DIGITAL SCIENTIFIC CORPORATION
META 4™ COMPUTER SYSTEM

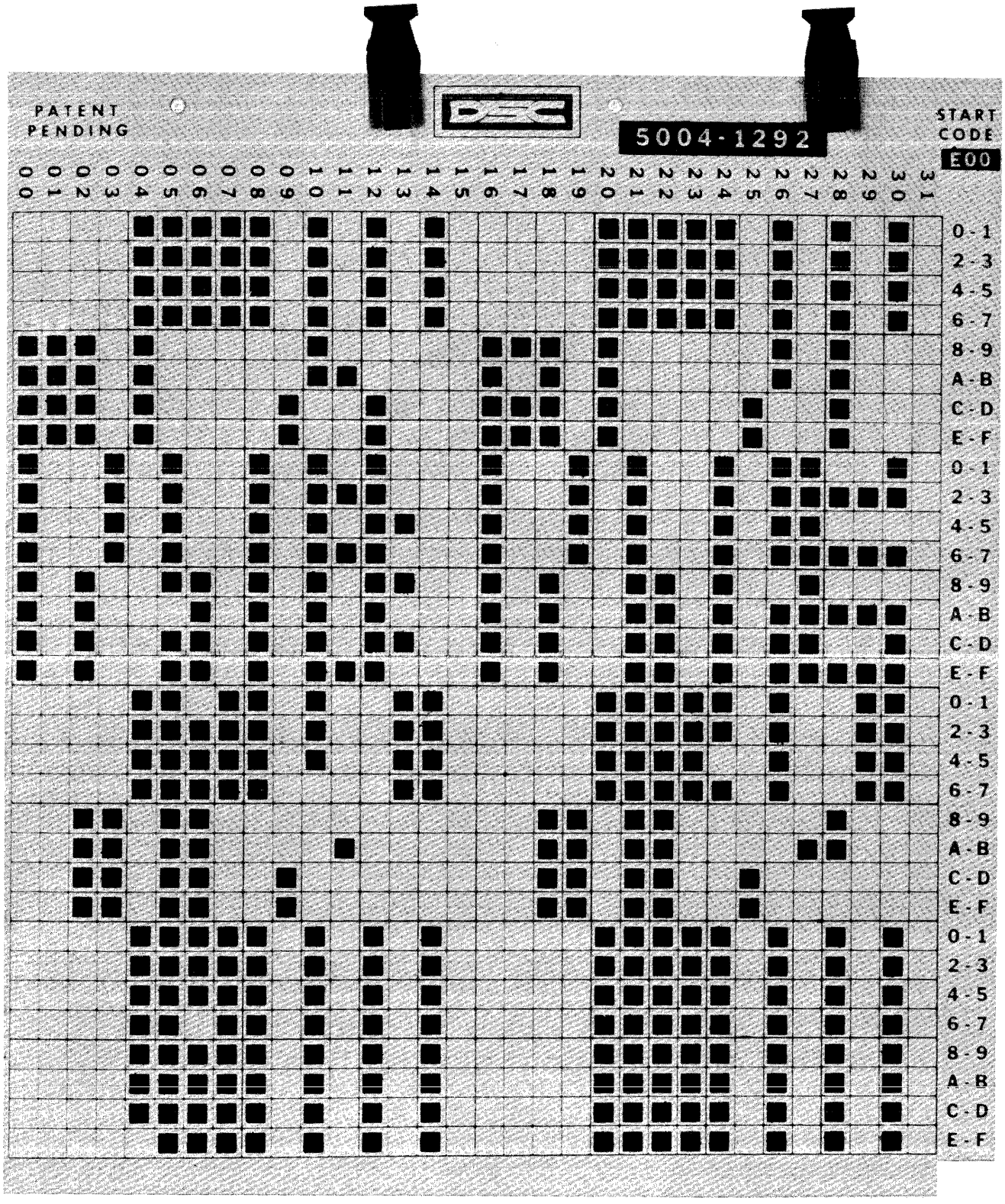
TYPICAL ROM PATTERN LISTING
AND
PROGRAM TO SIMULATE
THE IBM 1130 INSTRUCTION SET

Publication No. M4/005P-170

January 1970

DIGITAL SCIENTIFIC CORPORATION
11455 Sorrento Valley Road
San Diego, California 92121

PRINTED IN U.S.A



DIGITAL SCIENTIFIC META 4 COMPUTER SYSTEM ROM BOARD,

TYPICAL PATTERN

D.S.C. META 4 ROM PATTERN STARTING LOCATION 000 (HEX)

	0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	2 0	2 1	2 2	2 3	2 4	2 5	2 6	2 7	2 8	2 9	3 0	3 1		
000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	*	0	0	0	*	0	0	0	0	0	0	0-1	00000C40
002	*	0	*	0	*	0	0	*	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	2-3	A9640001
004	0	0	*	0	0	*	0	0	0	*	0	*	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4-5	24550000
006	0	0	*	0	*	0	0	*	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6-7	29640000
008	0	0	*	0	0	*	0	0	0	*	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8-9	24540000
00A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	A-B	0000022E
00C	*	0	*	0	*	0	0	0	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	C-D	A8640001
00E	0	0	*	0	*	0	*	0	0	*	0	*	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	E-F	2A550000
010	0	0	*	0	*	0	0	0	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0-1	28640000
012	0	0	*	0	*	0	*	0	0	*	0	0	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2-3	2A450000
014	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	4-5	0000022E
016	0	0	*	0	*	0	*	0	0	*	*	*	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6-7	2A750000
018	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	8-9	0000022E
01A	0	0	*	0	*	0	*	0	*	0	*	0	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A-B	2AA50000
01C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	C-D	0000022E
01E	0	0	*	0	*	0	*	0	*	0	*	*	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	E-F	2AB50000
020	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	0-1	0000022E
022	0	0	*	0	*	0	*	0	*	*	0	0	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2-3	2AC50000
024	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	4-5	0000022E
026	0	0	*	0	*	*	0	0	0	*	*	0	0	*	*	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6-7	2C670000
028	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	8-9	0000022E
02A	0	0	*	0	*	*	0	0	0	*	*	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A-B	2C6A0000
02C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	C-D	0000022E
02E	0	0	*	0	*	*	0	0	0	*	*	0	*	0	*	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	E-F	2C6B0000
030	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	0-1	0000022E
032	0	0	*	0	*	*	0	0	0	*	*	0	*	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2-3	2C6C0000
034	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	4-5	0000022E
036	*	0	0	*	0	0	0	0	0	*	0	*	0	0	*	*	*	*	*	*	*	*	*	*	0	0	0	0	0	0	0	0	6-7	9053FF00
038	*	0	0	*	*	*	0	0	*	0	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	*	8-9	9C880003
03A	0	0	*	0	*	0	*	0	*	0	0	0	0	*	0	*	0	0	*	*	0	0	0	0	0	0	0	0	0	0	0	0	A-B	2A853000
03C	*	0	*	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C-D	A4080000
03E	0	0	0	0	0	0	0	0	~	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	E-F	0000022E

D.S.C. META 4 ROM PATTERN STARTING LOCATION 040 (HEX)

	0 0	0 1	0 2	0 3	0 4	0 5	0 6	0 7	0 8	0 9	1 0	1 1	1 2	1 3	1 4	1 5	1 6	1 7	1 8	1 9	2 0	2 1	2 2	2 3	2 4	2 5	2 6	2 7	2 8	2 9	3 0	3 1		
040	0	*	0	0	0	*	0	0	0	*	0	*	0	*	0	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	0	0	0-1	44544080
042	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	*	0	0	0	2-3	00000048
044	*	0	*	*	0	*	0	0	0	*	0	*	0	*	*	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	4-5	B456FFFF
046	0	*	0	0	*	*	0	0	0	*	*	0	0	*	0	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	0	*	6-7	4C644081
048	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	0	0	0	0	*	0	0	0	0	0	*	0	0	*	*	0	0	8-9	0010104C
04A	*	0	*	0	*	*	0	0	*	0	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	A-B	AC880001
04C	0	0	0	0	0	0	0	0	0	0	0	*	*	0	0	0	0	0	0	0	0	0	0	0	0	*	0	*	0	0	*	0	C-D	00180052
04E	*	0	0	*	*	*	0	0	*	0	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	E-F	9C880001
050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	0-1	0000022E
052	*	0	*	0	*	*	0	0	*	0	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	2-3	AC880002
054	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	4-5	0000022E
056	0	0	0	*	0	*	0	0	0	*	0	*	0	*	0	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	0	0	6-7	14544080
058	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	8-9	0000022E
05A	0	0	*	0	0	*	0	0	0	*	0	*	0	*	0	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	0	0	A-B	24544080
05C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	C-D	0000022E
05E	0	0	*	*	0	*	0	0	0	*	0	*	0	*	0	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	0	0	E-F	34544080
060	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	*	0	*	*	*	0	0-1	0000022E
062	*	0	*	0	*	0	0	*	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	2-3	A9640001
064	0	*	0	0	0	*	0	0	0	*	0	*	0	*	0	*	0	*	0	*	0	0	0	0	*	0	0	0	0	0	0	0	4-5	44555080
066	0	0	*	0	*	0	0	*	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6-7	29640000
068	0	*	0	0	0	*	0	0	0	*	0	*	0	*	0	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	*	0	8-9	44544082
06A	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	*	0	0	0	A-B	00000048
06C	*	0	*	0	*	0	0	*	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	C-D	A9640001
06E	*	0	*	*	0	0	0	0	0	*	0	*	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	E-F	B053FFFF
070	0	*	0	0	0	*	0	0	0	0	*	*	0	*	0	*	0	*	0	*	0	0	0	0	*	0	0	0	0	0	0	*	0-1	44355081
072	0	0	*	0	*	0	0	*	0	*	*	0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2-3	29640000
074	*	0	*	*	0	0	0	0	0	*	0	*	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	4-5	B053FFFF
076	0	*	0	0	0	*	0	0	0	0	*	*	0	*	0	0	0	*	0	0	0	0	0	0	*	0	0	0	0	0	*	0	6-7	44344082
078	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	*	0	0	0	8-9	00000048
07A	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0	*	0	0	0	0	0	0	0	0	0	0	*	0	0	0	0	0	A-B	A0010010
07C	0	0	0	0	*	0	0	0	0	*	0	0	*	0	0	0	0	0	0	0	0	0	0	*	0	0	*	0	0	*	*	0	C-D	08480126
07E	0	0	*	0	*	*	0	0	0	*	0	0	0	*	0	*	0	0	0	0	0	0	*	0	0	0	0	*	0	0	0	0	E-F	2C450210

```
                *PROGRAM TO SIMULATE THE IBM 1130                                  00010
                *INSTRUCTION SET                                                   00020
                *                                                                  00030
0000            0     EQUR          0                 ADDRESS ZERO               00040
0001            C     EQUR          1$                COUNTER REGISTER           00050
0002            L     EQUR          2$                LINK REGISTER              00060
                *BITS 0,1&2 CONTAIN-READ ONLY-CARRY,OVFL                           00070
                *AND SHIFT CONDITION BITS                                          00080
0003            S     EQUR          3                 SCRATCH ACCUMULATOR        00090
0004            M     EQUR          4                 MEMORY ADDRESS REG         00100
0005            D     EQUR          5                 MEMORY DATA REGISTER       00110
0006            Y     EQUR          6                 IOCC OUT,INTERRUPT IN      00120
0007            Z     EQUR          7                 I/O DATA IN AND OUT        00130
001E            H     EQUR          1E$               CHARACTERISTIC REG         00140
001F            P     EQUR          1F                PRIORITY REG               00150
0015            Q     EQUR          15                ACCUMULATOR EXTENSION      00160
0016            U     EQUR          16                TEMP ACCUMULATOR           00170
0017            I     EQUR          17                INSTRUCTION ADDR REG       00180
0018            X     EQUR          18                STATUS REG                 00190
0019            O     EQUR          19                OPERAND REGISTER           00200
001A            1     EQUR          1A                1NDEX 1                    00210
001B            2     EQUR          1B                INDEX 2                    00220
001C            3     EQUR          1C                INDEX 3                    00230
001D            K     EQUR          1D                PRIORIYY MASK REG          00240
0014            A     EQUR          14                ACCUMULATOR                00250
```

```
                    *                                                                    00280
0000                0$   ORG                                                         00290
                    *THE AREA BETWEEN 000 AND OFF CAN BE                              00300
                    *ADDRESSED BY THE LEAST SIGNIFICANT 8                             00310
                    *BITS OF A BRANCH OR LOAD THIS AREA OF                            00320
                    *ROM IS DESIGNATED P2,AND CONTAINS THE                            00330
                    *OPERATION ROUTINES FOR MOST INSTRUCTIONS                         00340
                    *                                                                 00350
0000 00000C40       JMP          HALT                GO TO HALT ROUTINE TO        00360
                    *                                DETERMINE ACTION TO BE TAKEN 00370
                    *                                                                 00390
0002 A9640001 LDD   ORI  U  M    1$       MR,        EA+1,PICK UP 2ND WORD        00400
0004 24550000       COPY D  Q                        DATA TO (Q)                  00410
0006 29640000       COPY U  M                MR,     EA,PICK UP 1ST WD            00420
0008 24540000 LD    COPY D  A                        DATA TO (A)                  00430
000A 0000022E       JMP          RNI                 READ NEXT INSTRUCTION        00440
                    *                                                                 00450
000C A8640001 STD   ORI  U  M    1$                  SET TO STORE 2ND WORD        00460
000E 2A550000       COPY Q  D                MW,     STORE Q                      00470
0010 28640000       COPY U  M                        SET TO STORE 1ST WORD        00480
0012 2A450000 STO   COPY A  D                MW,     STORE A                      00490
0014 0000022E       JMP          RNI                                              00500
0016 2A750000 STI   COPY I  D                MW,     STORE (I) AT EA              00510
0018 0000022E       JMP          RNI                                              00520
001A 2AA50000 ST1   COPY 1  D                MW,     STORE XR1                    00530
001C 0000022E       JMP          RNI                                              00540
001E 2AB50000 ST2   COPY 2  D                MW,     STORE XR2                    00550
0020 0000022E       JMP          RNI                                              00560
0022 2AC50000 ST3   COPY 3  D                MW,     STORE XR3                    00570
0024 0000022E       JMP          RNI                                              00580
                    *                                                                 00590
0026 2C670000 LDI   COPY U  I                        EA TO (I)                    00600
0028 0000022E       JMP          RNI                                              00610
002A 2C6A0000 LD1   COPY U  1                        EA TO (1)                    00620
002C 0000022E       JMP          RNI                                              00630
002E 2C6B0000 LD2   COPY U  2                        EA TO (2)                    00640
0030 0000022E       JMP          RNI                                              00650
0032 2C6C0000 LD3   COPY U  3                        EA TO (3)                    00660
0034 0000022E       JMP          RNI                                              00670
                    *                                                                 00680
0036 9053FF00 STS   ANDI D  S    FF00$               SAVE UPPER 8 BITS            00690
0038 9C880003       ANDI X  X    3$                  SAVE INDICATORS              00700
```

```
003A 2A853000      OR   X  D  S        MW,          WRITE STATUS                    00710
003C A4080000      LDI     X     0$                RESET INDICATORS                00720
003E 0000022E      JMP              RNI                                           00730
                   *                                                              00740
0040 44544080 A    ADD  D  A  A                   ADD DATA TO (A)                 00750
0042 00000048      JMP              AOV                                           00760
                   *                                                              00770
0044 B456FFFF S    XORI D  U        FFFF$         1 S COMP DATA                   00780
0046 4C644081      ADD  U  A  A           +1      ADD WITH PLUS 1 = SUB           00790
0048 0010104C AOV  BFC  C     1     *+2           BR IF NO OVERFLOW               00800
004A AC880001      ORI  X  X        1$            SET OVFL INDICATOR              00810
004C 00180052      BTC  C     0     *+3           BR 1F CARRY = 1                 00820
004E 9C880001      ANDI X  X        1$            CLEAR CARRY INDICATOR           00830
0050 0000022E      JMP              RNI                                           00840
0052 AC880002      ORI  X  X        2$            SET CARRY INDICATOR             00850
0054 0000022E      JMP              RNI                                           00860
                   *                                                              00870
0056 14544080 AND  AND  D  A  A                   AND DATA WITH A                 00880
0058 0000022E      JMP              RNI                                           00890
005A 24544080 OR   OR   D  A  A                   OR DATA WITH A                  00900
005C 0000022E      JMP              RNI                                           00910
005E 34544080 EOR  XOR  D  A  A                   EXCLUSIVE OR WITH A             00920
0060 0000022E      JMP              RNI                                           00930
                   *                                                              00940
0062 A9640001 AD   ORI  U  M        1$      MR,   EA OF 2ND WORD                  00950
0064 44555080      ADD  D  Q  Q                   ADD TO Q                        00960
0066 29640000      COPY U  M              MR,    EA OF 1ST WORD                  00970
0068 44544082      ADD  D  A  A           CI     ADD TO A WITH CARRY             00980
006A 00000048      JMP              AOV           TEST CARRY AND OVFL             00990
                   *                                                              01000
006C A9640001 SD   ORI  U  M        1$      MR,   EA OF 2ND WD                    01010
006E B053FFFF      XORI D  S        FFFF$         COMP DATA                       01020
0070 44355081      ADD  S  Q  Q           +1      ADD TO Q WITH +1                01030
0072 29640000      COPY U  M              MR     EA OF 1ST WD                    01040
0074 B053FFFF      XORI D  S        FFFF$         COMP DATA                       01050
0076 44344082      ADD  S  A  A           CI,    ADD WITH CARRY IN               01060
0078 00000048      JMP              AOV           TEST CARRY & OVFL               01070
                   *                                                              01080
007A A0010010 M    LDI     C        10$           LOAD 16 INTO COUNTER            01090
007C 08480126      BTC  A     0     MNEG          BR IF MULTIPLIER NEG            01100
007E 2C450210      COPY A  Q              SO,R1    COPY MULTIPLIER INTO Q,RIGHT    01110
                   *                              SHIFTED 1 PLACE TO CONDITION    01120
```

```
ERR LOC.  INST.   LAB.  OP   BR DR AR OPRAND MODIFIERS AND COMMENTS                    PAGE 04

                  *                                          A SUBSEQUENT MULTIPLY STEP      01130
0080 A0030000           LDI      S     0$                    SET SIGN PLUS                   01140
0082 0000012C           JMP            MNEG+3                                              01150
                  *                                                                        01160
0084 A0010010 D         LDI      C     10$                   LOAD 16 INTO COUNTER            01170
0086 A0020164           LDI      L     DTOP                  TOP OF LOOP TO LINK             01180
0088 0848014C           BTC  A       0 DNEG                  BR IF DIVIDEND NEG              01190
008A A0030000           LDI      S     0$                    SET SIGN IF QUOT&DIV            01200
008C 24560000           COPY D   U                           SAVE MEMORY DATA
008E 08680162           BNZ  U       0 DN                    BRANCH IF DIVISOR NEG
0090 086C015C           BNZ  U         DNZ     W             BR IF DIVISOR NOT ZERO
0092 AC880001 OVFL      ORI  X   X     1$                    SET OVFL BIT                    01230
0094 0000022E           JMP            RNI                                                 01240
                  *SL DECODES MODIFIER BITS 8 & 9 TO                                       01250
                  *DETERMINE THE SHIFT OPERATION REQUIRED                                  01260
0096 005091B6 SL        BFC  D       9 SLX                   BR FOR SLA OR SLT               01270
0098 0014122E           BFC  C         RNI     R,            BR IF COUNT = 0                 01280
009A 24590000           COPY D   O                           SAVE INSTRUCTION                01290
009C AC880002           ORI  X   X     2$                    SET CARRY ON                    01300
009E 0898818E           BTC  O       8 SLC                   BR IF 8 = 1                     01310
00A0 A00200A2           LDI      L     SLCA                  TOP OF LOOP TO (LINK)           01320
00A2 0848019A SLCA      BTC  A       0 SET                   EXIT IF A0 = 1                  01330
00A4 2C440E20           COPY A   A             L1,SO,D,J,    SHIFT (A) LEFT                  01340
00A6 00000196           JMP            RSET                  TURN CARRY OFF                  01350
                  *                                                                        01360
                  *SR DECODES MODIFIER BITS 8 AND 9 TO                                     01370
                  *DETERMINE REQUIRED RIGHT SHIFT OPERATION                                01380
00A8 0014122E SR        BFC  C         RNI     R,            EXIT IF COUNT = 0               01390
00AA 005881D0           BTC  D       8 SRX                   SRT OR RTE                      01400
00AC A00200AE           LDI      L     SRA                   TOP OF LOOP TO LINK             01410
00AE 2C440C10 SRA       COPY A   A             R1,J,D,       SHIFT A AND LOOP                01420
00B0 0000022E           JMP            RNI                                                 01430
00B2 2C670000 BSC       COPY U   I                           EA TO (I)                       01440
00B4 0000022E           JMP            RNI                   READ NEXT INSTRUCTION           01450
00B6 2A750000 BSI       COPY I   D             MW,           STORE I AT EA                   01460
00B8 CC670001           ADDI U   I     1$                    EA+1 TO(I)FOR BRANCH            01470
00BA 0000022E           JMP            RNI                                                 01480
                  *                                                                        01490
                  *MDM1 COMPLETES THE MODIFICATION OF CORE                                  01500
                  *STORAGE,THEN GOES TO SKIP TO DETERMINE                                  01510
                  *IF NEXT WORD IS TO BE SKIPPED                                           01520
00BC 94598000 MDM1      ANDI D   O     8000$                 SAVE SIGN BIT OF WORD           01530
```

```
                  *                    TO BE MODIFIED                                    01540
    00BE 44563000      ADD  D  U  S                        ADD EXPANDED DISP           01550
    00C0 2A650000      COPY U  D                  MW,      WRITE BACK INTO CORE        01560
    00C2 0864022C      BFC  U          SKI        W,       DATA = ZERO                 01570
    00C4 00000222      JMP             SKIP                                            01580
```

```
                *MD1,MD2 AND MD3 MODIFY THEIR RESPECTIVE                                01590
                *INDEX REGISTERS WITH THE EFFECTIVE ADDR,                               01600
                *IN THE CASE OF LONG FORMAT INSTRUCTIONS,                               01610
                *OR WITH THE EXPANDED DISPLACEMENT IN THE                               01620
                *CASE OF SHORT FORMAT INSTRUCTIONS                                      01630
    00C6 24560040       COPY D  U             SE              EXPAND DISPLACEMENT       01640
    00C8 9CA98000 MD1   ANDI 1  0     8000$                   SIGN OF XR1               01650
    00CA 4CAA6080       ADD  1  1  U                          MODIFY XR1                01660
    00CC 2CA60000       COPY 1  U                             COPY FOR SKIP             01670
    00CE 00000222       JMP           SKIP                    NO,TEST FOR SIGN CHNG     01680
    00D0 24560040       COPY D  U             SE,             .                         01690
    00D2 9CB98000 MD2   ANDI 2  0     8000$                   .                         01700
    00D4 4CBB6080       ADD  2  2  U                          .                         01710
    00D6 2CB60000       COPY 2  U                             .                         01720
    00D8 00000222       JMP           SKIP                    .                         01730
    00DA 24560040       COPY D  U             SE,             .                         01740
    00DC 9CC98000 MD3   ANDI 3  0     8000$                   .                         01750
    00DE 4CCC6080       ADD  3  3  U                          .                         01760
    00E0 2CC60000       COPY 3  U                             .                         01770
    00E2 00000222       JMP           SKIP                    .                         01780
    00E4 A9640001 DCM   ORI  U  M     1$      ,MR             READ EA+1                 01790
    00E6 0000023E       JMP           DCMC                    CONTINUE IN R AREA        01800
    00E8 00000250 CMP   JMP           CMPC                    CONTINUE IN R AREA        01810
    00EA A9640001 XIO   ORI  U  M     1$      MR              FETCH EA+1                01820
    00EC 00000340       JMP           XIOC                    CONTINUE IN R AREA        01830
    00EE 000003C2 WT    JMP           WATE                    ENTERED BY ILLEGAL USE OF OP 5
U   00F0 00000000 LDC   JMP           LDCH                    LOAD CHARACTERISTIC
U   00F2 00000000 STC   JMP           STCH                    STORE CHARACTERISTIC
U   00F4 00000000 FAD   JMP           FADD                    FLOATING ADD
U   00F6 00000000 FSUB  JMP           SUBF                    FLOATING SUBTRACT
U   00F8 00000000 FMU   JMP           FMUL                    FLOATING MULTIPLY
U   00FA 00000000 FDIV  JMP           DIVF                    FLOATING DIVIDE
```

```
                100$ ORG                                                    01850
                *LOCATIONS 100 TO CFF ARE DESIGNATED                        01860
                *AS RANDOM ACCESS ROM AND MAY BE REACHED                    01870
                *BY BRANCHES FROM P1 AND P2 AREAS                           01880
                *                                                           01890
                *ST IS A CONTINUATION OF OPRL AND OPWL                      01900
                *ROUTINES  IN P1 AREA                                       01910
0100 CC770001 ST    ADDI I  I      1$                  INCREMENT I          01920
0102 20220050       COPY L  L              R8,          PREPARE EXIT         01930
0104 08986118       BTC  O      6  X23                  XR2 OR XR3           01940
0106 0898711E       BTC  O      7  X1                   NO,XR1               01950
0108 9C660000       ANDI U  U      0$                   ZERO TO U            01960
010A 44566080 EA    ADD  D  U  U                        (D)+(U)=EA           01970
010C 08908112       BFC  O      8  *+3                  BR IF NOT INDIRECT   01980
010E 29640000       COPY U  M              MR,          INDIRECT,READ UP EA  01990
0110 24560000       COPY D  U                           EA TO U              02000
0112 08800116       BFC  X      0  *+2                  BR IF READ FLAG CLEAR 02010
0114 29640800       COPY U  M              MR,J,        INITIATE READ & EXIT 02020
0116 28640800       COPY U  M              J,           EXIT THRU LINK       02030
                *                                                           02040
0118 08987122 X23   BTC  O      7  X3                   BR IF XR3            02050
011A 2CB60000       COPY 2  U                           (XR2) TO (U)         02060
011C 0000010A       JMP            EA                   COMPUTE EA           02070
011E 2CA60000 X1    COPY 1  U                           (XR1) TO (U)         02080
0120 0000010A       JMP            EA                   COMPUTE EA           02090
0122 2CC60000 X3    COPY 3  U                           (XR3) TO (U)
0124 0000010A       JMP            EA                   COMPUTE EFFECTIVE ADDRESS
                *MNEG IS A CONTINUATION OF THE MULTIPLY                     02100
                *ROUTINE IN P2 AREA                                         02110
0126 BC45FFFF MNEG  XORI A  Q      FFFF$                COMP MULTIPLIER TO(Q) 02120
0128 4C550211       ADD  Q  Q  0           +1,R1,SO,    +1 FOR 2'S COMP      02130
                *ALSO SHIFT MULTIPLIER TO CONDITION A                       02140
                *SUBSEQUENT MULTIPLY STEP                                   02150
012A A0030001       LDI      S     1$                   SET SIGN NEG         02160
012C A4040000       LDI      A     0$                                        02170
012E 00580134       BTC  D      0  *+3                  BR IF DATA NEG       02180
0130 24560000       COPY D  U                           POS MULTIPLICAND     02190
0132 0000013A       JMP            *+4                                       02200
0134 B456FFFF       XORI D  U      FFFF$                COMP MULTIPLIER      02210
0136 CC660001       ADDI U  U      1$                                        02220
0138 B0330001       XORI S  S      1$                   FLIP SIGN            02230
013A A002013C       LDI      L     MTOP                                      02240
```

```
     013C 5C644290 MTOP MULT U  A  A          SO,R1,          SHIFT AND ADD                    02250
                   *CONDITIONED BY PREVIOUS CONTENTS OF THE                                  02260
                   *SHIFT FLIP-FLOP                                                          02270
     013E 2C550F10      COPY Q  Q            SO,R1,SI,D,J,   FORM LEAST                       02280
                   *SIGNIFICANT  PORTION OF RESULT,ALSO                                      02290
                   *SHIFT NEXT BIT OF MULTIPLIER TO SHIFT FF                                 02300
                   *THEN LOOP UNTIL COMPLETION                                               02310
     0140 0030022E      BFC  S     0  RNI                EXIT IF POS                      02320
     0142 BC55FFFF      XORI Q  Q     FFFF$              COMP RESULT                      02330
     0144 BC44FFFF      XORI A  A     FFFF$                                               02340
     0146 CC550001      ADDI Q  Q     1$                                                  02350
     0148 4C440002      ADD  A  A  0        CI                                            02360
     014A 0000022E      JMP            RNI                                              02370
                   *DNEG IS A CONTINUATION OF THE DIVIDE                                     02380
                   *ROUTINE IN P2 AREA                                                       02390
     014C 00540092 DNEG BFC  D         OVFL   W           BR IF DIVISOR ZERO               02400
     014E BC44FFFF      XORI A  A     FFFF$              COMP DIVIDEND                    02410
     0150 BC55FFFF      XORI Q  Q     FFFF$                                               02420
     0152 CC550001      ADDI Q  Q     1$                                                  02430
     0154 4C440002      ADD  A  A  0        CI                                            02440
     0156 24560000      COPY D  U                        SAVE MEMORY DATA
     0158 A0030003      LDI      S     3$                SET SIGN OF QUOT. AND DIVIDEND
     015A 08680162      BNZ  U     0  DN                 BR IF DIVISOR NEG
     015C BC66FFFF DNZ  XORI U  U     FFFF$              COMPLEMENT DIVISOR TO
     015E CC660001      ADDI U  U     1$                 PERFORM SUBTRACT                 02490
     0160 00000164      JMP            DTOP                                              02500
     0162 B0330001 DN   XORI S  S     1$                 FLIP SIGN OF QUOTIENT
     0164 6C644280 DTOP DIV  U  A  A        SO,          TRIAL SUBTRACT OR                02530
                   *                 SUBTRACT                                            02540
     0166 2C550320      COPY Q  Q           L1,SO,SI       SHIFT DIVIDEND                   02550
                   *BIT IN AND SHIFT DIVIDEND BIT OUT                                        02560
     0168 2C440D20      COPY A  A           L1,SI,D,J,     SHIFT DIVIDEND                   02570
                   *AND LOOP TO COMPLETION                                                   02580
     016A 6C644280      DIV  U  A  A        SO,            LAST CYCLE REM NOW OK            02590
     016C 2C560320      COPY Q  U           L1,SO,SI       COMPLETE QUOT TO L               02600
     016E 00102092      BFC  C     2  OVFL                 SHIFT FF =0 = OVFL               02610
     0170 0030E182      BFC  S     E  RPOS                 BR IF REMAINDER PLUS             02620
     0172 BC45FFFF      XORI A  Q     FFFF$              COMP REMAINDER                   02630
     0174 CC550001      ADDI Q  Q     1$                 REMAINDER                        02640
     0176 08600186 QUOT BFC  U     0  UNEG                 QUOT IS NEG OK FOR OV            02650
     0178 0038F17E      BTC  S     F  QNEG                 BR,SIGN OF QUOT NEG              02660
     017A BC64FFFF      XORI U  A     FFFF$              QUOTIENT TO A                    02670
```

```
017C 0000022E      JMP          RNI                    EXIT                    02680
017E CC640001 QNEG ADDI U  A    1$                     CONVERT 1'S COMP QUOT   02690
                   *              TO 2'S COMP QUOTIENT                          02700
0180 0000022E      JMP          RNI                    EXIT                    02710
0182 2C450000 RPOS COPY A  Q                           REMAINDER TO (Q)        02720
0184 08680178      BTC  U    0   QUOT+1                 QUOT IS POS             02730
0186 0030F092 UNEG BFC  S    F   OVFL                   IF FORMED QUOTIENT IS   02740
                   *NEGATIVE AND THE SIGN OF THE QUOTIENT IS                     02750
                   *POSITIVE,AN OVERFLOW IS INDICATED                            02760
0188 CC698001      SUBI U  O    7FFF$                  IF THE FORMED QUOT      02770
                   *IS NEGATIVE AND THE SIGN OF THE QUOTIENT                     02780
                   *IS NEGATIVE AN OVERFLOW IS INDICATED                         02790
                   *EXCEPT FOR -2TO THE 15TH                                     02800
018A 0894017E      BFC  O       QNEG   W,              TEST FOR -2TO 15TH      02810
018C 00000092      JMP          OVFL                                            02820
                   *                                                            02830
018E A0020190 SLC  LDI     L    *+1                    TOP OF LOOP TO LINK     02840
0190 0848019A      BTC  A    0  SET                    EXIT IF A0 = 1          02850
0192 2C550220      COPY Q  Q             SO,L1          SHIFT Q                 02860
0194 2C440D20      COPY A  A             SI,L1,D,J,     SHIFT A AND LOOP        02870
0196 9C880001 RSET ANDI X  X    1$                     RESET CARRY             02880
0198 0000022E      JMP          RNI                                             02890
019A 089861A4 SET  BTC  O    6  *+5                    INDEX 2 OR 3            02900
019C 9CAAFF00      ANDI 1  1    FF00$                  SAVE HIGH 8 BITS,XR1    02910
019E 901300FF      ANDI C  S    FF$                    SAVE LOW 8 BITS,COUNT   02920
01A0 2CAA3000      OR   1  1  S                        COUNT TO LOW 8 OF XR1   02930
01A2 0000022E      JMP          RNI                                             02940
01A4 089871AE      BTC  O    7  *+5                    INDEX 3                 02950
01A6 9CBBFF00      ANDI 2  2    FF00$                                           02960
01A8 901300FF      ANDI C  S    FF$                                             02970
01AA 2CBB3000      OR   2  2  S                                                 02980
01AC 0000022E      JMP          RNI                                             02990
01AE 9CCCFF00      ANDI 3  3    FF00$                                           03000
01B0 901300FF      ANDI C  S    FF$                                             03010
01B2 2CCC3000      OR   3  3  S                                                 03020
01B4 0000022E      JMP          RNI                                             03030
01B6 0014122E SLX  BFC  C       RNI    R,              EXIT IF COUNT = ZERO    03040
01B8 005881C4      BTC  D    8  SLT-1                  SLT                     03050
01BA A00201BC      LDI     L    SLA                    TOP OF LOOP TO LINK     03060
01BC 2C440E20 SLA  COPY A  A             L1,SO,D,J,     SHIFT AND LOOP          03070
01BE 00102196      BFC  C    2  RSET                   SHIFT BIT = 0           03080
01C0 AC880002      ORI  X  X    2$                     SET CARRY               03090
```

```
01C2 0000022E       JMP          RNI                                    03100
01C4 A00201C6       LDI    L     SLT                TOP OF LOOP TO (LINK) 03110
01C6 2C550220 SLT   COPY Q  Q           L1,SO,       SHIFT Q               03120
01C8 2C440F20       COPY A  A           L1,SO,SI,D,J, SHIFT A &LOOP        03130
01CA 00102196       BFC  C     2  RSET                SHIFT BIT = ZERO      03140
01CC AC880002       ORI  X  X     2$                  SET CARRY             03150
01CE 0000022E       JMP          RNI                                       03160
01D0 005891E6 SRX   BTC  D     9  RTE                 RTE                   03170
01D2 A00201D6       LDI    L     SRT                TOP OF LOOP TO (LINK) 03180
01D4 C8438000       ADDI A  S     8000$               CONDITION CARRY FF    03190
01D6 2C440230 SRT   COPY A  A           SO,AV        SHIFT WITH ARITH      03200
                   *CARRY INTO A0                                          03210
01D8 2C550D10       COPY Q  Q           R1,SI,J,D,   SHIFT AND LOOP        03220
01DA 0000022E       JMP          RNI                                       03230
01DC 2C460000 G16   COPY A  U                        SWAP A AND Q          03240
01DE 2C540000       COPY Q  A                                              03250
01E0 2C650000       COPY U  Q                                              03260
01E2 0034022E       BFC  S       RNI    W,           EXIT IF COUNT = 0     03270
01E4 20310000       COPY S  C                        COUNT -16 TO COUNT    03280
01E6 9013003F RTE   ANDI C  S     3F$                SAVE COUNT IN S       03290
01E8 C033FFF0       SUBI S  S     10$                SUBTRACT 16 FROM S    03300
01EA 003001DC       BFC  S     0  G16                COUNT = OR GREATER 16 03310
01EC A00201EE       LDI    L     RT                 TOP OF LOOP TO (LINK) 03320
01EE 2C560210 RT    COPY Q  U           R1,SO,       LSB TO SHIFT FF       03330
01F0 2C440310       COPY A  A           R1,SO,SI     SHIFT A RIGHT         03340
01F2 2C550D10       COPY Q  Q           R1,SI,J,D,   SHIFT Q               03350
01F4 0000022E       JMP          RNI                                       03360
                   *                                                       03370
                   *T1 IS A CONTINUATION OF TEST FROM P1                   03380
01F6 08440FD0 T1    BFC  A       TZ     W,           (A) = ZERO            03390
01F8 08480212 TN    BTC  A     0 ANEG                (A) NEGATIVE          03400
01FA 9033000F       ANDI S  S     F$                 (A)POS NOT ZERO,CLEAR 03410
                   *INDICATORS FOR (A) NEG AND (A)ZERO                     03420
01FC 0840F200       BFC  A     F EVEN                (A) EVEN              03430
01FE 9033003B ODD   ANDI S  S     3B$                (A)ODD,CLEAR (A)EVEN  03440
0200 18933000 EVEN  AND  O  S  S                     AND WITH CONDITION    03450
                   *                 FIELD OF INSTRUCTION                  03460
0202 003C0218       BTC  S       TRUE   W,           ONE OR MORE TRUE      03470
0204 0890522E       BFC  O     5 RNI                 SHORT FORMAT,NO SKIP  03480
0206 9C887FFF       ANDI X  X     7FFF$              LONG FORMAT,BRANCH    03490
0208 0898920E       BTC  O     9 *+3                 RESET PRIORITY LEVEL  03500
020A 29740000       COPY I  M           MR,          READ 2ND WD OF INST   03510
```

```
020C 00000100      JMP           ST                    COMPUTE JUMP ADDRESS          03520
020E 20230000      COPY L  S                           SAVE (L)                      03530
0210 000002E2      JMP           SETT                  RESET LEVEL                   03540
0212 90330017 ANEG ANDI S  S     17$                   (A) NEGATIVE,CLEAR            03550
              *INDICATORS FOR (A)POS AND(A)ZERO                                      03560
0214 0848F1FE      BTC  A     F  ODD                   BR IF ODD                     03570
0216 00000200      JMP           EVEN                                               03580
0218 C9740001 TRUE ADDI I  M     1$      ,MR           SKIP IF SHORT FORMAT          03590
              *DO NOT BRANCH IF LONG FORMAT                                          03600
021A CC770001      ADDI I  I     1$                    INCREMENT I REGISTER          03610
021C 08985230      BTC  O     5  RNI+1                                              03620
021E 089892AC      BTC  O     9  SETI                  RESET INTERRUPT LEVEL         03630
0220 00000230      JMP           RNI+1                 L                             03640
              *SKIP IS USED BY MDX TO TEST AND SKIP IF                               03650
              *THE MODIFIED INDEX REGISTER OR CORE                                   03660
              *STORAGE WORD IS ZERO OR HAS CHANGED SIGN                              03670
0222 08600228 SKIP BFC  U     0  *+3                   POS OP AFTER MOD              03680
0224 0894022C      BFC  O        SKI     W,            OLD SIGN+,NEW SIGN -          03690
0226 0000022E      JMP           RNI                   BOTH SIGNS NEG                03700
0228 0864022C      BFC  U        SKI     W,            NEW SIGN-,OP = ZERO           03710
022A 0894022E      BFC  O        RNI     W,            BOTH SIGNS POS                03720
022C CC770001 SKI  ADDI I  I     1$                    BUMP I FOR SKIP COND          03730
022E 29740000 RNI  COPY I  M           MR,             MOVE I TO MEM ADDRESS         03740
              *                  AND READ NEXT INST                                  03750
0230 28D30000      COPY K  S                           MOVE MASK                     03760
0232 10633000      AND  Y  S  S                        AND MASK WITH RAW INT         03770
0234 003C025A      BTC  S        INT     W,            VALID INTERRUPT               03780
0236 CC770001      ADDI I  I     1$                    INCREMENT I                   03790
0238 20520050      COPY D  L           R8,             SHIFT OP CODE INTO L          03800
023A F0020E00      LOAD 0  L     E00$                  LOAD THE LINK FROM            03810
              *                  THE TABLE STARTING AT                               03820
              *                  E00 AND INDEXED BY                                  03830
              *                  THE CONTENTS OF L                                   03840
023C 00020F00      JMP           F00$    IX            JUMP TO P1 AREA               03850
              *                  INDEXED BY CONT OF L                                03860
023E B053FFFF DCMC XORI D  S     FFFF$                 COMPLEMENT DATA               03870
0240 4C593001      ADD  Q  O  S        +1               PERFORM SUBTRACT              03880
              *                  (Q)-(EA+1) TO (O)                                   03890
0242 29640000      COPY U  M           MR,             READ EA                       03900
0244 B053FFFF      XORI D  S     FFFF$                                               03910
0246 40334082      ADD  S  S  A        CI              (A)-(EA) TO(S)                03920
0248 08940254      BFC  O        SZRO    W,            (Q) AND (EA+1) EQUA           03930
```

```
     024A 0030004C DCNZ BFC  S      0  AOV+2                  (A)OR(AQ)GREATER THAN          03940
                  *                    (EA)OR(EA,EA+1)                                    03950
     024C CC770001      ADDI I  I     1$                      (A)OR(A,Q)LESS THAN            03960
     024E 0000004C      JMP             AOV+2                 (EA)OR(EA,EA+1)                03970
     0250 B053FFFF CMPC XORI D  S      FFFF$                   COMP DATA TO S                 03980
     0252 40334081      ADD  S  S  A           +1             (A)-(EA)TO(S)                  03990
     0254 003C024A SZRO BTC  S          DCNZ   W,              (A)AND(EA)NOT EQUAL            04000
     0256 CC770002      ADDI I  I     2$                      (A)AND(EA)EQUAL                04010
     0258 0000004C      JMP             AOV+2                 TAKE CARE OF CARRY             04020
                  *                                                                         04050
     025A A00202A6 INT  LDI     L     INTX                    LOAD EXIT INTO LINK            04080
     025C A2063804      LDI     Y     3804$  PZ               SET UP FOR LEVEL               04090
     025E 003C1C40      BTC  S          HALT   R,              META INTERRUPT                 04100
     0260 00680274      BTC  Y      0   P0                     PRIORITY ZERO                  04110
     0262 00681280      BTC  Y      1   P1                                                    04120
     0264 00682288      BTC  Y      2   P2                                                    04130
     0266 00683292      BTC  Y      3   P3                                                    04140
     0268 0068429C      BTC  Y      4   P4                                                    04150
     026A A104000D P5   LDI     M     D$     MR               READ INTERRUPT VECTOR          04160
     026C A40DF81E      LDI     K     F81E$                   MASK LEVEL 5                   04170
     026E A1070066      LDI     Z     66$    IO               LEVEL 5 IN PROCESS             04180
     0270 ACFF0400      ORI  P  P     400$                    LEVEL 5 FLAG                   04190
     0272 00020000      JMP             0$     ,IX              EXIT THRU LINK                 04200
     0274 A1040008 P0   LDI     M     8$     MR               LEVEL 0                        04210
     0276 A40D001E      LDI     K     1E$                                                   04220
     0278 A1070055      LDI     Z     55$    IO               LEVEL 5 1N PROCESS             04230
     027A ACFF8000      ORI  P  P     8000$                                                 04240
     027C 00020000      JMP             0$     IX                                          04250
     027E A1040009      LDI     M     9$     MR               LEVEL 1                        04260
     0280 A40D801E P1   LDI     K     801E$                                                 04270
     0282 A1070056      LDI     Z     56$    IO                                            04280
     0284 ACFF4000      ORI  P  P     4000$                                                 04290
     0286 00020000      JMP             0$     IX                                          04300
     0288 A104000A P2   LDI     M     A$     ,MR              LEVEL 2                        04310
     028A A40DC01E      LDI     K     C01E$                                                 04320
     028C A1070059      LDI     Z     59$    IO                                            04330
     028E ACFF2000      ORI  P  P     2000$                                                 04340
     0290 00020000      JMP             0$     IX                                          04350
     0292 A104000B P3   LDI     M     B$     MR               LEVEL 3                        04360
     0294 A40DE01E      LDI     K     E01E$                                                 04370
     0296 A107005A      LDI     Z     5A$    IO                                            04380
     0298 ACFF1000      ORI  P  P     1000$                                                 04390
```

```
     029A 00020000      JMP              $       IX                                  04400
     029C A104000C P4   LDI      M       C$      MR                                  04410
     029E A40DF01E      LDI      K       F01E$                                       04420
     02A0 A1070065      LDI      Z       65$     IO                                  04430
     02A2 ACFF0800      ORI   P  P       800$                                        04440
     02A4 00020000      JMP              0$      IX                                  04450
     02A6 24560000 INTX COPY  D  U                                EFFECTIVE ADDRESS 04460
     02A8 28640000      COPY  U  M                                                   04470
     02AA 000000B6      JMP              BSI                                         04480
     02AC A0020230 SETI LDI      L       RNI+1                    SET RETURN         04490
     02AE A2063804      LDI      Y       3804$   PZ                                  04500
     02B0 08F802C4      BTC   P     0    F0                                          04510
     02B2 08F812D2      BTC   P     1    F1                                          04520
     02B4 08F822D6      BTC   P     2    F2                                          04530
     02B6 08F832DA      BTC   P     3    F3                                          04540
     02B8 08F842DE      BTC   P     4    F4                                          04550
     02BA 08F052BC      BFC   P     5    *+1                                         04560
     02BC BCFF0400      XORI  P  P       400$                     FLIP BIT           04570
     02BE A40DFC1E PX   LDI      K       FC1E$                    ENABLE ALL INTERRUPTS 04580
     02C0 A10700AA      LDI      Z       AA$     IO                                  04590
     02C2 00020000      JMP              0$      IX                                  04600
     02C4 BCFF8000 F0   XORI  P  P       8000$                    FLIP FLAG BIT      04610
     02C6 08F81282      BTC   P     1    P1+1                     SET PROPER LEVEL   04620
     02C8 08F8228A      BTC   P     2    P2+1                                        04630
     02CA 08F83294      BTC   P     3    P3+1                                        04640
     02CC 08F8429E      BTC   P     4    P4+1                                        04650
     02CE 08F8526C      BTC   P     5    P5+1                                        04660
     02D0 000002BE      JMP              PX                                          04670
     02D2 BCFF4000 F1   XORI  P  P       4000$                                       04680
     02D4 000002C8      JMP              F0+2                                        04690
     02D6 BCFF2000 F2   XORI  P  P       2000$                                       04700
     02D8 000002CA      JMP              F0+3                                        04710
     02DA BCFF1000 F3   XORI  P  P       1000$                                       04720
     02DC 000002CC      JMP              F0+4                                        04730
     02DE BCFF0800 F4   XORI  P  P       800$                                        04740
     02E0 000002CE      JMP              F0+5                                        04750
     02E2 A00202E6 SETT LDI      L       SETX                     SET RETURN         04760
     02E4 000002AE      JMP              SETI+1                                      04770
     02E6 20320000 SETX COPY  S  L                                                04780
     02E8 29740000      COPY  I  M               MR              READ 2ND HALF OF INST 04790
     02EA 00000100      JMP              ST                                          04800
     0300 93931434 300$ HEX              93931434$                                   04820
```

```
    0302 2D2D4C4C      HEX          2D2D4C4C$          BLACK +LF                04830
    0304 86868787      HEX          86868787$          FG                       04840
    0306 82828383      HEX          82828383$          BC                       04850
    0308 88889090      HEX          88889090$          IH                       04860
    030A 00000000      HEX          0$                 BLANK                    04870
    030C 84848585      HEX          84848585$          DE                       04880
    030E 00008181      HEX          00008181$          A1                       04890
F   0310 53528036 310$HEX          53528036$          $ +7                     04900
    0312 8A000000      HEX          8A000000$          TAB                      04910
    0314 46464747      HEX          46464747$          OP                       04920
    0316 42424343      HEX          42424343$          K,L                      04930
    0318 48485050      HEX          48485050$          RO                       04940
    031A 00000000      HEX          $                                           04950
    031C 44444545      HEX          44444545$          MN                       04960
    031E 00004141      HEX          00004141$          J                        04970
F   0320 33124037 320$HEX          33124037$          , -                      04980
    0322 4D000000      HEX          4D000000$          CARRIAGE RET             04990
    0324 26262727      HEX          26262727$          WX                       05000
    0326 22222323      HEX          22222323$          ST                       05010
    0328 28283030      HEX          28283030$          YZ                       05020
    032A 00000000      HEX          $                                           05030
    032C 24242525      HEX          24242525$          UV                       05040
    032E 00002135      HEX          00002135$          0-                       05050
    0330 13162097 330$ HEX          13162097$          =01                      05060
    0332 00000000      HEX          $                                           05070
    0334 06560754      HEX          06560754$          6 7*                     05080
    0336 02960394      HEX          02960394$          2+3                      05090
    0338 08171015      HEX          08171015$          9 8                      05100
    033A 00000000      HEX          0$                                          05110
    033C 04570555      HEX          04570555$          4 5)                     05120
    033E 00000195      HEX          00000195$          /(                       05130
    0340           340$ ORG                                                    05140
    0340 24590000 XIOC  COPY D  O                      SAVE IOCC                05150
    0342 08985350       BTC  O     5  INIT           CODE 4,5,6 OR7           05160
    0344 08986368       BTC  O     6  RSEN           CODE 2 OR 3              05170
    0346 0890722E       BFC  O     7  RNI            CODE ZERO-ILLEGAL        05180
    0348 29640000       COPY U  M          MR,        READ EA                  05190
    034A 24540000       COPY D  A                     SAVE EA                  05200
    034C 21540000       COPY D  M          MR,        READ CONTENTS OF EA      05210
    034E 00000358       JMP              XIT                                   05220
    0350 08906360 INIT  BFC  O     6  OP5            CODE 4 OR 5              05230
    0352 0898737C       BTC  O     7  SENS           CODE 7                   05240
```

```
     0354 29640000       COPY U  M              MR,              READ ADDR                05250
     0356 2C940000       COPY O  A                               SAVE IOCC                05260
     0358 20530000 XIT   COPY D  S                               (EA)FOR WRITE 1 WORD     05270
                  *                    EA FOR INIT READ/RITE                            05280
                  *                    EA FOR INIT READ/RITE                            05290
     035A 2A960000       COPY O  Y              PZ,              IOCC OUT                 05300
     035C 21370000       COPY S  Z              IO,              DATA OUT                 05310
     035E 0000022E       JMP                 RNI                                         05320
     0360 08987354 OP5   BTC  O      7       INIT+2                                      05330
     0362 29640000       COPY U  M              MR,              CODE 5,INFT WRITE        05340
     0364 24540000       COPY D  A                               CODE 4,CONTROL           05350
     0366 00000358       JMP                 XIT                 SAVE CONTROL CODE       05360
     0368 0898737C RSEN  BTC  O      7       SENS                                        05370
     036A 29640000       COPY U  M              MR,              CODE 3 SENSE INT         05380
     036C 24540000       COPY D  A                               READ / WORD              05390
     036E 2B960000       COPY O  Y              PZ,IO            ADDR OOF  DATA IN        05400
     0370 23720000       COPY Z  L              PZ,IO            IOCC OUT                 05410
     0372 BC990A00       XORI O  O       A00$                   DATA IN                  05420
     0374 0890038A       BFC  O          KEYB                   TEST FOR KEYBOARD        05430
     0376 28440000 NORM  COPY A  M                              KEYBOARD                 05440
     0378 22250000       COPY L  D              MW,             ADDR                     05450
     037A 0000022E       JMP                 RNI                DATA                     05460
     037C A0020382 SENS  LDI     L       *+3                    EXIT                     05470
     037E 2B960000       COPY O  Y              PZ,IO            SET UP LOOP CONTROL      05480
     0380 A4040000       LDI     A       0$                     IOCC OUT (MAY MOVEUP     05490
     0382 0068F386       BTC  Y      F   *+2                    ZERO(A)                  05500
     0384 24744880       OR   Z  A   A                          LOOK FOR FINISH BIT      05510
     0386 25744080       OR   Z  A   A          J,              OR STATUS BITS & LOOP    05520
     0388 0000022E       JMP                 RNI                1 MORE TIME              05530
     038A B0000000 KEYB  NOP                                                             05540
                            SPACER CAN BE REMOVED IF NEW ROM IS MADE                     05550
     038C 20220050       COPY L  L              R8               SHIFT FOR LOOK UP        05560
     038E 20220210       COPY L  L              R1,SO            SHIFT +SAVE CONT BIT     05570
     0390 001823B2       BTC  C      2       TCON                CONTROL(SHIFT BII =L)    05580
     0392 20220210       COPY L  L              R1,SO            SAVE CASE BIT            05590
     0394 F0030300       LOAD O  S       300$                   PICK UP FROM TABLE       05600
     0396 0018239A       BTC  C      2   *+2                    USE RIGHT BYTE           05610
     0398 20330050       COPY S  S              R8               SHIFT TO LOW END         05620
     039A 90310007       ANDI S  C       7$                     SAVE LOW 3 BITS          05630
     039C 94390018       ANDI S  O       18$                    COL 8 AND 9              05640
     039E 2C990020       COPY O  O              L1               POSITION FOR PACK        05650
     03A0 20330060       COPY S  S              L8               POSITION DGR PACK        05660
     03A2 9033E000       ANDI S  S       E000$                  SAVE COL 11,12,0
```

```
     03A4 A00203AA       LDI      L       UNP                    ADDR OF DECODE LOOP          05670
     03A6 001413AE       BFC   C          UNP+2     R,          COL 1-7 ARE ZERO             05680
     03A8 A4062000       LDI      U       2000$                  SET 1 IN POSITION            05690
     03AA 2C660C10 UNP   COPY  U  U               R1,D,J,        SHIFT AND LOOP               05700
     03AC 4C966080       ADD   O  U  U                          COMBINE COL 1-7+8,9          05710
     03AE 42356080       ADD   S  D  U            MW            COMBINE WITH 11,12,0         05720
     03B0 0000022E       JMP                  RNI                                           05730
     03B2 002093D6 TCON  BRZ   L              TCO1   9$         IF NOT CR THEN GO            05740
     03B4 0020E3BA       BRZ   L              *+3    E$         IF NOT UPPER CASE            05750
     03B6 A2050002       LDI      D       2$     MW            SET ERF                      05760
     03B8 0000022E       JMP                  RNI                                           05770
     03BA A2050008       LDI      D       8$     MW            SET EOF                      05780
     03BC 0000022E       JMP                  RNI                                           05790
     03BE A2050004 TCO2  LDI      D       4$     MW            SET ERC                      05792
     03C0 0000022E       JMP                  RNI                                           05794
     03C2 A2063804 WATE  LDI      Y       3804$  PZ             SET TO READ SWITCH           05810
     03C4 CC77FFFF       SUBI  I  I       1$                    DECREMENT I                  05820
     03C6 B0000000       NOP                                    KILL TIME                    05830
     03C8 B0000000       NOP                                    TO READ SWITCHES             05840
     03CA 007863D2       BTC   Z        6 WRET                 START SWITCHES               05850
     03CC A2063801       LDI      Y       3801$  PZ             SET TO LIGHT WAIT            05860
     03CE A1070004       LDI      Z       4$     IO             LIGHT                        05870
     03D0 0000022E       JMP                  RNI              READ SAME WAIT INSTR         05880
     03D2 CC770001 WRET  ADDI  I  I       1$                    SET TO READ NEXT INST        05890
     03D4 00000CB8       JMP                  RUN1             TURN OFF WAIT, RUN ON        05900
```

```
                  * PART OF TYPEWRITTER ROUTINE
    03D6 0028C3BE TCO1 BNZ  L        TCO2   C$      IF BACKSPACE GO SET ERC      05902
    03D8 0020A3DE      BRZ  L        *+3    A$      IF NOT TAB                   05903
    03DA A2058110      LDI     D     8110$  MW      SET TABULATE CHAR            05904
    03DC 0000022E      JMP           RNI                                        05905
    03DE 0020F3E4      BRZ  L        *+3    F$      IF NOT LINE FEED             05906
    03E0 A2054110      LDI     D     4110$  MW      SET LINEFEED                 05907
    03E2 0000022E      JMP           RNI                                        05908
    03E4 A2050000      LDI     D     0$     MW      MUST BE SPACE                05909
    03E6 0000022E      JMP           RNI                                        05909
```

```
                  * ROUTINE TO LOAD PROGRAM FROM 3461 CARD READER                    05902
    0C00          C00$ ORG                                                          05903
    0C00 A3064F00 LOAD LDI      Y     4F00$  PZ,IO        GET STATUS              05904
    0C02 027C1C40      BNZ  Z         HALT   R,PZ         IF NOT READY GO TO HALT 05905
    0C04 A0040000      LDI      M     0$                   SET MEM ADD TO ZERO     05906
    0C06 A2050050      LDI      D     50$    MW            SET LOC ZERO TO 80 FOR WORD CNT 05907
    0C08 A2064E00      LDI      Y     4E00$  PZ            SET TO READ INTO LOC 1  05908
    0C0A A1070000      LDI      Z     0$     IO            READ A CARD             05909
    0C0C A3064F00      LDI      Y     4F00$  PZ,IO        GET STATUS              05910
    0C0E 02704C0C      BRZ  Z         *-1    4$,PZ        LOOP UNTILL OP COMPLETE 05911
    0C10 00782C40      BNZ  Z         HALT   2$           IF ERROR GO TO HALT     05912
    0C12 A3064F01      LDI      Y     4F01$  PZ,IO        RESET STATUS LINES      05913
    0C14 A4070051      LDI      I     51$                  SET MEM ADD 1 PAST END OF CARD 05914
    0C16 CC77FFFF LOD1 SUBI I   I     1$                   COUNT DOWN MEMORY ADDRESS 05915
    0C18 29740000      COPY I   M           MR            READ A WORD             05916
    0C1A 24560000      COPY D   U                         SAVE THE WORD           05917
    0C1C 22350000      COPY S   D           MW            WRITE REARANGED WORD    05918
    0C1E 08740CB8      BRZ  I         RUN1   W            EXIT IF MEM ADD ZERO    05919
    0C20 28630010      COPY U   S           R1            SHIFT ADDRESS TO RIGHT END 05920
    0C22 20330010      COPY S   S           R1                                    05921
    0C24 20330010      COPY S   S           R1                                    05922
    0C26 243D0010      COPY S   K           R1                                    05923
    0C28 90330080      ANDI S   S     0080$                SAVE BIT 5 FOR BIT 8 OF INST 05924
    0C2A 9CDD007F      ANDI K   K     007F$                SAVE DISPLACEMENT OF INST 05925
    0C2C 9C66F800      ANDI U   U     F800$                SAVE OP CODE            05930
    0C2E 2033D080      OR   S   S  K                       ASSEMBLE PARTS IN S     05932
    0C30 20336080      OR   S   S  U                                               05934
    0C32 00000C16      JMP          LOD1                                          05940
```

```
                    * ROUTINE TO LOAD PROGRAM FROM PAPER TAPE READER                05942
0C00                C00$ ORG                                  START OF LOADO        05950
0C00 A0020C08       LOAD LDI     L     DEL                    ADDRESS OF DELETE     05960
0C02 A3061C00            LDI     Y     1C00$  PZ,IO           CONTROL IOCC          05970
0C04 A3061A00            LDI     Y     1A00$  PZ,IO           READ IOCC             05980
0C06 27760850            COPY Z  U           R8,PZ,J,IO        READ                  05990
0C08 9C6900C0       DEL  ANDI U  O     C0$                    SAVE CHAN 7 AND 8     06000
0C0A BC9900C0            XORI O  O     C0$                    BOTH ARE 1 IF DELETE  06010
0C0C 08940C02            BFC  O        LOAD+1 W,              ;                     06020
0C0E A4070000            LDI     I     0$                     ZERO I REGISTER       06030
0C10 A4090010            LDI     O     10$                    SET SHIFT FLAG        06040
0C12 0868BC24       PAK  BTC  U     B  LXIT                  CHAN 5 SET            06050
0C14 9C66000F            ANDI U  U     F$                     SAVE 4 CHANNELS       06060
0C16 2C996080            OR   O  O  U                         COMBINE WITH OTHERS   06070
0C18 00182C2A            BTC  C     2  FULL                  LOOK AT SHIFT BIT     06080
0C1A A0020C1C            LDI     L     *+1                    SET UP FOR SHIFT      06090
0C1C A0010004            LDI     C     4$                     OF 4                  06100
0C1E 2C990220            COPY O  O           L1,SO            SHIFT LEFT 4          06110
0C20 A0020C12            LDI     L     PAK                    SET TO RD NEXT FRAME  06120
0C22 00000C02            JMP           LOAD+1                 READ                  06130
0C24 A4070000       LXIT LDI     I     0$                     SET START AT ZERO     06140
0C26 A2061F01            LDI     Y     1F01$  PZ              CLEAR PT INTERRUPT    06150
0C28 00000CB8            JMP           RUN1                   ENABLE INT AND GO     06160
0C2A 28950000       FULL COPY O  D                            WORD TO MEM DATA      06170
0C2C 2A740000            COPY I  M           MW               WRITE INTO CORE       06180
0C2E CC770001            ADDI I  I     1$                     ;                     06190
0C30 A0020C10            LDI     L     PAK-1                  SET FLAG AND READ     06200
0C32 00000C02            JMP           LOAD+1                 ;                     06210
0C40                C40$ ORG                                  START OF HALT SEQ     06220
0C40 A2063804       HALT LDI     Y     3804$  PZ              SET FOR SWITCHES      06230
0C42 A0020C76            LDI     L     DISP                   SET ENTRY POINT       06240
0C44 A003FFFF            LDI     S     FFFF$                  SET 'AND' MASK        06250
0C46 A4060000       H1   LDI     U     0$                     SET  OR  MASK         06260
0C48 24790000            COPY Z  O                            READ SWITCHES         06270
0C4A 08980CDE            BTC  O     0  MC                     TEST RESET SWITCHESOO 06280
0C4C A1075400            LDI     Z     5400$  IO              RESET RUN SI AND MC   06290
0C4E 0068ECF2            BTC  Y     E  PRTY                  PARITY INTERRUPT      06300
0C50 0068DCF2            BTC  Y     D  PRTY                  PROTECT INTERRUPT     06310
0C52 A2063802            LDI     Y     3802$  PZ              SET TO LIGHT LIGHTS   06320
0C54 9891000F       RPAK ANDI O  C     F$                     SAVE CODED REG BITS   06330
0C56 A40D0000            LDI     K     0$                     SET FLAG              06340
0C58 A0020C5A            LDI     L     *+1                    SET TO SHIFT          06350
```

```
     0C5A 2CDD0C20       COPY K  K          D,J,L1      CODED AMOUNT                06360
     0C5C A0020C76       LDI    L     DISP             SET ENTRY POINT             06370
     0C5E 00000C60       JMP          RSEL             DUMMY FOR PATCHING          06380
     0C60 08D8BC70 RSEL  BTC  K     B  MD               MEM DATA REGISTER           06390
     0C62 08D8ACC6       BTC  K     A  AREG             A REGISTER                  06400
     0C64 08D89CCA       BTC  K     9  QREG             Q REGISTER                  06410
     0C66 08D8ECD2       BTC  K     E  XR1              INDEX 1                     06420
     0C68 08D88CCE       BTC  K     8  IREG             I REGISTER                  06430
     0C6A 08D8DCD6       BTC  K     D  XR2              INDEX 2                     06440
     0C6C 08D8CCDA       BTC  K     C  XR3              INDEX 3                     06450
     0C6E 00000C40       JMP          HALT             ILLEGAL REG POSITION        06460
     0C70 29740000 MD    COPY I  M          MR          READ MEMORY                 06470
     0C72 245D6080       OR   D  K  U                   OR WITH SWITCH DATA         06480
     0C74 18D53000       AND  K  D  S                   AND WITH CLEAR MASK         06490
     0C76 19D73000 DISP  AND  K  Z  S          IO       LIGHT MAIN REG LAMPS        06500
     0C78 08989CAC       BTC  O     9  RUN              RUN MODE                    06510
     0C7A 0898ACA2       BTC  O     A  DS               DISPLAY OR STEP MODE        06520
     0C7C 0890BC40       BFC  O     B  HALT             ILLEGAL MODE                06530
     0C7E 24760000 RLD   COPY Z  U                     PUT SWITCHES IN MASK        06540
     0C80 0898DC8C       BTC  O     D  MDQ              MD OR Q                     06550
     0C82 A2063804 LO1   LDI    Y     3804$    PZ       SET FOR FUNCTION SW         06560
     0C84 08984C8A       BTC  O     4  *+3              CLEAR BUTTON                06570
     0C86 A003FFFF       LDI    S     FFFF$            RESTORE AND  MASK           06580
     0C88 00000C48       JMP          H1+1             LOOK FOR NEW MODE ETC       06590
     0C8A A0030000       LDI    S     0$               CLEAR'AND' MASK             06600
     0C8C 0898FC82 MDQ   BTC  O     F  LO1              A OR I REG                  06610
     0C8E A2063801       LDI    Y     3801$    PZ       AUX REGISTER                06620
     0C90 0898EC9E       BTC  O     E  QR               Q REG                       06630
     0C92 29770000       COPY I  Z          IO          DISPLAY 'I'ON AUXLTS        06640
     0C94 08906C82       BFC  O     6  LO1              START BUTTON NOT SET        06650
     0C96 2A740000 STRT  COPY I  M          MW          WRITE MEMORY                06660
     0C98 CC770001       ADDI I  I          1$          BUMP I                      06670
     0C9A 08987C40       BTC  O     7  HALT             LOOK AT RESET SIDE          06680
     0C9C 00000C9A       JMP          *-1              WAIT                        06690
     0C9E 29470000 QR    COPY A  Z          IO          DISPLAY A ON AUX LTS        06700
     0CA0 00000C82       JMP          LO1                                          06710
     0CA2 0898BCA8 DS    BTC  O     B  STEP                                       06720
     0CA4 08986C98       BTC  O     6  STRT+1          BUMP I                      06730
     0CA6 00000C40       JMP          HALT                                         06740
     0CA8 A2063804 STEP  LDI    Y     3804$    PZ       SET FOR FUNCTION'E'         06750
     0CAA A1078000       LDI    Z     8000$    IO       SET SELF INTERRUPT          06760
     0CAC A2063804 RUN   LDI    Y     3804$    PZ                                   06770
```

```
     0CAE 08986CB8       BTC  O      6  RUN1                START SWITCH SET          06780
     0CB0 A2063801       LDI     Y      3801$  PZ           SET FOR AUX LIGHTS        06790
     0CB2 99870003       ANDI X  Z      3$     IO           CARRY AND OUFL            06800
     0CB4 08982C00       BTC  O      2  LOAD                PROGRAM LOAD              06810
     0CB6 00000C40       JMP            HALT                                          06820
     0CB8 A1070800 RUN1  LDI     Z      800$   IO           SET RUN FLAG              06830
     0CBA A002022E       LDI     L      RNI                 SET ENTRY POINT           06840
     0CBC A2063802       LDI     Y      3802$  PZ           SET FOR MAIN LIGHTS       06850
     0CBE A1070000       LDI     Z      0$     IO           TURN OFF MAIN LIGHTS      06860
     0CC0 A2063801       LDI     Y      3801$  PZ           SET FOR AUX LIGHTS        06870
     0CC2 A1070000       LDI     Z      0$     IO                                     06880
     0CC4 000002C6       JMP            F0+1                SET UP INTERRUPT SYST     06890
     0CC6 2C6D4080 AREG  OR   U  K  A                       OR WITH SWITCH DATA       06900
     0CC8 1CD43800       AND  K  A  S           J,          'AND WITH CLEAR MASK      06910
     0CCA 2C6D5080 QREG  OR   U  K  Q                                                 06920
     0CCC 1CD53800       AND  K  Q  S           J,                                    06930
     0CCE 2C6D7080 IREG  OR   U  K  I                                                 06940
     0CD0 1CD73800       AND  K  I  S           J,                                    06950
     0CD2 2CAD6080 XR1   OR   1  K  U                                                 06960
     0CD4 1CDA3800       AND  K  1  S           J,                                    06970
     0CD6 2CBD6080 XR2   OR   2  K  U                                                 06980
     0CD8 1CDB3800       AND  K  2  S           J,                                    06990
     0CDA 2CCD6080 XR3   OR   3  K  U                                                 07000
     0CDC 1CDC3800       AND  K  3  S           J,                                    07010
     0CDE A4040000 MC    LDI     A      0$                                            07020
     0CE0 A4050000       LDI     Q      0$                                            07030
     0CE2 A4080000       LDI     X      0$                 STATUS                     07040
     0CE4 A40A0000       LDI     1      0$                                            07050
     0CE6 A40B0000       LDI     2      0$                                            07060
     0CE8 A40C0000       LDI     3      0$                                            07070
     0CEA A4070000       LDI     I      0$                                            07080
     0CEC A40F0000       LDI     P      0$                 PRIORITY FLAGS             07090
     0CEE A1073500       LDI     Z      3500$  IO          SET MC, RESET OTHER        07100
     0CF0 00000C40       JMP            HALT                                          07110
     0CF2 A1070200 PRTY  LDI     Z      200$   IO          SET PARITY STOP            07120
     0CF4 00000C40       JMP            HALT                                          07130
```

```
     0D00 F0F0F2F2 D00$ HEX          LDC LDC,STC STC       LOAD AMD STORE CHARACTERISTIC
     0D02 F4F4F6F6      HEX          FAD FAD,FSUB FSUB     FLOATING ADD AND SUBTRACT
     0D04 F8F8FAFA      HEX          FMU FMU,FDIV FDIV     FLOATING MULT AND DIVIDE
     0D06 EEEEEEEE      HEX          WT WT,WT WT           WAIT
     0D08 EEEEEEEE      HEX          WT WT,WT WT
     0D0A EEEEEEEE      HEX          WT WT,WT WT
     0D0C EEEEEEEE      HEX          WT WT,WT WT
     0D0E EEEEEEEE      HEX          WT WT,WT WT
     0E00 0FAC0FAC E00$ HEX          WAIT,WAIT                                    07140
     0E02 0FAC0FAC      HEX          WAIT,WAIT                                    07150
     0E04 0FAC0FAC      HEX          WAIT,WAIT                                    07160
     0E06 0FAC0FAC      HEX          WAIT,WAIT                                    07170
     0E08 EA20EA28      HEX          XIO OWSI,XIO OWS1     XIO                    07180
     0E0A EA30EA38      HEX          XIO OWS2,XIO OWS3                            07190
     0E0C EA48EA48      HEX          XIO OPWL,XIO OPWL                            07200
     0E0E EA48EA48      HEX          XIO OPWL,XIO OPWL                            07210
F    0E10 96AE96B4 E10$HEX           SL SP,SL S1           SHIFT LEFT             07220
     0E12 96BA96C0      HEX          SL S2,SL S3                                  07230
     0E14 96AE96B4      HEX          SL SP,SL S1                                  07240
     0E16 96BA96C0      HEX          SL S2,SL S3                                  07250
     0E18 A8AEA8B4      HEX          SR SP,SR S1                                  07260
     0E1A A8BAA8C0      HEX          SR S2,SR S3                                  07270
     0E1C A8AEA8B4      HEX          SR SP,SR S1                                  07280
     0E1E A8BAA8C0      HEX          SR S2,SR S3                                  07290
F    0E20 0FA80FA8 E20$HEX           LDS,LDS               LOAD STATUS            07300
     0E22 0FA80FA8      HEX          LDS,LDS                                      07310
     0E24 0FA80FA8      HEX          LDS,LDS                                      07320
     0E26 0FA80FA8      HEX          LDS,LDS                                      07330
     0E28 36003608      HEX          STS ORSI,STS ORS1     STS                    07340
     0E2A 36103618      HEX          STS ORS2,STS ORS3                            07350
     0E2C 36403640      HEX          STS OPRL,STS OPRL                            07360
     0E2E 36403640      HEX          STS OPRL,STS OPRL                            07370
F    0E30 0FAC0FAC E30$HEX           WAIT,WAIT             WAIT                   07380
     0E32 0FAC0FAC      HEX          WAIT,WAIT                                    07390
     0E34 0FAC0FAC      HEX          WAIT,WAIT                                    07400
     0E36 0FAC0FAC      HEX          WAIT,WAIT                                    07410
     0E38 0FAC0FAC      HEX          WAIT,WAIT                                    07420
     0E3A 0FAC0FAC      HEX          WAIT,WAIT                                    07430
     0E3C 0FAC0FAC      HEX          WAIT,WAIT                                    07440
     0E3E 0FAC0FAC      HEX          WAIT,WAIT                                    07450
F    0E40 B620B628 E40$HEX           BSI OWSI,BSI OWS1     BR&STORE               07460
     0E42 B630B638      HEX          BSI OWS2,BSI OWS3                            07470
```

```
    0E44 B6C6B6C6      HEX          BSI TEST,BSI TEST                          07480
    0E46 B6C6B6C6      HEX          BSI TEST,BSI TEST                          07490
    0E48 0FC60FC6      HEX          TEST,TEST             SKIP ON CONDITION    07500
    0E4A 0FC60FC6      HEX          TEST,TEST                                  07510
    0E4C B2C6B2C6      HEX          BSC TEST,BSC TEST                          07520
    0E4E B2C6B2C6      HEX          BSC TEST,BSC TEST                          07530
    0E50 E8E8E8E8 E50$ HEX          OP5S OP5S,OP5S OP5S   DECODE OP CODE 5
    0E52 E8E8E8E8      HEX          OP5S OP5S,OP5S OP5S
    0E54 E8E8E8E8      HEX          OP5S OP5S,OP5S OP5S
    0E56 E8E8E8E8      HEX          OP5S OP5S,OP5S OP5S
    0E58 E8E8E8E8      HEX          OP5S OP5S,OP5S OP5S
    0E5A E8E8E8E8      HEX          OP5S OP5S,OP5S OP5S
    0E5C E8E8E8E8      HEX          OP5S OP5S,OP5S OP5S
    0E5E E8E8E8E8      HEX          OP5S OP5S,OP5S OP5S
F   0E60 0F980F9C  E60$HEX          LDXI,LDX1             LOAD INDEX           07620
    0E62 0FA00FA4      HEX          LDX2,LDX3                                  07630
    0E64 26902A90      HEX          LDI STXL,LD1 STXL                          07640
    0E66 2E903290      HEX          LD2 STXL,LD3 STXL                          07650
    0E68 0F700F78      HEX          STXI,STX1             STORE INDEX          07660
    0E6A 0F800F88      HEX          STX2,STX3                                  07670
    0E6C 16901A90      HEX          STI STXL,ST1 STXL                          07680
    0E6E 1E902290      HEX          ST2 STXL,ST3 STXL                          07690
F   0E70 0FD20FE2  E70$HEX          MDXI,MDX1             MODIFY INDEX         07700
    0E72 0FE40FE6      HEX          MDX2,MDX3                                  07710
    0E74 BCD8C890      HEX          MDM1 MDM,MD1 STXL                          07720
    0E76 D290DC90      HEX          MD2 STXL,MD3 STXL                          07730
    0E78 0FAC0FAC      HEX          WAIT,WAIT                                  07740
    0E7A 0FAC0FAC      HEX          WAIT,WAIT                                  07750
    0E7C 0FAC0FAC      HEX          WAIT,WAIT                                  07760
    0E7E 0FAC0FAC      HEX          WAIT,WAIT                                  07770
F   0E80 40004008  E80$HEX          A ORSI,A ORS1         ADD                  07780
    0E82 40104018      HEX          A ORS2,A ORS3                              07790
    0E84 40404040      HEX          A OPRL,A OPRL                              07800
    0E86 40404040      HEX          A OPRL,A OPRL                              07810
    0E88 62206228      HEX          AD OWSI,AD OWS1       DBL ADD              07820
    0E8A 62306238      HEX          AD OWS2,AD OWS3                            07830
    0E8C 62486248      HEX          AD OPWL,AD OPWL                            07840
    0E8E 62486248      HEX          AD OPWL,AD OPWL                            07850
F   0E90 44004408  E90$HEX          S ORSI,S ORS1         SUBTRACT             07860
    0E92 44104418      HEX          S ORS2,S ORS3                              07870
    0E94 44404440      HEX          S OPRL,S OPRL                              07880
    0E96 44404440      HEX          S OPRL,S OPRL                              07890
```

```
    0E98 6C206C28      HEX          SD OWSI,SD OWS1        DBL SUB          07900
    0E9A 6C306C38      HEX          SD OWS2,SD OWS3                         07910
    0E9C 6C486C48      HEX          SD OPWL,SD OPWL                         07920
    0E9E 6C486C48      HEX          SD OPWL,SD OPWL                         07930
F   0EA0 7A007A08  EA0$HEX           M ORSI,M ORS1          MULTIPLY         07940
    0EA2 7A107A18      HEX          M ORS2,M ORS3                           07950
    0EA4 7A407A40      HEX          M OPRL,M OPRL                           07960
    0EA6 7A407A40      HEX          M OPRL,M OPRL                           07970
    0EA8 84008408      HEX          D ORSI,D ORS1          DIVIDE           07980
    0EAA 84108418      HEX          D ORS2,D ORS3                           07990
    0EAC 84408440      HEX          D OPRL,D OPRL                           08000
    0EAE 84408440      HEX          D OPRL,D OPRL                           08010
F   0EB0 E800E808  EB0$HEX           CMP ORSI,CMP ORS1      COMPARE          08020
    0EB2 E810E818      HEX          CMP ORS2,CMP ORS3                       08030
    0EB4 E840E840      HEX          CMP OPRL,CMP OPRL                       08040
    0EB6 E840E840      HEX          CMP OPRL,CMP OPRL                       08050
    0EB8 E420E428      HEX          DCM OWSI,DCM OWS1      DBL CMP          08060
    0EBA E430E438      HEX          DCM OWS2,DCM OWS3                       08070
    0EBC E448E448      HEX          DCM OPWL,DCM OPWL                       08080
    0EBE E448E448      HEX          DCM OPWL,DCM OPWL                       08090
F   0EC0 08000808  EC0$HEX           LD ORSI,LD ORS1        LOAD ACC         08100
    0EC2 08100818      HEX          LD ORS2,LD ORS3                         08110
    0EC4 08400840      HEX          LD OPRL,LD OPRL                         08120
    0EC6 08400840      HEX          LD OPRL,LD OPRL                         08130
    0EC8 02200228      HEX          LDD OWSI,LDD OWS1      DBL LOAD         08140
    0ECA 02300238      HEX          LDD OWS2,LDD OWS3                       08150
    0ECC 02480248      HEX          LDD OPWL,LDD OPWL                       08160
    0ECE 02480248      HEX          LDD OPWL,LDD OPWL                       08170
F   0ED0 0F500F58  ED0$HEX           STO1,STO1              STORE ACC        08180
    0ED2 0F600F68      HEX          STO2,STO3                               08190
    0ED4 12481248      HEX          STO OPWL,STO OPWL                       08200
    0ED6 12481248      HEX          STO OPWL,STO OPWL                       08210
    0ED8 0C200C28      HEX          STD OWSI,STD OWS1      DBL STORE        08220
    0EDA 0C300C38      HEX          STD OWS2,STD OWS3                       08230
    0EDC 0C480C48      HEX          STD OPWL,STD OPWL                       08240
    0EDE 0C480C48      HEX          STD OPWL,STD OPWL                       08250
F   0EE0 56005608  EE0$HEX           AND ORSI,AND ORS1      AND              08260
    0EE2 56105618      HEX          AND ORS2,AND ORS3                       08270
    0EE4 56405640      HEX          AND OPRL,AND OPRL                       08280
    0EE6 56405640      HEX          AND OPRL,AND OPRL                       08290
    0EE8 5A005A08      HEX          OR ORSI,OR ORS1        OR               08300
    0EEA 5A105A18      HEX          OR ORS2,OR ORS3                         08310
```

```
     0EEC 5A405A40      HEX          OR OPRL,OR OPRL                          08320
     0EEE 5A405A40      HEX          OR OPRL,OR OPRL                          08330
F    0EF0 5E005E08 EF0$HEX           EOR ORSI,EOR ORS1      EXCL OR           08340
     0EF2 5E105E18      HEX          EOR ORS2,EOR ORS3                        08350
     0EF4 5E405E40      HEX          EOR OPRL,EOR OPRL                        08360
     0EF6 5E405E40      HEX          EOR OPRL,EOR OPRL                        08370
     0EF8 0FAC0FAC      HEX          WAIT,WAIT                                08380
     0EFA 0FAC0FAC      HEX          WAIT,WAIT                                08390
     0EFC 0FAC0FAC      HEX          WAIT,WAIT                                08400
     0EFE 0FAC0FAC      HEX          WAIT,WAIT                                08410
```

```
    0F00           F00$ ORG                                                    08420
                   *THE AREA BETWEEN F00 AND FFF CA                            08430
                   *ADDDDRESSED BY THE LEAST SIGNIFICANT 8                     08440
                   *BITS INDDDEEEEXED LLOGICCALLY BBBBY F00,                   08450
                   *IS DESIGNATED P1,AAND IS USEEED FOR                        08460
                   *PRE-PROCESSING SUCH AS COMPUTING OPERAND                   08470
                   *ADDRESSES ETC                                              08480
                   *                                                           08490
                   *ORSI COMPUTES THE OPERAND ADDRESS OF                       08500
                   *SHORT FORMAT INSTRUCTIONS (REL TO I) AND                   08510
                   *INITIATTEES THE READ OF THE OPERAND                        08520
    0F00 20530040  ORSI COPY D  S            SE,          EXTEND SIGN OF DISP  08530
    0F02 41347080       ADD  S  M  I         MR,          COMP OP ADDR,READ OP 08540
    0F04 20220050       COPY L  L            R8           SHIFT P2 ADDRESS INTO 08550
                   *                  LOW 8 OF LINK                            08560
    0F06 00020000       JMP          0$      IX           JUMP TO P2 AREA      08570
                   *                  INDEXED BY CONT OF L                     08580
                   *                                                           08590
                   *ORS1,ORS2 AND ORS3 ARE IDENTICAL TO ORSI                   08600
                   *EXCEPT FOR INDEX REGISTER USED TO                          08610
                   *COMPUTE THE EFFECTIVE ADDRESS (EA) OF                      08620
                   * HE OPERAND                                                08630
    0F08 20530040  ORS1 COPY D  S            SE,                              08640
    0F0A 49A43000       ADD  1  M  S         MR,                              08650
    0F0C 20220050       COPY L  L            R8,                              08660
    0F0E 00020000       JMP          0$      IX                               08670
    0F10 20530040  ORS2 COPY D  S            SE,                              08680
    0F12 49B43000       ADD  2  M  S         MR,                              08690
    0F14 20220050       COPY L  L            R8,                              08700
    0F16 00020000       JMP          0$      IX,                              08710
    0F18 20530040  ORS3 COPY D  S            SE,                              08720
    0F1A 49C43000       ADD  3  M  S         MR,                              08730
    0F1C 20220050       COPY L  L            R8,                              08740
    0F1E 00020000       JMP          0$      IX,                              08750
                   *                                                           08760
                   *OWSI,OWS1,OWS2 AND OWS3 COMPUTE THE EA                     08770
                   *WITH RESPECT TO THE I REG AND XR1,XR2                      08780
                   *AND XR3,THE EA IS IN U AT EXIT FROM EACH                   08790
                   *ROUTINE                                                    08800
                   *                                                           08810
    0F20 20530040  OWSI COPY D  S            SE,          EXTEND SIGN OF DISP  08820
    0F22 4C763000       ADD  I  U  S                      COMPUTE EA           08830
```

```
     0F24 20220050       COPY L  L           R8,          PREPARE EXIT ADDRESS          08840
     0F26 00020000       JMP             0$  ,IX          JUMP TO OPERATION (P2)        08850
     0F28 20530040 OWS1  COPY D  S           SE                                        08860
     0F2A 4CA63000       ADD  1  U  S                                                  08870
     0F2C 20220050       COPY L  L           R8,                                       08880
     0F2E 00020000       JMP             0$  IX                                        08890
     0F30 20530040 OWS2  COPY D  S           SE                                        08900
     0F32 4CB63000       ADD  2  U  S                                                  08910
     0F34 20220050       COPY L  L           R8,                                       08920
     0F36 00020000       JMP             0$  IX                                        08930
     0F38 20530040 OWS3  COPY D  S           SE,                                       08940
     0F3A 4CC63000       ADD  3  U  S                                                  08950
     0F3C 20220050       COPY L  L           R8,                                       08960
     0F3E 00020000       JMP             0$  IX                                        08970
```

```
                *OPRL AND OPWL COMPUTE THE EFFECTIVE                                    08980
                *ADDRESS OF LONG FORMAT INSTRUCTIONS THE                                08990
                *EA IS IN U AT EXIT FROM ROUTINE, IF READ                               09000
                *IS SPECIFIED  THE READ IS INITIATED BY                                 09010
                *THE LAST INSTRUCTION OF THE ROUTINE                                    09020
                *                                                                       09030
0F40 AC888000 OPRL ORI   X  X     8000$                 SET READ FLAG                   09040
0F42 24590000      COPY  D  O                           SAVE 1ST WD OF INST             09050
0F44 29740000      COPY  I  M           MR,             READ 2ND WD OF INST             09060
0F46 00000100      JMP          ST                     CONTINUE IN R AREA              09070
                *                                                                       09080
0F48 9C887FFF OPWL ANDI  X  X     7FFF$                 CLEAR READ FLAG
0F4A 24590000      COPY  D  O                                                           09100
0F4C 29740000      COPY  I  M           MR,                                          09110
0F4E 00000100      JMP          ST                                                   09120
                *                                                                       09130
                *STOI,STO1,STO2 AND STO3 ARE SHORT FORMAT                               09140
                *STO INSTRUCTIONS,THERE OPERATION IS THE                                09150
                *SAME EXCEPT FOR THE REGISTER USED TO                                   09160
                *COMPUTE EA                                                            09170
0F50 20530040 STOI COPY  D  S           SE,             EXTEND BIT8 OF DISP            09180
0F52 48743000      ADD   I  M  S                        EA TO (M)                      09190
0F54 2A450000      COPY  A  D           MW,             (A)TO(D),WRITE CORE            09200
0F56 0000022E      JMP          RNI                    READ NEXT INSTRUCTION          09210
0F58 20530040 STO1 COPY  D  S           SE,                                          09220
0F5A 48A43000      ADD   1  M  S                                                     09230
0F5C 2A450000      COPY  A  D           MW,                                          09240
0F5E 0000022E      JMP          RNI                                                  09250
0F60 20530040 STO2 COPY  D  S           SE,                                          09260
0F62 48B43000      ADD   2  M  S                                                     09270
0F64 2A450000      COPY  A  D           MW,                                          09280
0F66 0000022E      JMP          RNI                                                  09290
0F68 20530040 STO3 COPY  D  S           SE                                           09300
0F6A 48C43000      ADD   3  M  S                                                     09310
0F6C 2A450000      COPY  A  D           MW,                                          09320
0F6E 0000022E      JMP          RNI                                                  09330
                *STXI,STX1,STX2 AND STX3 STORE THE                                     09340
                *SPECIFIED REGISTER AT EA                                              09350
0F70 20530040 STXI COPY  D  S           SE,             EXTEND BIT8                    09360
0F72 48743000      ADD   I  M  S                        COMPUTE EA                     09370
0F74 2A750000      COPY  I  D           MW,             STORE I                        09380
0F76 0000022E      JMP          RNI                                                  09390
```

```
0F78 20530040 STX1 COPY D  S           SE,                                  09400
0F7A 48743000      ADD  I  M  S                                              09410
0F7C 2AA50000      COPY 1  D           MW,                                  09420
0F7E 0000022E      JMP            RNI                                       09430
0F80 20530040 STX2 COPY D  S           SE,                                  09440
0F82 48743000      ADD  I  M  S                                              09450
0F84 2AB50000      COPY 2  D           MW,                                  09460
0F86 0000022E      JMP            RNI                                       09470
0F88 20530040 STX3 COPY D  S           SE,                                  09480
0F8A 48743000      ADD  I  M  S                                              09490
0F8C 2AC50000      COPY 3  D           MW                                   09500
0F8E 0000022E      JMP            RNI                                       09510
                   *STXL MODIFIES EFFECTIVE ADDRESS                          09520
                   *COMPUTATION  FOR CERTAIN OPERATIONS                      09530
0F90 9459FCFF STXL ANDI D  O      FCFF$          CLEAR TAG BITS TO          09540
                   *DISABLE INDEXING IN COMPUTATION OF EA                    09550
0F92 9C887FFF      ANDI X  X      7FFF$          CLEAR READ FLAG            09560
0F94 29740000      COPY I  M           MR        READ 2ND WD OF INST OF     09570
0F96 00000100      JMP            ST                                        09580
                   *                                                        09590
0F98 24570040 LDXI COPY D  I           SE,       LOAD I WITH DISPP          09600
0F9A 0000022E      JMP            RNI                                       09610
0F9C 245A0040 LDX1 COPY D  1           SE,       LOAD XR1 WITH DISP         09620
0F9E 0000022E      JMP            RNI                                       09630
0FA0 245B0040 LDX2 COPY D  2           SE        LOAD XR2 WITH DISP         09640
0FA2 0000022E      JMP            RNI                                       09650
0FA4 245C0040 LDX3 COPY D  3           SE        LOAD XR3 WITH DISPLACEMENT
0FA6 0000022E      JMP            RNI
                   *                                                        09670
0FA8 94580003 LDS  ANDI D  X      3$             SAVE STATUS BITS           09680
0FAA 0000022E      JMP            RNI                                       09690
                   *                                                        09700
0FAC 000003C2 WAIT JMP            WATE           INTERRUPTABLE HALT         09710
                   *SD,S1,S2,S3 SET THE SHIFT COUNT INTO THE                 09720
                   *COUNTER,THEN EXIT TO THE APPROPRIATE                     09730
                   *RIGHT OR  LEFT SHIFT ROUTINE                             09740
0FAE 9051003F SP   ANDI D  C      3F$            LOW 6 BITS = COUNT         09750
0FB0 20220050      COPY L  L           R8,       PREPARE EXIT               09760
0FB2 00020000      JMP            0$     IX,     JUMP TO OPERATION          09770
0FB4 98A1003F S1   ANDI 1  C      3F$                                       09780
0FB6 20220050      COPY L  L           R8,                                  09790
0FB8 00020000      JMP            0$     IX,                                09800
```

```
     0FBA 98B1003F S2   ANDI 2  C      3F$                                  09810
     0FBC 20220050      COPY L  L            R8,                            09820
     0FBE 00020000      JMP             0$   IX,                            09830
     0FC0 98C1003F S3   ANDI 3  C      3F$                                  09840
     0FC2 20220050      COPY L  L            R8,                            09850
     0FC4 00020000      JMP             0$   IX,                            09860
                        *                                                   09870
                        *TEST PERFORMS CONDITION TESTING FOR THE            09880
                        *BSC AND BSI INSTRUCTIONS                           09890
     0FC6 24590000 TEST COPY D  O                          SAVE INSTRUCTION 09900
     0FC8 B883FFFF      XORI X  S      FFFF$              FLIP STATUS      09910
     0FCA 0890F1F6      BFC  O    F    T1                 BR,OV FF NOT TESTED 09920
     0FCC 9C880002      ANDI X  X      2$                 RESET OVERFLOW FF 09930
     0FCE 084C01F8      BTC  A         TN     W,          A NOT EQUAL ZERO 09940
     0FD0 90330027 TZ   ANDI S  S      27$                A EQUAL ZERO,CLEAR 09950
                        *INDICATORS FOR (A) POS NOT ZERO AND (A)            09960
                        *NEGATIVE                                           09970
     0FD2 24560040 MDXI COPY D  U            SE,        EXTEND SIGN OF DISP 09980
     0FD4 4C677080      ADD  U  I  I                     MODIFY I         09990
     0FD6 0000022E      JMP             RNI                                10000
                        *MDM INHIBITS INDIRECT ADDRESSING AND               10010
                        *SETS UP THE OPERAND TO BE MODIFIED                 10020
     0FD8 20530040 MDM  COPY D  S            SE,        EXTEND SIGN OF DISP 10030
     0FDA 9459FFEF      ANDI D  O      FFEF$              INHIBIT INDIRECT ADDR 10040
     0FDC AC888000      ORI  X  X      8000$              SET READ FLAG    10050
     0FDE 29740000      COPY I  M            MR           READ 2ND WORD OF INST 10060
     0FE0 00000100      JMP             ST                 COMPUTE EA       10070
                        *                                                   10080
     0FE2 000000C6 MDX1 JMP             MD1-1              JUMP TO P2 AREA  10090
     0FE4 000000D0 MDX2 JMP             MD2-1                               10100
     0FE6 000000DA MDX3 JMP             MD3-1                               10110
                        *THE FOLLOWING ROUTINE EXPANDS OP CODE 5 INSTRUCTIONS INTO SPECIAL
                        *PURPOSE INSTRUCTIONS TO AUGMENT THE BASIC 1130/1800 SET
     0FE8 9052000F OP5S ANDI D  L      F$                 USE 4 BITS OF DISP AS OP CPDE
     0FEA F0020D00      LOAD 0  L      D00$               LOAD THE LINK FROM TABLE AT D00
     0FEC 00000F48      JMP             OPWL               JUMP TO TWO WORD OPERAND LOCATOR
                        END
```