2(1),FSB,SF1,IM4,LA1,   P+1->S1,IF(A),0->WS,   *      259
260                        CFF0,CF3,WR1,WF1,24(S1)           FSEL(7)                *      260
261       IMMIX    GEN      /F(IMPRE),2(1),FSB,SF1,IM4,LA1,   P+1->S1,IF(A),0->WS,   *      261
262                        CFF0,CF3,WR1,WF1,24(S1)           FSEL(7)                *      262
263       IMMIB    GEN      /F(IMPRE),2(1),FSB,SF1,IM4,LA1,   P+1>S1,IF(A),0->WS,    *      263
264                        CFF0,CF3,WR1,WF1,24(S1)           FSEL(7)                *      264
265       IMMNI    GEN      /N(IMNI),SF1,IM4,LA1,FF0,CF3,     P+1->S1,IF(A)          *      265
266                        CWR1,24(S1)                                              *      266
267       DE#1     GEN      /F(DEPRE),2(1),FSB,SF1,IM4,LA1,   P+1->S1,IF(A),0->WS,   *      267
268                        CFF0,CF3,WR1,WF1,XF1,AAE          FSEL(7),0->CINTF       *      268
269       DE#2     GEN      /F(DEPRE),2(1),FSB,SF1,IM4,LA1,   P+1->S1,IF(A),0->WS,   *      269
270                        CFF0,CF3,WR1,WF1,XF1,AAE          FSEL(7),0->CINTF       *      270
271       DE#3     GEN      /F(DEPRE),2(1),FSB,SF1,IM4,LA1,   P+1->S1,IF(A),0->WS,   *      271
272                        CFF0,CF3,WR1,WF1,XF1,AAE          FSEL(7),0->CINTF       *      272
273       DE#4     GEN      /F(DEPRE),2(1),FSB,SF1,IM4,LA1,   P+1->S1,IF(A),0->WS,   *      273
274                        CFF0,CF3,WR1,WF1,XF1,AAE          FSEL(7),0->CINTF       *      274
275       ********************************************************************************275
276                MORG     16  REF BY FSEL(5-4), (5-4)=11 IMPOSSIBLE, REG FLD SELECT     276
277       HRDP1    GEN      /N(HRPT),LB1,RF3,FFA,MF1          STATUS->DOR                  277
278       HRDP2    GEN      /N(HFPT),LA1,RF3                  P->DOR                       278
279       HRDP3    GEN      /N(HFPT),RF3                      R**->DOR                     279
280       *        HFPT IS NOT REFERENCED BY FSEL.                                        280
281       HFPT     GMSK     /N(HFPT1),1(7),AB2,IMF,LB3,FFA,   START IO @ X'1E       FREE 281
282                        CMKBFC0                           X'403F->ALU (DTO)            282
283                MORG     2   /T(HSTT6,HSTT2)                                         283
284       HSTT2    GEN      /F(SS2),2(1),FS3,SF1,IM8          FSEL(STEP),IF(P)             284
```

```
285       HSTT6   GEN     /N(HSTT7),IM4,24(IH)                IH=>IBR,MIL                     285
286       HSTT1   GEN     /T(HSTT6,HSTT2),TF3,GF9             ALUZ=>HSTT2             FREE    286
287       HSTT5   GEN     /N(HLT),GF1                         IBR=>I                  FREE    287
288               MORG    8   REF BY FSEL(8=6)                                            288
289       HRDP    GEN     /F(HRDP1),1(0),2(3),FS8,MR1,AB2,    FSEL(5=4), (3=0)=>AA            289
290                       CBB3                                STATUS=>BB                      290
291       HMEM    GEN     /N(HLT1),1(4),GF4,IMC               SAMPLE INTERRUPTS               291
292       HSTT    GMSK    /N(HSTT1),GF2,LB3,FFB,MK01FF,       IH&X'FE00,SAMPLE ALU            292
293                       C16(IH)                             (WAS IT A HALT INST?)           293
294       HMEN    GMSK    /N(HMEN1),1(3),AB2,IMF,LB3,FFA,     START IO @ X'E,                 294
295                       CMKDFC0                             X'203F=>ALU (DTI)               295
296       ABL10   GMSK    /N(ABL20),SF1,GF4,IM6,LB3,RF1,      OS(ALU),X'80=>P,                296
297                       CFFA,MKFF7F                         RESET OVERFLOW                  297
298       HMDP    GEN     /N(HMDP1),SF1,IM9                   OF(P)                           298
299       ABL11   GMSK    /N(ABL21),SF1,GF2,IM6,LB3,RF1,      OS(ALU),X'80=>P                 299
300                       CFFA,MKFF7F                         SET OVERFLOW                    300
301       ABL30   GMSK    /N(ABL31),SF1,IM6,LB3,RF1,FFA,      OS(ALU),X'258=>P                301
302                       CMKFDA7                                                             302
303       ********************    D E C O D E R    ************************                 303
304               ORG     X'80    DECODER REFERENCES 'HLT'                        *           304
305       HLT     GEN     /F(HLT8),2(1),FS1                   FSEL(CINTF)         *           305
306       ********************************************************************************  306
307       HLT2    GMSK    /N(HLT3),LB3,RF3,FFA,MKFFC0         0'77=>DOR               FREE    307
308       HLT3    GEN     /N(HLT4),IM2,LB1,FFB,MF1,WR1,BB0,   WAIT(IO),DOR&DR=>DR     FREE    308
309                       C24(DR)                                                             309
310       HLT4    GEN     /N(HLT5),IM4,LB1,RF3,FFA,MF1,BB2    IOR=>DOR,ALU=>IBR       FREE    310
311       HLT5    GEN     /N(HLT6),GF1,LB1,FFA,MF1,BB0        DOR=>ALU,IBR=>IR        FREE    311
312       HLT6    GEN     /F(HRDP),2(7),FSA                   FSEL(8=6)               FREE    312
313       HSTT7   GEN     /N(HSTT8),GF1,24(IH)                IH=>ALU,IBR=>IR         FREE    313
314       HSTT8   GEN     /N(HSTT9),SF1,IM8                   IF(P)                   FREE    314
315       HSTT9   GEN     /N(IWAIT),1(4),GF5                  IBR=>I,DECODE,ETC       FREE    315
316       HFPT1   GEN     /N(HMEN),IM2,LB1,FFA,MF1            WAIT(IO),DOR=>ALU       FREE    316
317       HMEN1   GEN     /N(HMEN2),IM2                       WAIT(IO)                FREE    317
318       HMEN2   GEN     /F(HREN),2(3),FSA,LB1,RF3,FFA,      FSEL(7=6),IOR=>DOR      FREE    318
319                       CMF1,BB2                                                            319
320               EJEC                                                                        320
321       *       DIVIDE INSTRUCTION.                                                         321
322       *                                                                                   322
323       ********************    D E C O D E R    ************************                 323
324               ORG     X'90    DECODER REFERENCES 'JMP'                        *           324
325       JMP     GEN     /T(JMP1,SS1M),TF2,SF3,GF5,IMC,      620=>JMP1,RESET CINTF,*         325
326                       CRF4,XF1                            CND(620) IF(MIL), INCP*         326
327       ********************************************************************************  327
328       DIV5    GEN     /N(DIV6),GF2,RF3,FF2,CF3,SH1,BB1    =B=>DOR,SAMPLE          FREE    328
329               MORG    2   /T(DIV2,DIV1)                                                  329
330       DIV1    GEN     /N(DIV2),FF6,CF3,WR1,SH1,BBF,AAF    =RF=>RF                         330
331       DIV2    GEN     /T(DIV4,DIV3),TF2,GFA,FF2,CF3,      DSB=>DIV4,=RF=>RF               331
332                       CWR1,SH1,BBF,AAE                                                    332
333               MORG    4   BE CAREFUL:  /S(DIV13&DIV14,DIV12) SELECTS DSB                 333
334       DIV13   GEN     /S(SS3M,DIV15),2(8),FSF,TF3,SF2,    FSEL(15) IF NO QS               334
335                       CGFF,IM8,RF4                        IF(P) IF NO QS, INCP            335
336       DIV6    GEN     /N(DIV7),FF2,CF1,WR1,SC1,WF1,SH1    DOR(RT),CARRY=A=>A      FREE    336
337               ORG     DIV13+2   REF BY FSEL(DSB)                                         337
338       DIV14   GEN     /S(SS3M,DIV15),2(8),FSF,TF2,SF3,    FSEL(15) IF QS,                 338
339                       CGFF,IM8,RF4,FF6,CF3,WR1,SH1,BB0,   IF(P) IF QS, INCP               339
340                       CAA0                                =A=>A                           340
341       DIV7    GEN     /N(DIV8),GF2,FF9,WR1,SC1,XF2,BBE    FIRST DIVIDE STEP       FREE    341
```

```
342                MORG    2   /S(DIV13,DIV12)                                                    342
343     DIV12      GEN     /T(DIV14,DIV13),TF2,GFA,FF9,WR1,  DSB=>DIV14,A+RF=>A                 343
344                       CBBF,AAO                                                               344
345     DIV8       GMSK    /T(DIV9,DIV10),TF3,GFD,LB3,RF2,   RO(15)=>DIV10,-14=>SC  FREE        345
346                       CFFA,MKE                                                               346
347                MORG    2   /T(DIV4,DIV3)                                                    347
348     DIV3       GEN     /N(DIV7),RF3,AA1                  B=>DOR                           348
349     DIV4       GEN     /N(DIV5),LA2,WR1,AA1              B(L)=>B,0=>DLA0                  349
350                MORG    2   /T(DIV9,DIV10), /T(DIV11,DIV10)                                   350
351     DIV10      GEN     /T(DIV11,*),TF2,GFC,MR1,LA2,RF5,  DIVIDE STEP                      351
352                       CFF9,WR1,SC1,XF2,SH2,BBF                                               352
353     DIV11      GEN     /S(DIV13,DIV12),2(6),FS2,TF3,GFD, FSEL(DSB) IF R0(15)=0            353
354                       CLB1,FFA,MF1,WR1,BB0,AA1          DOR=>B                           354
355     DIV9       GEN     /N(DIV10),SF1,GF2                 SET OVERFLOW                     355
356                EJEC                                                                       356
357     *          JUMP AND MARK INSTRUCTION.                                                 357
358     *                                                                                     358
359     ********************   D E C O D E R   ************************                       359
360                ORG     X'A0     DECODER REFERENCES 'JMRK'                        *        360
361     JMRK       GEN     /T(JMRK1,JMRK5),TF2,SF3,GF5,IMD,  620=>JMRK1,              *        361
362                       CXF1                              CND(620) OF(MIL)         *        362
363     ********************************************************************************       363
364     JMRK3      GEN     /N(SS2M),SF1,IM4,LB1,RF1,FF9,CF3, IF(ALU),MIL+1=>P       FREE      364
365                       CSH1,BB1                                                               365
366                MORG    2   /T(JMRK1,JMRK5)                                                366
367     JMRK5      GEN     /N(SS2M),SF1,IM8,RF4              IF(P),INCP                       367
368     JMRK1      GEN     /T(JMRK2,JMRK4),TF3,SF2,GFB,IM2,  MIL15=>JMRK4,P+1=>ALU,           368
369                       CLA1,CF3                          CND(MIL15=0) OS(MIL)             369
370                MORG    2   /T(JMRK2,JMRK4)                                                370
371     JMRK4      GEN     /N(JMRK1),SF1,IMD,RF5             OF(MIL),INC SC                   371
372     JMRK2      GEN     /N(JMRK3),IM1,LA1,CF3             WAIT(M),P+1=>ALU                 372
373                EJEC                                                                       373
374     *          EXECUTE REMOTE INSTRUCTION.                                                374
375     *                                                                                     375
376     ********************   D E C O D E R   ************************                       376
377                ORG     X'B0     DECODER REFERENCES 'XEC'                         *        377
378     XEC        GEN     /T(XEC1,XEC3),TF2,SF3,GF5,IMC,XF1 620=>XEC1,RESET CINTF,*          378
379     *                                                    CND(620) IF(MIL)         *        379
380     ********************************************************************************       380
381     XEC2       GEN     /N(XEC1),SF1,IMC,RF5              IF(MIL),INC SC                   381
382                MORG    2   /T(XEC1,XEC3)                                                  382
383     XEC3       GEN     /N(SS2),SF1,IM8,RF4               IF(P),INCP                       383
384     XEC1       GEN     /T(XEC2,SS2),TF2,GFB              MIL15=>XEC2                      384
385                MORG    2   /T(XEC2,SS2), FSEL(STEP)                                       385
386     SS2        GEN     /N(SS3),SF1,IM8,RF4               IF(P),INCP                       386
387     HSTT4      GEN     /N(HSTT5),IM1,RF4                 WAIT(M),INCP                     387
388     SS3        GEN     /N(IWAIT),1(N),GF5,IM6            DECODE, EN INTS, ETC     FREE    388
389                EJEC                                                                       389
390     *          SINGLE WORD ADDRESSING STORE INSTRUCTIONS.                                 390
391     *                                                                                     391
392     DORTOP     GEN     /N(RESET),RF1,LB1,FFA,MF1,BB0     DOR=>P FOR DEBUGGING     FREE    392
393                MORG    8   REF BY FSEL(13-12), 00 NOT POSSIBLE.                          393
394     STABX1     EQU     *                                                                  394
395     DS1        GEN     /N(DS2),IM1,AA0                   WAIT(M), A=>ALU          FREE    395
396     DS2        GEN     /N(ESTB),SF1,IM6,LB1,FF9,CF3,SH1, OS(ALU),DOR+1=>ALU       FREE    396
397                       CBB0                                                                   397
398                ORG     STABX1+2                                                           398
```

```
399       STA1    GEN     /T(SWA50,SS2MI),TF2,SF3,GFB,AB3,  A=>ALU,SET AA,INCS          399
400                      CIM1,RF5,AA0                       MIL15=>OVR(OF),=>SWA50     400
401               ORG     STA1+2                                                        401
402       STB1    GEN     /T(SWA50,SS2MI),TF2,SF3,GFB,AB3,  B=>ALU,SET AA,INCS          402
403                      CIM1,RF5,AA1                       MIL15=>OVR(OF),=>SWA50     403
404               ORG     STB1+2                                                        404
405       STX1    GEN     /T(SWA50,SS2MI),TF2,SF3,GFB,AB3,  X=>ALU,SET AA,INCS          405
406                      CIM1,RF5,AA2                       MIL15=>OVR(OF),=>SWA50     406
407               MORG    8   REF BY FSEL(13-12), 00 NOT POSSIBLE.                       407
408       STABX2  EQU     *                                                             408
409       ********************   D E C O D E R   ************************************   409
410               ORG     X'C0    DECODER REFERENCES 'SHFT'                      *      410
411       SHFT    GMSK    /N(SHFT1),GF2,LB2,RF7,FFA,MKFFE0  I(4-0)=>DOR,INCP,SAMPL*     411
412       ********************************************************************************412
413               ORG     STABX2+2                                                      413
414       STA2    GEN     /N(SS2MI),AB3,AA0                  A=>ALU,SET AA              414
415               ORG     STA2+2                                                        415
416       STB2    GEN     /N(SS2MI),AB3,AA1                  B=>ALU,SET AA              416
417               ORG     STB2+2                                                        417
418       STX2    GEN     /N(SS2MI),AB3,AA2                  X=>ALU,SET AA              418
419               EJEC                                                                  419
420       *       INTERRUPT SERVICE.                                                    420
421       *       THE STATES WHICH ENABLE THE DECODER ALSO ENABLE INTERRUPTS.  IF       421
422       *       A SINGLE WORD INSTRUCTION WAS AT THE INTERRUPT ADDRESS, THE NEXT      422
423       *       WORD IS IN THE INSTRUCTION BUFFER BUT SHOULD NOT BE PROCESSED.        423
424       *       PROCESSING OF THIS WORD IS INHIBITED BY TRANSFERRING TO INTBAS        424
425       *       IF CINTF=1 AND IM=6.  IF CINTF=0 AND NO INTERRUPT IS PENDING, THE     425
426       *       INSTRUCTION IS DECODED.  IF AN INTERRUPT IS PENDING AND CINTF=0       426
427       *       OR IM NOT =6, CONTROL GOES TO INTBAS+7, WHERE THE INTERRUPT           427
428       *       MICROPROGRAM IN I/OCS IS INITIATED.  IF THE INTERRUPT WASN'T          428
429       *       REALLY AN INTERRUPT, BUT INSTEAD WAS A DMA REQUEST, INTERRUPT         429
430       *       PROCESSING MUST BE ABORTED AND THE PIPELINE RESTARTED.  IF IT         430
431       *       REALLY IS AN INTERRUPT, CONTROL GOES TO INTBAS+1 WHERE INTERRUPT      431
432       *       SERVICING IS COMPLETED.                                               432
433       *                                                                             433
434       ILLOP1  GMSK    /N(ILLOP2),SF1,IM4,LB3,RF3,        ILLAD=>DOR, IF(ALU)        434
435                      C15(=1-ILLAD)                                                 435
436       ILLOP2  GEN     /N(INT4),SF1,IM4,LB1,FF9,CF3,WR1,  DOR+1=>RF, IF(ALU)         436
437                      CSH1,BB0,AAF                                                  437
438       INT2    GEN     /N(INT3),SF1,IM4,LB1,FFA,MF1,      IF(ALU),IOR=>RF      FREE 438
439                      CWR1,BB2,AAF                                                  439
440       INT3    GEN     /N(INT4),SF1,IM4,CF3,WR1,AAF       IF(ALU),RF+1=>RF     FREE 440
441       INT4    GEN     GF5,IM7,FFA,MF1,WR1,BBF,AAE        IBR=>I,DECODE,       FREE 441
442       *                                                  1=>CINTF,S2=>S1              442
443               MORG    16  INTERRUPT HARDWARE USES INTBAS, INTBAS+1, INTBAS+7        443
444       INTBAS  EQU     *                                                             444
445               MORG    2   /T(SWA50,SS2MI)                                         445
446       SS2MI   GEN     /N(SS3MI),SF1,IM8,RF4              (AA)=>ALU,IF(P),INCP       446
447               ORG     INTBAS+1 INTERRUPT SECOND TRY COMES HERE                      447
448       INT1    GEN     /N(INT2),IM2,RF1,FFF,AAE           WAIT(IO),RE-1=>P          448
449       SWA50   GEN     /F(STA1-2),2(6),FSF,SF1,IME        FSEL(13-12),OS(MIL)       449
450       ABORT   GMSK    /N(ABORT1),IM4,LB2,FFA,MK0         I=>IBR                    450
451       ABORT1  GMSK    /N(SS2MI),IM4,LB2,FFA,MK0          I=>IBR               FREE 451
452       *       HLT1 AND HLT7 ARE REFERENCED WITH INTERRUPTS ENABLED.                 452
453       HLT1    GMSK    /N(HLT2),1(3),AB2,IM4,LB3,FFA,     START I/O AT X'E'.        453
454                      CMKEFC0                            X'103F=>ALU               454
455       HLT7    GEN     /N(HLT1),LB1,FFA,MF1,WR1,24(IH)    DOR=>IH                   455
```

```
456                ORG     INTBAS+7 INTERRUPT FIRST TRY COMES HERE                                  456
457       IWAIT    GEN     2(ABORT),1(7),MT1,GF4,MR1,IME,    ENABLE INTERRUPTS,                     457
458                       CLA1,RF1,FFF,WR1,AAE     DISABLE DEC, START IO, P-1->P,RE                 458
459       ************************   D E C O D E R   ************************                        459
460                ORG     X'D8     DECODER REFERENCES NEXT 8 INSTRUCTIONS          *               460
461       REG10    GEN     /F(REG20),2(7),FS4,SF1,IM8,RF7,SH1                       *               461
462       REG12    GEN     /F(REG20),2(7),FS4,SF1,IM8,RF7,AA1                       *               462
463       REG14    GEN     /F(REG20),2(7),FS4,SF1,IM8,RF7,AA2                       *               463
464       REG16    GEN     /F(REG20),2(7),FS4,SF1,IM8,RF7,FFE,BB2,AA1,MF1           *               464
465       REG11    GEN     /F(REG20),2(7),FS4,SF1,IM8,RF7                           *               465
466       REG13    GEN     /F(REG20),2(7),FS4,SF1,IM8,RF7,FFE,BB1,MF1               *               466
467       REG15    GEN     /F(REG20),2(7),FS4,SF1,IM8,RF7,FFE,BB2,MF1               *               467
468       REG17    GEN     /N(REG30),RF7,FFE,BB2,AA1,MF1                            *               468
469       ************************************************************************                 469
470                EJEC                                                                             470
471       *        JUMP TO SUBROUTINE, SOME OF BIT TEST, SOME OF SKIP IF REGISTER                   471
472       *        EQUAL, SOME OF TRANSFER SWITCHES TO A.                                           472
473       *                                                                                         473
474                MORG    16  JSR40,JSR41 REF BY CND FSEL, FAIL TO JSR30                           474
475       JSR40    GEN     /N(SS3M),SF1,IM8,LB1,FFA,MF1,WR1, IF(P),DOR->X                           475
476                       CBB0,AA2                                                                 476
477       JSR41    GEN     /N(SS3M),SF1,IM8,LB1,FFA,MF1,WR1, IF(P),DOR->B                           477
478                       CBB0,AA1                                                                 478
479                MORG    2   FAIL OF CND FSEL TO JSR40,JSR41                                    479
480       JSR30    GEN     /N(JSR20),SF1,IMC,RF5              IF(MIL),INCSC                         480
481       ************************   D E C O D E R   ************************                        481
482                ORG     X'E3     DECODER REFERENCES 'BT10'                       *               482
483       BT10     GEN     /F(BT20),2(F),FS4,RF4,XF1          FSEL(3-0),INCP,      *               483
484       *                                                   RESET CINTF          *               484
485       ************************************************************************                 485
486                MORG    2   /T(BT51,BT52)                                                       486
487       BT52     GEN     /N(SS3M),SF1,IM8                   IF(P)                                 487
488       BT51     GEN     /N(BT50),SF1,IMC,RF5               IF(MIL),INCSC                         488
489       JSR20    GEN     /S(JSR40,JSR30),2(1),FS5,TF3,GFB,  MIL=>JSR30,FSEL(1),     FREE          489
490                       CLB1,RF1,FF9,CF3,XF2,SH1,BB1       MIL+1->P,JUMP SIGNAL                  490
491       ************************   D E C O D E R   ************************                        491
492                ORG     X'E7     DECODER REFERENCES 'JSR10'                      *               492
493       JSR10    GEN     /N(JSR20),SF1,IMC,LA1,RF3,CF3,XF1  IF(MIL),P+1->DOR,    *               493
494       *                                                   RESET CINTF          *               494
495       ************************************************************************                 495
496       TSA1     GEN     /N(TSA2),IM2                       WAIT(IO)                  FREE          496
497       TSA2     GEN     /N(SS3M1),SF1,IM8,LB1,RF4,FFA,     IF(P),INCP,IOR->A         FREE          497
498                       CMF1,WR1,BB2,AA0                                                         498
499       SRE81    GEN     /N(SS2M),SF1,IM8,RF4,XF2           IF(P),INCP,J=SIG          FREE          499
500       ************************   D E C O D E R   ************************                        500
501                ORG     X'EB     DECODER REFERENCES 'SRE10'                      *               501
502       SRE10    GEN     /F(SRE20),2(3),FS4,IM1,LA1,RF3,    F,SEL(1-0),WAIT(M),   *               502
503                       CCF3,XF1                           P+1=>DOR,0->CINTF     *               503
504       ************************************************************************                 504
505                EJEC                                                                             505
506       *        INDEXED JUMP INSTRUCTION.                                                        506
507       *        NOTE THAT THIS VERSION OF IJMP DIFFERS FROM VARIAN'S VERSION                     507
508       *        SINCE BIT 2 MAY BE USED TO INDICATE PRE- OR POST-INDEXING, AS MAY                508
509       *        BE DONE USING BIT 7 WITH OTHER EXTENDED ADDRESSING INSTRUCTIONS.                 509
510       *        ALSO, BITS 4&3 HAVE THE FOLLOWING MEANING:                                       510
511       *            00 - JUMP                                                                    511
512       *            10 - XEC                                                                     512
```

```
513     *           01 - JSR,X                                                  513
514     *           11 - JSR,B                                                  514
515             MORG    2   REF BY FSEL(2)                                     515
516     IJPRE   GEN     /F(IJRL),2(3),FS4,LB1,RF3,          FSEL(1-0),          516
517                     CFFA,MF1,WF1,BB1                    MIL->DOR,MIL15->QS  517
518     IJPOST  GEN     /S(IJRL,IJIND),2(3),FS4,TF3,SF3,    C.FSEL(1-0) (MIL15=0),518
519                     CGFB,IM5,LB1,RF3,FFA,MF1,BB1        C.OF(ALU) (MIL15=1),519
520     *                                                   MIL->DOR            520
521     ************************   D E C O D E R   ************************** 521
522             ORG     X'EF    DECODER REFERENCES 'IJMP'                     * 522
523     IJMP    GEN     /F(IJPRE),2(1),FS6,IM1,FF0,WF1,     FSEL(2),0->QS,      * 523
524                     CXF1,SH1                            0->CINTF            * 524
525     ************************************************************************ 525
526     *       IJRL,IJIX,IJIB,IJNI,IJIND1 ALL DO CND FSEL (6,4,3) BUT BIT 6 = 1. 526
527             MORG    16    REF BY CND FSEL(1-0)                            527
528     IJRL    GEN     /S(IJXIT,IJIND),2(B),FS7,TF3,SF1,   QS=>IJIND,P+DOR->DOR,528
529                     CGFF,IM4,LA1,LB1,RF3,FF9,XF2,BB0    IF(A),EN JUMP SIGNAL529
530     IJIX    GEN     /S(IJXIT,IJIND),2(B),FS7,TF3,SF1,   QS=>IJIND,X+DOR->DOR,530
531                     CGFF,IM4,LB1,RF3,FF9,XF2,BB0,AA2    IF(A),EN JUMP SIGNAL531
532     IJIB    GEN     /S(IJXIT,IJIND),2(B),FS7,TF3,SF1,   QS=>IJIND,B+DOR->DOR532
533                     CGFF,IM4,LB1,RF3,FF9,XF2,BB0,AA1    IF(A),EN JUMP SIGNAL533
534     IJNI    GEN     /S(IJXIT,IJIND),2(B),FS7,TF3,SF1,   QS=>IJIND,DOR->ALU, 534
535                     CGFF,IM4,LB1,FFA,MF1,XF2,BB0        IF(A),EN JUMP SIGNAL535
536             MORG    2   /T(IJPOST,IJIND1)                                  536
537     IJIND1  GEN     /S(IJXIT,IJIND),2(B),FS7,TF3,SF1,   MIL15=>IJIND,MIL->DOR,537
538                     CGFB,IM4,LB1,RF3,FFA,MF1,WF1,XF2,   MIL15->QS,J-SIG,IF(A)538
539                     CBB1                                                    539
540             MORG    2   /S(IJXIT,IJIND)                                   540
541     IJIND   GEN     /T(IJPOST,IJIND1),TF3,GFF,IM1,RF5 WAIT(M),QS=>IJIND1,INCS 541
542             MORG    8   REF BY CND FSEL(6,4,3) WITH BIT 6 ALWAYS = 1.      542
543     IJXIT   GEN     /N(SS3M),SF1,IM4,LB1,RF1,FF9,CF3,   IF(A),DOR+1->P      543
544                     CSH1,BB0                                                544
545     IJPX    GEN     /N(IJXIT),LA1,FF0,CF3,WR1,AA2       P+1->X              545
546     IJXEC   GEN     /N(SS3),SF1,IM8,RF4 (SAME AS SS2)   IF(P),INCP          546
547     IJPB    GEN     /N(IJXIT),LA1,FF0,CF3,WR1,AA1       P+1->B              547
548     ************************   D E C O D E R   ************************** 548
549             ORG     X'FC    DECODER REFERENCES NEXT 4 INSTRUCTIONS        * 549
550     ROF     GEN     /N(SS3MI),SF1,GF4,IM8,RF4           RESET OVFL, IF(P),INCP*550
551     SOF     GEN     /N(SS3MI),SF1,GF2,IM8,RF4           SET OVFL, IF(P), INCP * 551
552     TSA     GMSK    /N(TSA1),1(3),AB2,IME,LB3,FFA,      START IO              * 552
553                     CMKDFC0                                                 * 553
554             ILLOP                                                           * 554
555     ************************************************************************ 555
556             EJEC                                                            556
557     *       BIT TEST INSTRUCTION.                                           557
558     *                                                                       558
559             MORG    16  REF BY FSEL(3-0)                                    559
560     BT20    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFFFE                    560
561     BT21    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFFFD                    561
562     BT22    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFFFB                    562
563     BT23    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFFF7                    563
564     BT24    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFFEF                    564
565     BT25    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFFDF                    565
566     BT26    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFFBF                    566
567     BT27    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFF7F                    567
568     BT28    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFEFF                    568
569     BT29    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFDFF                    569
```

```
570       BT2A    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKFBFF                                  570
571       BT2B    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKF7FF                                  571
572       BT2C    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKEFFF                                  572
573       BT2D    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKDFFF                                  573
574       BT2E    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MKBFFF                                  574
575       BT2F    GMSK    /F(BT30),2(1),FS8,LB3,RF3,FFA,MK7FFF                                  575
576               MORG    2   REF BY FSEL(4)                                                   576
577       BT30    GEN     /F(BT40),2(1),FS9,GF2,LB1,FFB,MF1                                     577
578       BT31    GEN     /F(BT40),2(1),FS9,GF2,LB1,FFB,MF1,AA1                                 578
579               MORG    2   REF BY FSEL(5)                                                   579
580       BT40    GEN     /T(BT50,SS1M),TF3,SF2,GF9,IMC                                         580
581       BT41    GEN     /T(BT50,SS1M),TF2,SF3,GF9,IMC                                         581
582               EJEC                                                                          582
583       *       MULTIPLY INSTRUCTION.                                                         583
584       *                                                                                     584
585               MORG    2   /T(MUL4,MUL1)                                                    585
586       MUL1    GEN     /N(MUL2),GF2,FF9,BBF,AA1          RF+B->ALU,SAMPLE                    586
587       MUL4    GEN     /N(MUL5),LA2,RF3,VF1,AA1          B(L)->DOR,B15->DSB                  587
588               MORG    2   /T(MUL3,MUL5)                                                    588
589       MUL5    GEN     /N(MUL6),FF3,MF1,WR1,AAE          ZERO->RE                            589
590       MUL3    GEN     /N(MUL5),GF8,FF9,BBF,AA1          RF+B->ALU,SAMPLE OVFL               590
591       MUL2    GEN     /T(MUL3,MUL5),TF2,GF9,LA2,RF3,    ALUZ=>MUL3,B(L)->DOR,       FREE    591
592                       CVF1,AA1                          B15->DSB                            592
593       MUL6    GEN     /N(MUL7),MR1,SC1,WF1,XF1          DOR(RT),DOR01->BADO         FREE    593
594               MORG    2                                                                     594
595       MUL8    GEN     /T(MUL9,*),TF2,GFC,MR1,LA3,RF5,   MULTIPLY STEP                       595
596                       CFF9,WR1,SC1,WF1,XF1                                                  596
597       MUL7    GEN     /N(MUL8),MR1,RF5,FF9,WR1,SC1,WF1, FIRST MULTIPLY STEP         FREE    597
598                       CXF1                                                                  598
599               MORG    2   /T(MUL12,MUL13)                                                  599
600       MUL13   GEN     /F(SS3M),2(8),FSF,SF1,IM8,RF4,    FSEL(14),IF(P),INCP               600
601                       CLA3,WR1,SH4,AA1                  SHIFT B                           601
602       MUL12   GEN     /N(MUL13),FF6,CF3,WR1,BBF         A=RF->A                           602
603       MUL9    GEN     /N(MUL11),LA3,WR1,SC1,WF1,XF1     A,DOR(RT),MULS->DLA15             603
604       MUL11   GEN     /T(MUL12,MUL13),TF2,GFA,LB1,FFA,  DSB=>MUL12,DOR->A           FREE    604
605                       CMF1,WR1,AA1                                                          605
606               EJEC                                                                          606
607       *       DOUBLE WORD IMMEDIATE INSTRUCTIONS.  NOTE THAT THESE INSTRUCTIONS             607
608       *       ARE DIFFERENT FROM VARIAN'S SINCE AN M-FIELD EQUAL TO ZERO OR 8               608
609       *       DOES NOT CAUSE UNPREDICTABLE RESULTS.  IN ADDITION, BITS 7, 1               609
610       *       AND 0 ARE USED SIMILARLY TO THE USAGE FOR DOUBLE-WORD EXTENDED              610
611       *       ADDRESSING INSTRUCTIONS.  WHEN BITS 1 AND 0 ARE EQUAL TO 00, 01             611
612       *       OR 10, INDEXING AND INDIRECT ADDRESSING ARE PERFORMED.  THUS,               612
613       *       NEGATIVE OPERANDS ARE NOT POSSIBLE.  WHEN BITS 1 AND 0 ARE EQUAL            613
614       *       TO 11, THE ACTION IS THE SAME AS VARIAN'S ACTION.  NOTE THAT THE            614
615       *       VARIAN ASSEMBLER ASSEMBLES BITS 1 AND 0 AS 00, NOT 11.  THIS              615
616       *       PROGRAM MAY BE MADE TO EXECUTE PROGRAMS SO ASSEMBLED BY                     616
617       *       EXCHANGING IMMREL(X'68) AND IMMNI(X'6B), AND BY EXCHANGING                  617
618       *       IMRL AND IMNI BELOW.                                                        618
619       *                                                                                   619
620               MORG    32  REF BY CND FSEL, FAIL TO IMIND                                620
621       IMMBAS  ILLOP                                                                       621
622       *       NEXT 7 INSTRUCTIONS ALL DO: 0 -> CINTF.                                     622
623       ILDA    GEN     /N(SS2M),LB1,RF4,FFA,MF1,WR1,XF1, INCP, DOR->A                      623
624                       CBB0,AA0                                                            624
625       ILDB    GEN     /N(SS2M),LB1,RF4,FFA,MF1,WR1,XF1, INCP, DOR->B                      625
626                       CBB0,AA0                                                            626
```

```
627       ILDX    GEN      /N(SS2M),LB1,RF4,FFA,MF1,WR1,XF1,  INCP, DOR->X                627
628                       CBB0,AA2                                                        628
629       IINR    GEN      /F(IST2),2(1),FS1,GF8,LB1,RF3,     FSEL(CINTF), DOR+1->DOR      629
630                       CFF9,CF3,XF1,SH1,BB0                SAMPLE OVERFLOW             630
631       ISTA    GEN      /F(IST2),2(1),FS1,RF3,XF1,AA0      FSEL(CINTF),A->DOR           631
632       ISTB    GEN      /F(IST2),2(1),FS1,RF3,XF1,AA1      FSEL(CINTF),B->DOR           632
633       ISTX    GEN      /F(IST2),2(1),FS1,RF3,XF1,AA2      FSEL(CINTF),X->DOR           633
634               ILLOP                                                                   634
635       *       NEXT 7 INSTRUCTIONS ALL DO: 0->CINTF, INCP                              635
636       IORA    GEN      /N(SS2M),LB1,RF4,FFE,MF1,WR1,XF1,  A V DOR -> A                636
637                       CBB0,AA0                                                        637
638       IADD    GEN      /N(SS2M),GF8,LB1,RF4,FF9,WR1,XF1,  A+DOR->A,SAMPLE OVFL        638
639                       CBB0,AA0                                                        639
640       IERA    GEN      /N(SS2M),LB1,RF4,FF6,MF1,WR1,XF1,  A XOR DOR -> A              640
641                       CBB0,AA0                                                        641
642       ISUB    GEN      /N(SS2M),GF8,LB1,RF4,FF6,CF3,WR1,  A-DOR->A,SAMPLE OVFL        642
643                       CXF1,BB0,AA0                                                    643
644       IANA    GEN      /N(SS2M),LB1,RF4,FFB,MF1,WR1,XF1,  A&DOR->A                    644
645                       CBB0,AA0                                                        645
646       IMUL    GEN      /N(MUL),LB1,RF4,FFA,MF1,WR1,XF1,   DOR->RF                     646
647                       CBB0,AAF                                                        647
648       IDIV    GEN      /N(DIV),LB1,RF4,FFA,MF1,WR1,WF1,   DOR->RF, ALU15->OS          648
649                       CXF1,BB0,AAF                                                    649
650               MORG     16  REF BY CND FSEL, FAIL TO IMIND                          650
651       IMXC    EQU      *   IMMEDIATE INSTRUCTION INDEX COMPUTATION                 651
652       *       NEXT 3 INSTRUCTIONS DO C.FSEL(6-3) (QS=0), C.OF(A) (QS=1)               652
653       IMRL    GEN      /S(IMMBAS,IMIND),2(F),FS7,TF3,     P+DOR->DOR                  653
654                       CSF3,GFF,IM5,LA1,LB1,RF3,FF9,BB0                                654
655       IMIX    GEN      /S(IMMBAS,IMIND),2(F),FS7,TF3,     X+DOR->DOR                  655
656                       CSF3,GFF,IM5,LB1,RF3,FF9,BB0,AA2                                656
657       IMIB    GEN      /S(IMMBAS,IMIND),2(F),FS7,TF3,     B+DOR->DOR                  657
658                       CSF3,GFF,IM5,LB1,RF3,FF9,BB0,AA1                                658
659       IMNI    GEN      /F(IMMBAS),2(F),FS7,LB1,RF3,FFA,   MIL->DOR                    659
660                       CMF1,BB1                                                        660
661               MORG     2   REF BY FSEL(7)                                           661
662       IMPRE   GEN      /F(IMXC),2(3),FS4,LB1,RF3,         FSEL(1-0),                  662
663                       CFFA,MF1,WF1,BB1                    MIL->DOR,MIL15->OS          663
664       IMPOST  GEN      /S(IMXC,IMPST1),2(3),FS4,TF3,GFB,  C.FSEL(1-0) (MIL15=0),      664
665                       CLB1,RF3,FFA,MF1,BB1                MIL->DOR                    665
666               MORG     2   /T(IMPSTI,IMIND1)                                         666
667       IMIND1  GEN      /S(IMMBAS,IMIND),2(F),FS7,TF3,     C.FSEL(6-3) (MIL15=0),      667
668                       CSF3,GF3,IM5,LB1,RF3,FFA,MF1,BB1    C.OF(A) (MIL15=1),          668
669       *                                                   MIL->DOR                    669
670       IMPSTI  GEN      /S(IMXC,IMIND),2(3),FS4,TF3,SF3,   C.FSEL(1-0) (MIL15=0),      670
671                       CGFB,IM5,LB1,RF3,FFA,MF1,BB1        C.OF(A) (MIL15=0),          671
672       *                                                   MIL->DOR                    672
673               MORG     2   /S(IMXC,IMIND)                                           673
674       IMIND   GEN      /T(IMPSTI,IMIND1),TF3,GFF,IM1,RF5 WAIT(M),QS=>IMIND1,INCS     674
675       IST4    GEN      /N(SS3M),SF1,IM8,LB1,RF4,FFA,MF1   IF(P),INCP,DOR->ALU    FREE 675
676               MORG     2   /S(IMXC,IMPST1)                                          676
677       IMPST1  GEN      /N(IMIND),SF1,IM5,LB1,FFA,MF1,BB0  OF(ALU),DOR->ALU            677
678       IST3    GEN      /N(IST4),LB1,RF4,FFA,MF1,BB0       INCP,DOR->ALU           FREE 678
679               MORG     2   REF BY FSEL(CINTF)                                       679
680       IST2    GEN      /N(IST3),SF1,IM6,AAF               OS(ALU),RF->ALU             680
681       IST1    GEN      /N(IST3),SF1,IMA                   OS(P)                       681
682               EJEC                                                                    682
683       *       SHIFT INSTRUCTIONS AND INSTRUCTIONS CONSTRAINED TO BE                   683
```

```
684         *       NEAR SHIFT INSTRUCTIONS BY ADDRESSING.  NOTE THAT THESE                 684
685         *       SHIFT INSTRUCTIONS ARE DIFFERENT FROM VARIAN'S SHIFT INSTRUCTIONS        685
686         *       SINCE SETTING THE 'LONG' AND 'A' BITS SIMULTANEOUSLY DOES NOT           686
687         *       PRODUCE UNPREDICTABLE RESULTS.                                          687
688         *                                                                               688
689                 MORG    32  SHIFTS REF BY CND FSEL(8-5), FAIL TO SS4MI               689
690         ASLB    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA2,RF5,WR1,SH4,AA1              690
691         LRLB    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA2,RF5,WR1,SH1,AA1              691
692         ASRB    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA3,RF5,WR1,SH2,AA1              692
693         LSRB    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA3,RF5,WR1,SH4,AA1              693
694         ASLA    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA2,RF5,WR1,SH4                  694
695         LRLA    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA2,RF5,WR1,SH1                  695
696         ASRA    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA3,RF5,WR1,SH2                  696
697         LSRA    GEN     /T(*,SS3MI),TF3,SF3,GFC,IM8,LA3,RF5,WR1,SH4                  697
698         LASL    GEN     /N(LASL1),LA2,RF3,VF1,AA1                                    698
699         LLRL    GEN     /N(LLRL1),RF3,AA1                                            699
700         LASR    GEN     /N(LASR1),LA2,RF3,VF1,AA1                                    700
701         LLSR    GEN     /N(LLSR1),RF3,AA1                                            701
702         *       THE NEXT 4 INSTRUCTIONS ARE IDENTICAL TO THE PREVIOUS 4, BUT         702
703         *       ARE USED WHEN BOTH THE 'LONG' AND 'A' BITS OF THE SHIFT              703
704         *       INSTRUCTION ARE SET.  IT IS POSSIBLE THAT THIS SHOULD BE TREATED     704
705         *       AS AN ERROR CONDITION.                                               705
706         LASL#2  GEN     /N(LASL1),LA2,RF3,VF1,AA1                                    706
707         LLRL#2  GEN     /N(LLRL1),RF3,AA1                                            707
708         LASR#2  GEN     /N(LASR1),LA2,RF3,VF1,AA1                                    708
709         LLSR#2  GEN     /N(LLSR1),RF3,AA1                                            709
710                 MORG    16  REF BY CND FSEL FROM DIV, FSEL FROM MUL,INR              710
711         SS3M    GEN     /N(IWAIT),I(M),GF5,IM6              IBR->I,DECODE, ETC       711
712         SHFT1   GEN     /S(ASLB,SS3MI),2(F),FS9,TF3,SF3,    CND(NOT ALUZ) FSEL(8-5)FREE 712
713                         CGF9,IM8,LB1,RF2,FF6,CF3,SH1,BB0    CND(ALUZ) IF(P),        713
714         *                                                   -DOR->SHIFT COUNTER      714
715                 MORG    2                                                            715
716         LASL1   GEN     /T(LASL2,*),TF2,GFC,LA2,RF5,WR1,    SHFT OVFL => LASL2       716
717                         CSC1,XF3,SH6,AA0                    (A,DOR)LEFT, INC SC      717
718         LASL2   GEN     /N(LASL3),SF1,IM8,SC1,WF1,XF3       IF(P),DOR RT,DSB->DOR15  718
719                 MORG    2   /T(*,LASL3)                                             719
720         LASL3   GEN     /N(IWAIT),I(I),GF5,IM6,LB1,FFA,     DECODE,IBR->I,EN INT,    720
721                         CMF1,WR1,AA1                        DOR->B                   721
722         LLRL1   GEN     /T(*,LASL3),TF3,SF3,GFC,IM8,LA2,    SHFT OVFL=>LASL3,INC SC  722
723                         CRF5,WR1,SC1,XF1,SH2                (A,DOR) LEFT ROTATE      723
724                 MORG    2   /T(LASL2,*)                                             724
725         LASR1   GEN     /T(LASL2,*),TF2,GFC,LA3,RF5,WR1,SC1,WF1,XF1,SH2                725
726         LLSR1   GEN     /T(*,LASL3),TF3,SF3,GFC,IM8,LA3,RF5,WR1,SC1,                   726
727                         CWF1,XF1,SH4                                                   727
728                 ORG     SS3M+8   REF BY FSEL, MUST BE MORG 32 WITH SHIFTS              728
729         SS3MI   GEN     /N(IWAIT),I(I),GF5,IM6              IBR->I,DECODE, ETC       729
730                 MORG    2   /S(SS3M,DIV15)                                           730
731         DIV15   GEN     /F(SS3M),2(8),FSF,SF1,IM8,FF6,      FSEL(14),IF(P),          731
732                         CCF3,WR1,SH1,BB1,AA1                -B->B                    732
733                 EJEC                                                               733
734         *       EXTERNAL CONTROL INSTRUCTION.                                      734
735         *                                                                          735
736         ************************   D E C O D E R   ***************************** 736
737                 ORG     X'180    DECODER REFERENCES NEXT 3 INSTRUCTIONS        *  737
738         EXC1    GMSK    /N(EXC2),LB3,RF3,FFE,MKF7FF,16(DR)  X'800 V DR => OPR  *  738
739         SWA40   GMSK    /F(STA2-2),2(6),FSF,SF1,IM6,LB2,    FSEL(13-12),OS(ALU),*  739
740                         CFFA,MKF800                         I&X'7FF->ALU         *  740
```

```
741       SWA10  GMSK    /F(LDA1),2(C),MT1,FSF,SF1,IM5,    FSEL(15-13),OS(ALU),  *      741
742                      CLB2,RF3,FFA,MKF800               I&X'7FF->DOR          *      742
743       ********************************************************************          743
744       EXC2   GEN     /N(EXC3),1(1),IMF,LB1,FFA,MF1,    START IO,DOR->S1        FREE 744
745                      CWR1,BB0,24(S1)                                                745
746       EXC3   GMSK    /N(SS2),IM2,LB2,FFE,MKFE00,16(S1) WAIT(IO),I&0'777 V S1   FREE 746
747       EXC22  GMSK    /N(SS2),IM2,LB2,FFE,MK7E00,   I&X'81FF V DR, WAIT(IO)     FREE 747
748                      C16(DR)                                                        748
749              EJEC                                                                   749
750       *      SET I/O DEVICE SELECTION REGISTER.  THIS IS NOT A VARIAN               750
751       *      INSTRUCTION.  THIS INSTRUCTION PLACES 6 BITS IN REGISTER DR,           751
752       *      WHICH ARE SUBSEQUENTLY INCLUSIVE-OR'ED WITH ALL DEVICE SELECTION       752
753       *      SPECIFICATIONS SENT TO THE I/O BUS.  THE OCTAL FORM FOR THIS           753
754       *      INSTRUCTION IS:                                                        754
755       *      107TDD   WHERE                                                         755
756       *                T=0:  DD->DR          T=4:  X->DR                            756
757       *                T=1:  A->DR           T=5:  A V X -> DR                      757
758       *                T=2:  B->DR           T=6:  B V X -> DR                      758
759       *                T=3:  A V B -> DR     T=7:  A V B V X -> DR                  759
760       *                                                                             760
761       SETXIT GEN     /N(IWAIT),1(1),GF5,IM6,LB1,FFA,   DECODE,IBR->I,EN INT    FREE 761
762                      CMF1,WR1,BB0,24(DR)               DOR->DR                      762
763              MORG    8   REFERENCED BY FSEL(8-6)                                    763
764       SETI   GMSK    /N(SETXIT),SF1,IM8,LB2,RF7,FFA,   I&0'77->DOR                  764
765                      CMKFFC0                           IF(P),INCP                   765
766       SETA   GMSK    /N(SETXIT),SF1,IM8,LB3,RF7,FF7,   DR&0'77->DOR                 766
767                      CMK3F,16(DR)                      IF(P),INCP                   767
768       SETB   GMSK    /N(SETXIT),SF1,IM8,LB3,RF7,FF7,   B&0'77->DOR                  768
769                      CMK3F,AK1                         IF(P),INCP                   769
770       SETAB  GEN     /N(SETA),FFE,MF1,WR1,BB1,24(DR)   B V DR -> DR                 770
771       SETX   GMSK    /N(SETXIT),SF1,IM8,LB3,RF7,FF7,   X&0'77->DOR                  771
772                      CMK3F,AK2                         IF(P),INCP                   772
773       SETAX  GEN     /N(SETA),FFE,MF1,WR1,BB2,24(DR)   X V DR -> DR                 773
774       SETBX  GEN     /N(SETAX),FFA,MF1,WR1,BB1,24(DR)  B->DR                        774
775       SETABX GEN     /N(SETAX),FFE,MF1,WR1,BB1,24(DR)  B V DR -> DR                 775
776              EJEC                                                                   776
777       *      SENSE INSTRUCTION.                                                     777
778       *                                                                             778
779       ************************   D E C O D E R   ***************************       779
780              ORG     X'190    DECODER REFERENCES NEXT 3 INSTRUCTIONS          *     780
781       SEN1   GMSK    /N(SEN2),LB3,RF3,FFE,MKEFFF,16(DR)  X'1000 V DR -> DOR  *     781
782       SWA41  GMSK    /F(STA2-2),2(6),FSF,SF1,IM6,LB2,  FSEL(13-12),OS(ALU),  *     782
783                      CFFA,MKF800                       I&X'7FF->ALU           *     783
784       SWA11  GMSK    /F(LDA1),2(C),MT1,FSF,SF1,IM5,    FSEL(15-13),OS(ALU),  *     784
785                      CLB2,RF3,FFA,MKF800               I&X'7FF->DOR           *     785
786       ********************************************************************          786
787       SEN2   GEN     /N(SEN3),1(1),IMF,LB1,FFA,MF1,    START I/O AT X'4,       FREE 787
788                      CWR1,RF1,BB0,24(S1)               DOR->S1,RESET CINTF          788
789       SEN3   GMSK    /N(SEN4),IM2,LB2,FFE,MKFE00,      WAIT(IO),               FREE 789
790                      C16(S1)                           I&X'1FF V S1 ->ALU       FREE 790
791       SEN4   GEN     /T(SEN5,SS1M),TF2,SF3,GF1,IMC,RF4 I/O SENSE=>SEN5,INCP,   FREE 791
792       *                                                IF(MIL) IF I/O SENSE         792
793              MORG    2   /T(SEN6,SEN7)                                              793
794       SEN7   GEN     /N(SEN5),SF1,IMC,RF5              IF(MIL),INCSC                794
795       SEN6   GEN     /N(SS3M),SF1,IM8                  IF(P)                        795
796              EJEC                                                                   796
797       *      INSTRUCTIONS FORCED TOGETHER BY TEST ADDRESSING                        797
```

```
         *
                MORG     4   /T(JMP1,SS1M), /T(BT50,SS1M), /T(SEN5,SS1M)
         SS1M   GEN      /N(SS2M),SF1,IM8                    IF(P)
         JMP1   GEN      /T(JMP2,SS2M),TF2,GFB,IM5,LB1,      MILS=>JMP2,OF(ALU),
                         CRF1,FFA,MF1,XF2,BB1                MIL->P,JUMP SIGNAL
         BT50   GEN      /T(BT51,BT52),TF2,GFB,LB1,RF1,      MILS=>BT51,MIL+1->P,
                         CFF9,CF3,XF2,SH1,BB1                JUMP SIGNAL
         SEN5   GEN      /T(SEN6,SEN7),TF3,GFB,LB1,RF1,      MILS=>SEN7,MIL+1->P
                         CFF9,CF3,SH1,BB1
                MORG     4   /T(JMP2,SS2M), /T(SRE80,SS2M)
         SS2M   GEN      /N(SS3M),SF1,IM8,RF4,FF0,24(*-*)    IF(P),INCP,(AA)->ALU
         JMP2   GEN      /N(JMP1),SF1,IMC,RF5                IF(MIL),INC SC
         SRE80  GEN      /N(SRE81),LB1,RF1,FF9,CF3,SH1       DOR+1->P
                EJEC
         *      INPUT DATA TRANSFER INSTRUCTIONS.
         *
         *************************    D E C O D E R    *************************
                ORG      X'1A0    DECODER REFERENCES NEXT 3 INSTRUCTIONS            *
         IABM1  GMSK     /N(IABM2),LB3,RF3,FFE,MKDFFF,16(DR)  X'2000 V DR => OPR    *
         SWA42  GMSK     /F(STA2-2),2(6),FSF,SF1,IM6,LB2,    FSEL(13-12),OS(ALU),   *
                         CFFA,MKF800                         I&X'7FF->ALU           *
         SWA12  GMSK     /F(LDA1),2(C),MT1,FSF,SF1,IM5,      FSEL(15-13),OF(ALU),   *
                         CLB2,RF3,FFA,MKF800                 I&X'7FF->DOR           *
         *************************************************************************
         IABM2  GEN      /N(IABM3),1(3),IMF,LB1,FFA,MF1,     START I/O AT X'C,      FREE
                         CWR1,BB0,24(S1)                     DOR=>S1
         IABM3  GMSK     /F(IME1),2(7),FSA,IM2,LB2,FFE,      FSEL(8-6),WAIT(IO),    FREE
                         CMKFE00,16(S1)                      IR & X'1FF V S1
         INAB2  GEN      /N(IAB),SF1,IM8,LB1,RF7,FFE,MF1,    IF(P),INCP,            FREE
                         CWR1,BB0,AA0                        DOR V A => A,DOR
         IAB    GEN      /N(IWAIT),1(M),GF5,IM6,LB1,FFA,     IBR->I,DEC,INTRPT,     FREE
                         CMF1,WR1,BB0,AA1                    DOR=>B
         IME2   GEN      /N(SS2M),SF1,IM8,RF4,LB1,FFA,MF1    IF(P),INCP,DOR->ALU
                MORG     8   REF BY FSEL(8-6)
         IME1   GEN      /N(IME2),SF1,IME,LB1,RF3,FFA,MF1,   OS(MIL),IOR->DOR,
                         CXF1,BB2                            RESET CINTF
         INA    GEN      /N(SS3M),SF1,IM8,LB1,RF4,FFE,MF1,   IF(P),INCP,IOR V A => A
                         CWR1,BB2,AA0
         INB    GEN      /N(SS3M),SF1,IM8,LB1,RF4,FFE,MF1,   IF(P),INCP,IOR V B => B
                         CWR1,BB2,AA1
         INAB1  GEN      /N(INAB2),LB1,RF3,FFE,MF1,BB2,AA1   IOR V B => DOR
                ILLOP
         CIA    GEN      /N(SS3M),SF1,IM8,LB1,RF4,FFA,MF1,   IF(P),INCP,IOR->A
                         CWR1,BB2,AA0
         CIB    GEN      /N(SS3M),SF1,IM8,LB1,RF4,FFA,MF1,   IF(P),INCP,IOR->B
                         CWR1,BB2,AA1
         CIAB   GEN      /N(IAB),SF1,IM8,LB1,RF7,FFA,MF1,    IF(P),INCP,IOR->A,DOR
                         CWR1,BB2,AA0
                EJEC
         *      OUTPUT DATA TRANSFER INSTRUCTIONS.
         *
         *************************    D E C O D E R    *************************
                ORG      X'1B0    DECODER REFERENCES NEXT 3 INSTRUCTIONS            *
         OABM1  GMSK     /F(OME1),2(3),FSA,LB3,RF3,FFE,      FSEL(7-6),             *
                         CMKBFFF,16(DR)                      X'2000 V DR => DOR     *
         SWA43  GMSK     /F(STA2-2),2(6),FSF,SF1,IM6,LB2,    FSEL(13-12),OS(ALU),   *
                         CFFA,MKF800                         I&X'7FF->ALU           *
```

```
855      SWA13  GMSK    /F(LDA1),2(C),MT1,FSF,SF1,IM5,    FSEL(15-13),OF(ALU),  *      855
856                     CLB2,RF3,FFA,MKF800               I&X'7FF->DOR          *      856
857      ********************************************************************     857
858      OME2   GEN     /N(OAR1),IM1                      WAIT(M)                FREE 858
859             MORG    4   REF BY FSEL                                            859
860      OME1   GEN     /N(OME2),SF1,IMD,XF1              OF(MIL),RESET CINTF          860
861      OAR1   GEN     /N(OABM2),1(7),IMF,LB1,FFA,MF1,   START I/O AT X'1C,          861
862                     CWR1,BBO,24(S1)                   DOR->S1                     862
863      OBR1   GEN     /N(OABM2),1(7),IMF,LB1,FFA,MF1,   START I/O AT X'1C,          863
864                     CWR1,BBO,24(S1)                   DOR->S1                     864
865      OAB1   GEN     /N(OABM2),1(7),IMF,LB1,FFA,MF1,   START I/O AT X'1C,          865
866                     CWR1,BBO,24(S1)                   DOR->S1                     866
867             MORG    4   REF BY FSEL                                            867
868      OME3   GEN     /N(SS1M),IM2,LB1,RF4,FFA,MF1,BB1  WAIT(IO),INCP,MIL->ALU      868
869      OAR2   GEN     /N(SS2M),IM2,AA0                  WAIT(IO),A->ALU             869
870      OBR2   GEN     /N(SS2M),IM2,AA1                  WAIT(IO),B->ALU             870
871      OAB2   GEN     /N(SS2M),IM2,FFE,MF1,BB1,AA0      WAIT(IO),A V B->ALU         871
872      OABM2  GMSK    /F(OME3),2(3),FSA,IM2,LB2,FFE,    FSEL(7-6),WAIT(IO),   FREE 872
873                     CMKFE00,16(S1)                    I&X'1FF V S1 -> ALU         873
874             EJEC                                                                874
875      *      AUTOMATIC BOOTSTRAP LOADER FOR TELETYPE AND HIGH-SPEED PAPER TAPE    875
876      *      DEPOSITED AT 0200, STARTED WITH X=07000 AT 0212.                     876
877      *                                                                           877
878      ABL20  GMSK    /N(ABL3),SF1,IMA,LB3,RF4,FFA,MK7A7E    0200 CIB  001      FREE 878
879      ABL21  GMSK    /N(ABL3),SF1,IMA,LB3,RF4,FFA,MK7A60    0200 CIB  037      FREE 879
880      ABL3   GMSK    /N(ABL4),SF1,IMA,LB3,RF4,FFA,MKF7F6    0201 ASLB 9        FREE 880
881      ************************   D E C O D E R   ***************************     881
882             ORG     X'1C0   DECODER REFERENCES NEXT 3 INSTRUCTIONS        *      882
883      EXC21  GEN     /N(EXC22),1(1),IME                START I/O AT 4       *      883
884      SWA44  GMSK    /F(STA2-2),2(6),FSF,SF1,IM6,LB2,  FSEL(13-12),OS(ALU),  *      884
885                     CLA1,FF9,MKFE00                   I&X'1FF + P => ALU   *      885
886      SWA14  GMSK    /F(LDA1),2(C),MT1,FSF,SF1,IM5,    FSEL(15-13),OF(ALU),  *      886
887                     CLB2,LA1,RF3,FF9,MKFE00           I&X'1FF + P => DOR   *      887
888      ********************************************************************     888
889      ABL4   GMSK    /N(ABL5),SF1,IMA,LB3,RF4,FFA,MKF7DE    0202 LRLB 1        FREE 889
890      ABL5   GMSK    /N(ABL6),SF1,IMA,LB3,RF4,FFA,MKF6D9    0203 LLRL 6        FREE 890
891      ABL6   GMSK    /N(ABL7),SF1,IMA,LB3,RF4,FFA,MKFDEF    0204 JBZ           FREE 891
892      ABL7   GMSK    /N(ABL8),SF1,IMA,LB3,RF4,FFA,MKFF73    0205 0214          FREE 892
893      ABL8   GMSK    /N(ABL9),SF1,IMA,LB3,RF4,FFA,MKA5FF    0206 STA  0,X      FREE 893
894      ABL9   GMSK    /N(ABLA),SF1,IMA,LB3,RF4,FFA,MKFDF7    0207 JAZ           FREE 894
895      ABLA   GMSK    /N(ABLB),SF1,IMA,LB3,RF4,FFA,MKF1FF    0210 07000->DOR    FREE 895
896      ABLB   GMSK    /N(ABLC),SF1,IMA,LB3,RF4,FFA,MKF59B    0211 IXR           FREE 896
897      ABLC   GMSK    /T(ABLD1,ABLD0),TF2,SF1,IMA,LB3,RF4,   0212 INCR 1        FREE 897
898                     CFFA,MKF5BE                                                 898
899             MORG    2   /T(ABLD1,ABLD0)                                        899
900      ABLD0  GMSK    /N(ABLE0),SF1,IMA,LB3,RF4,FFA,MK7A7E    0213 CIB  001              900
901      ABLD1  GMSK    /N(ABLE1),SF1,IMA,LB3,RF4,FFA,MK7EA0    0213 EXC  037              901
902      ABLE0  GMSK    /N(ABLF),SF1,IMA,LB3,RF4,FFA,MK7D7E     0214 SEN  001      FREE 902
903      ABLE1  GMSK    /N(ABLF),SF1,IMA,LB3,RF4,FFA,MK7CA0     0214 SEN  037      FREE 903
904      ************************   D E C O D E R   ***************************     904
905             ORG     X'1D0   DECODER REFERENCES NEXT 3 INSTRUCTIONS        *      905
906      *      NOTE - BCS WILL BE DETECTED BY WCS ON MACHINES SO EQUIPPED,    *      906
907      *      RATHER THAN CAUSING ILLEGAL INSTRUCTION PROCESSING.            *      907
908      BCS    ILLOP                                                          *      908
909      SWA45  GMSK    /F(STA2-2),2(6),FSF,SF1,IM6,LB2,  FSEL(13-12),OS(ALU),  *      909
910                     CFF9,MKFE00,AK2                   I&X'1FF + X => ALU   *      910
911      SWA15  GMSK    /F(LDA1),2(C),MT1,FSF,SF1,IM5,    FSEL(15-13),OF(ALU),  *      911
```

```
912                 CLB2,RF3,FF9,MKFE00,AK2          I&X'1FF + X => DOR    *      912
913     ********************************************************************      913
914     ABLF   GMSK   /N(ABLX0),SF1,IMA,LB3,RF4,FFA,MKFF7F   0215 0200        FREE 914
915     ABLX0  GMSK   /N(ABLX1),SF1,IMA,LB3,RF4,FFA,MKFDFF   0216 JMP         FREE 915
916     ABLX1  GMSK   /N(ABLX2),IM1,LB3,FFA,MKFF73           0217 0214        FREE 916
917     ABLX2  GEN    /N(ABLX3),LB1,FFA,MF1,WR1,AA2          DOR=>X           FREE 917
918     ABLX3  GMSK   /N(SS2),SF1,GF4,IM4,LB3,RF1,FFA,MKFF75 0212=>P           FREE 918
919     *                                                                       919
920     *     AUTOMATIC BOOTSTRAP LOADER FOR DISC.                              920
921     *     DEPOSITED AT 01130, STARTED AT 01130.                             921
922     *                                                                       922
923     ABL31  GMSK   /N(ABL32),SF1,IMA,LB3,RF4,FFA,MK7EF1   01130 EXC  0416   FREE 923
924     ABL32  GMSK   /N(ABL33),SF1,IMA,LB3,RF4,FFA,MK77F1   01131 EXC, 0016   FREE 924
925     ABL33  GMSK   /N(ABL34),SF1,IMA,LB3,RF4,FFA,MK7F71   01132 EXC  0216   FREE 925
926     ABL34  GMSK   /N(ABL35),SF1,IMA,LB3,RF4,FFA,MKF5FE   01133 TZA         FREE 926
927     ABL35  GMSK   /N(ABL36),SF1,IMA,LB3,RF4,FFA,MK79B1   01134 OAR  0016   FREE 927
928     ABL36  GMSK   /N(ABL37),SF1,IMA,LB3,RF4,FFA,MK7DF1   01135 SEN  0016   FREE 928
929     ABL37  GMSK   /N(ABL38),SF1,IMA,LB3,RF4,FFA,MKFD9E   01136 01141       FREE 929
930     ABL38  GMSK   /N(ABL39),SF1,IMA,LB3,RF4,FFA,MKFDFF   01137 JMP         FREE 930
931     ************************   D E C O D E R   ***************************      931
932             ORG    X'1E0    DECODER REFERENCES NEXT 3 INSTRUCTIONS        *      932
933             ILLOP                                                         *      933
934     SWA46  GMSK   /F(STA2=2),2(6),FSF,SF1,IM6,LB2,  FSEL(13-12),OS(ALU),  *      934
935                 CFF9,MKFE00,AK1                    I&X'1FF + B => ALU    *      935
936     SWA16  GMSK   /F(LDA1),2(C),MT1,FSF,SF1,IM5,    FSEL(15-13),OF(ALU),  *      936
937                 CLB2,RF3,FF9,MKFE00,AK1            I&X'1FF + B => DOR    *      937
938     ********************************************************************      938
939     ABL39  GMSK   /N(ABL3A),SF1,IMA,LB3,RF4,FFA,MKFDA2   01140 01133       FREE 939
940     ABL3A  GMSK   /N(ABL3B),SF1,IMA,LB3,RF4,FFA,MK7AB1   01141 CIA  0016   FREE 940
941     ABL3B  GMSK   /N(ABL3C),SF1,IMA,LB3,RF4,FFA,MK2D89   01142 ANA  01166  FREE 941
942     ABL3C  GMSK   /N(ABL3D),SF1,IMA,LB3,RF4,FFA,MKFDF1   01143 JANZ        FREE 942
943     ABL3D  GMSK   /N(ABL3E),SF1,IMA,LB3,RE4,FFA,MKFDA7   01144 01130       FREE 943
944     ABL3E  GMSK   /N(ABL3F),SF1,IMA,LB3,RF4,FFA,MK7FEE   01145 EXC  0021   FREE 944
945     ABL3F  GMSK   /N(ABL40),SF1,IMA,LB3,RF4,FFA,MK7F31   01146 EXC  0316   FREE 945
946     ABL40  GMSK   /N(ABL41),SF1,IMA,LB3,RF4,FFA,MKF5BD   01147 INCR 2       FREE 946
947     ABL41  GMSK   /N(ABL42),SF1,IMA,LB3,RF4,FFA,MK7971   01150 OBR  0016   FREE 947
948     ABL42  GMSK   /N(ABL43),SF1,IMA,LB3,RF4,FFA,MK79AF   01151 OAR  0020   FREE 948
949     ABL43  GMSK   /N(ABL44),SF1,IMA,LB3,RF4,FFA,MKF9EE   01152 OME  0021   FREE 949
950     ABL44  GMSK   /N(ABL45),SF1,IMA,LB3,RF4,FFA,MKFD9B   01153 01144       FREE 950
951     ABL45  GMSK   /N(ABL46),SF1,IMA,LB3,RF4,FFA,MK7FEF   01154 EXC  0020   FREE 951
952     ************************   D E C O D E R   ***************************      952
953             ORG    X'1F0    DECODER REFERENCES NEXT 3 INSTRUCTIONS        *      953
954     SETDR  GEN    /F(SETI),2(7),FSA,FFA,MF1,WR1,    FSEL(8-6), A=>DR    *      954
955                 CBB0,24(DR)                                               *      955
956     SWA47  GMSK   /N(SWA50),SF1,IM5,LB2,FFA,MKFE00  OF(ALU),I&X'1FF=>ALU  *      956
957     SWA17  GMSK   /N(SWA21),SF1,IM5,LB2,FFA,MKFE00  OF(ALU),I&X'1FF=>ALU  *      957
958     ********************************************************************      958
959     ABL46  GMSK   /N(ABL47),SF1,IMA,LB3,RF4,FFA,MK7FF1   01155 EXC  0016   FREE 959
960     ABL47  GMSK   /N(ABL48),SF1,IMA,LB3,RF4,FFA,MK7CF1   01156 SEN  0416   FREE 960
961     ABL48  GMSK   /N(ABL49),SF1,IMA,LB3,RF4,FFA,MKFD91   01157 01156       FREE 961
962     ABL49  GMSK   /N(ABL4A),SF1,IMA,LB3,RF4,FFA,MK7AB1   01160 CIA  0016   FREE 962
963     ABL4A  GMSK   /N(ABL4B),SF1,IMA,LB3,RF4,FFA,MK2D89   01162 ANA  01166  FREE 963
964     ABL4B  GMSK   /N(ABL4C),SF1,IMA,LB3,RF4,FFA,MKFDF1   01162 JANZ        FREE 964
965     ABL4C  GMSK   /N(ABL4D),SF1,IMA,LB3,RF4,FFA,MKFDA7   01163 01130       FREE 965
966     ABL4D  GMSK   /N(ABL4E),SF1,IMA,LB3,RF4,FFA,MKFDFF   01164 JMP         FREE 966
967     ABL4E  GMSK   /N(ABL4F),SF1,IMA,LB3,RF4,FFA,MKFE7F   01165 00600       FREE 967
968     ABL4F  GMSK   /N(ABL50),IM1,LB3,FFA,MKF00F           01166 07760       FREE 968
```

```
969       ABL50  GMSK    /N(SS2),SF1,IM4,LB3,RF1,FFA,MKFDA7     01130->P          FREE 969
970       ********************************   R E S E T   ********************************     970
971              ORG     X'1FE    REF BY FSEL(CINTF)                                  *     971
972       HLT8   GEN     /N(RESET),RF4,XF1                  RESET CINTF, INCP        *     972
973              ORG     X'1FF    HARDWARE INTERRUPTS HERE FOR STEP AND RESET         *     973
974       RESET  GEN     /N(HLT7),1(4),GF4,IMC,LB2,RF3,FFA INTRPT,I->DOR             *     974
975       ********************************************************************************     975
976              END                                                                          976

CPU:.240   CTP:.024   SUPS:4.818

@HDG,P                                        SOURCE DECK: 50

     620-DIFF/DOC

@PRT,S          PFPFILE.620-DIFF/DOC
FURPUR 27-S  08/16/78  10:14:25
```