MPF-I
MONITOR
PROGRAM
SOURCE LISTING

Multitech
INDUSTRIAL CORP.

OFFICE/ 9FL. 266 SUNG CHIANG ROAD, TAIPEI 104/ TAIWAN R.O.C.
TEL:(02)551-1101 TELEX:"19162 MULTIIC" FAX:(02) 542-2805
FACTORY/ 1 INDUSTRIAL E. RD., III HSINCHU SCIENCE - BASED
INDUSTRIAL PARK, HSINCHU, TAIWAN, R.O.C.

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                1  ;*************************************************************
                2  ;*                                                           *
                3  ;*     COPYRIGHT , MULTITECH INDUSTRIAL CORP. 1981           *
                4  ;*     All right reserved.                                   *
                5  ;*     No part of this software may be copied without        *
                6  ;*     the express written consent of MULTITECH              *
                7  ;*     INDUSTRIAL CORP.                                      *
                8  ;*                                                           *
                9  ;*************************************************************
               10  ;
               11  ;
               12  ;
               13  ;
               14  ;
               15  P8255    EQU     03H     ;8255 I control port
               16  DIGIT    EQU     02H     ;8255 I port C
               17  SEG7     EQU     01H     ;8255 I port B
               18  KIN      EQU     00H     ;8255 I port A
               19  PWCODE   EQU     0A5H    ;Power-up code
               20  ZSUM     EQU     71H     ;This will make the sum of all
               21                           ;monitor codes to be zero.
               22
               23  ; The following EQUATEs are used for timing.  Their values
               24  ; depend on the CPU clock frequency. (In this version, the
               25  ; crystal frequency is 1.79 MHz.)
               26
               27  COLDEL   EQU     201     ;Column delay time for routine
               28                           ;SCAN and SCAN1.
               29  F1KHZ    EQU     65      ;Delay count for 1K Hz square wave,
               30                           ;used by routin TONE1K.
               31  F2KHZ    EQU     31      ;Delay count for 2K Hz square wave,
               32                           ;used by routine TONE2K.
               33  MPERIOD  EQU     42      ;1K Hz and 2K Hz threshold, used by
               34                           ;tape input routine PERIOD.
               35
               36  ; The following EQUATEs are for tape modulation.
               37  ; If the quality of tape recorder is good, the user may
               38  ; change '4 4 2 8' to '2 2 1 4'.  This will double
               39  ; the tape data rate.
               40  ; If the quality of tape recorder is poor, the user may
               41  ; change '4 4 2 8 ' to '6 6 3 12'.  This will improve
               42  ; error performance but slow down the data rate.
               43  ; Although the data format is changed, the tape is still
               44  ; compatible in each case, because only the ratio is
               45  ; detected in the Tape-read.
               46
               47  ONE_1K   EQU     4
               48  ONE_2K   EQU     4
               49  ZERO_1K  EQU     2
               50  ZERO_2K  EQU     8
               51
               52  ;*************************************************************
               53  ; I/O port assignment: (8255 I)
               54
               55  ; port A (address 00H):
               56  ;       bit 7 -- tape input
               57  ;       bit 6 -- 'USER KEY' on keyboard, active low
               58  ;       bit 5-0 row of keyboard matrix input ,active low
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                      59  ; port B (address 01H): 7 segaments of LED, active high
                      60  ;       bit 7 -- segament d
                      61  ;       bit 6 -- decimal point
                      62  ;       bit 5 -- segament c
                      63  ;       bit 4 -- segament b
                      64  ;       bit 3 -- segament a
                      65  ;       bit 2 -- segament f
                      66  ;       bit 1 -- segament g
                      67  ;       bit 0 -- segament e
                      68  ; port C (address 02H):
                      69  ;       bit 7 -- tape & tone output
                      70  ;       bit 6 -- BREAK enable. NMI (CPU pin 17) will goes to
                      71  ;                low 5 M1's (machine cycle one) after this
                      72  ;                bit goes to low.  (This bit is connected to
                      73  ;                the reset input of external counter.)
                      74  ;       bit 5¢0 -- columns of keyboard and display matrix,
                      75  ;              active high.  Bit 5 is the leftmost column.
                      76
                      77  ;*********************************************************
                      78  ;  -- reset --
                      79  ; There are two cases that will generate a RESET signal:
                      80  ;    (i) power-up
                      81  ;   (ii) 'RS' key pressed
                      82  ; In both cases, the follow actions will be taken:
                      83  ;   a) disable interrupt, set interrupt mode to 0
                      84  ;      set I register to 00 and start execution
                      85  ;      at address 0000 (by Z80 CPU itself).
                      86  ;   b) initial user's PC to the lowest RAM address;
                      87  ;   c) set user's SP to 1F9FH;
                      88  ;   d) set user's I register to 00 and disable user's
                      89  ;      interrupt flip-flop;
                      90  ; In addition, subroutine INI will be called on power-up
                      91  ; reset, which has the following effects:
                      92  ;   e) disable BREAK POINT;
                      93  ;   f) set the contents of location 1FEEH  1FEFH to 66 and
                      94  ;      and 00 respectively. This will make instruction RST
                      95  ;      38H (opcode FF) have the same effect as BREAK.
                      96  ; Memory location POWERUP is used to distinguish power-up
                      97  ; from RS-key.  (POWERUP) contains a random data when
                      98  ; power-up and contains PWCODE (0A5H) thereafter.
                      99
0000  0600           100         LD      B,0
0002  10FE           101         DJNZ    $       ;Power-up delay
                     102
                     103  ; Initial 8255 to mode 0 with port A input, port B and C
                     104  ; output. The control word is 90H.
                     105
0004  3E90           106         LD      A,10010000B
0006  D303           107         OUT     (P8255),A
                     108
                     109  ; When the control word is sent out to 8255, all output
                     110  ; ports are cleared to 0.  It is necessary to disable
                     111  ; BREAK and deactivate all I/O by sending 0C0H to
                     112  ; port C.
                     113
0008  3EC0           114         LD      A,0C0H
000A  D302           115         OUT     (DIGIT),A
000C  31AF1F         116         LD      SP,SYSSTK       ;initial system stack
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                  117
                  118  ; If the content of location POWERUP is not equal to
                  119  ; PWCODE, call subroutine INI. Continue otherwise.
                  120
000F  3AE51F      121          LD      A,(POWERUP)
0012  FEA5        122          CP      PWCODE
0014  C4C103      123          CALL    NZ,INI
                  124
                  125  ; Determine the lowest RAM address by checking whether
                  126  ; address 1000H is RAM.  If yes, set user's PC to this
                  127  ; value.  Otherwise, set it to 1800H.
                  128
0017  210010      129          LD      HL,1000H
001A  CDF605      130          CALL    RAMCHK
001D  2802        131          JR      Z,PREPC
001F  2618        132          LD      H,18H
0021  22DC1F      133  PREPC   LD      (USERPC),HL
0024  2600        134          LD      H,0
                  135
                  136  ; Address 28H and 30H are reserved for BREAK (RST 28H)
                  137  ; and software BREAK (RST 30H).  Skip these area, monitor
                  138  ; program resumes at RESET1.
                  139
0026  180A        140          JR      RESET1
                  141  ;
                  142  ;*************************************************************
0028              143  RST28   ORG     28H
                  144  ; Address 28H is the entry point of BREAK trap.
                  145  ; If a location is set as a BREAK point, the monitor
                  146  ; will change the content of this location to C7 (RST 28H)
                  147  ; before transfering control to user's program.
                  148  ; In execution of user's program, a trap will occur if
                  149  ; user's PC passes this location.  The monitor then takes
                  150  ; over control and the content of BREAK address
                  151  ; will be restored.  Monitor takes care of everything
                  152  ; and makes the whole mechanism transparant to the user.
                  153  ; The return address pushed onto stack is the PC after
                  154  ; executing RST 28H.  The original break address should
                  155  ; be one less than that.  The following 3 instructions
                  156  ; decrease the content of (SP) by one without changing
                  157  ; HL.
                  158
0028  E3          159          EX      (SP),HL
0029  2B          160          DEC     HL
002A  E3          161          EX      (SP),HL
002B  22E81F      162          LD      (HLTEMP),HL
002E  180E        163          JR      CONT28
                  164  ;
                  165  ;*************************************************************
0030              166  RST30   ORG     30H
                  167
                  168  ; Instruction RST 30H (opcode F7) is usually used as:
                  169  ;   i) Software break;
                  170  ;  ii) Terminator of user's program.
                  171  ; The effect of this instruction is to save all user's
                  172  ; registers and return to monitor.
                  173
0030  1834        174          JR      NMI
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                  175
                  176  ;*********************************************************
                  177  ; This is a part of reset routine.  Address 0028 and
                  178  ; 0030 are reserved for break point.  Reset routine
                  179  ; skips this area and resumes here.
                  180  ;
0032   22D21F     181  RESET1  LD      (USERIF),HL     ;set user's I register and
                  182                                 ;interrupt flip flop to 0
0035   181D       183          JR      RESET2  ;monitor resumes at RESET2
                  184
                  185  ;*********************************************************
                  186
                  187  ; The following byte makes the sum of the monitor
                  188  ; code in ROM zero. ROMTEST is a self-checking routine.
                  189  ; This routine requires the sum of ROM to be zero.
                  190
0037   71         191          DEFB    ZSUM
                  192  ;
                  193  ;*********************************************************
0038              194  RST38   ORG     38H
                  195
                  196  ; Entry point of RST 38H (opcode FF) or mode 1 interrupt.
                  197  ; Fetch the address stored in location 1FEE and 1FEF,
                  198  ; then jump to this address.  Initially, 1FEE and 1FEF
                  199  ; are set to 0066.  So RST 38 will have the same effect
                  200  ; as software break.  Fy changing the content of 1FEE
                  201  ; and 1FEF, the user can define his or her own service
                  202  ; routine.
                  203  ; The next three instructions push the contents of 1FEE
                  204  ; and 1FEF to stack without changing any registers.
                  205
0038   E5         206          PUSH    HL
0039   2AEE1F     207          LD      HL,(IM1AD)
003C   E3         208          EX      (SP),HL
                  209
                  210  ; The top of the stack is now the address of user
                  211  ; defined service routine.  Pop out this address then
                  212  ; branch to it.
                  213
003D   C9         214          RET
                  215  ;
                  216  ;*********************************************************
                  217  CONT28:
                  218  ; This is a part of break service routine.  It continues
                  219  ; the program at RST28.
                  220
003E   32E71F     221          LD      (ATEMP),A
                  222
                  223  ; The monitor has changed the content of user's
                  224  ; program at break address.  The next 3 instructions
                  225  ; restored the destroyed content.  BRAD contains the
                  226  ; break address, BRDA contains the original data at
                  227  ; break address.
                  228
0041   2AE01F     229          LD      HL,(BRAD)
0044   3AE21F     230          LD      A,(BRDA)
0047   77         231          LD      (HL),A
                  232
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                  233  ; Send break enable signal to hardware counter.
                  234  ; A nonmaskable interrupt will be issued at the 5th M1's.
                  235
0048  3E80         236          LD      A,10000000B
004A  D302         237          OUT     (DIGIT),A
004C  3AE71F       238          LD      A,(ATEMP)          ; 1st M1
004F  2AE81F       239          LD      HL,(HLTEMP)        ; 2nd M1
0052  00           240          NOP                        ; 3rd M1
0053  C9           241          RET                        ; 4th M1
                  242
                  243  ; Return to user's program.  Execute the instruction
                  244  ; at break address.  After finishing one instruction,
                  245  ; a nonmaskable interrupt happens and control is
                  246  ; transferred to the monitor again.
                  247  ;
                  248  RESET2:
0054  219F1F       249          LD      HL,USERSTK
0057  22D01F       250          LD      (USERSP),HL        ;set user's SP
005A  AF           251          XOR     A
005B  32E61F       252          LD      (TEST),A
                  253
                  254  ; TEST is a flag for monitor's own use.  Illegal key-in
                  255  ; blanking (bit 7 of TEST) and automatic leading zero
                  256  ; (bit 0) use this flag.  Clear it here.
                  257
005E  DD219F07     258          LD      IX,MPF_I           ;Initial display pattern.
                  259
                  260  ; Address 0066 is the address for nonmaskable interrupt.
                  261  ; Skip this area, monitor resumes at SETST0
                  262
0062  C3D000       263          JP      SETST0
                  264  ;
                  265  ;*********************************************************
0066              266  NMI     ORG     66H
                  267
                  268  ; Entry point of nonmaskable interrupt.  NMI will occur
                  269  ; when MONI key is pressed or when user's program is
                  270  ; breaked.  The service routine which starts here saves all
                  271  ; user's registers and status.  It also check the validity
                  272  ; of user's SP.
                  273
0066  32E71F       274          LD      (ATEMP),A          ;save A register
0069  3E90         275          LD      A,10010000B
006B  D303         276          OUT     (P8255),A          ;set 8255 to mode 0.
                  277                                     ;Port A input; B,C output.
006D  3EC0         278          LD      A,0C0H
006F  D302         279          OUT     (DIGIT),A          ;disable break and LED's
0071  3AE71F       280          LD      A,(ATEMP)          ;restore A register
0074  22E81F       281  RGSAVE  LD      (HLTEMP),HL        ;save register HL
0077  E1           282          POP     HL       ;get return address from stack
0078  22DE1F       283          LD      (ADSAVE),HL        ;Save return address into
                  284                                     ;ADSAVE.
007B  22DC1F       285          LD      (USERPC),HL        ;Set user's PC to return
                  286                                     ;address.
007E  2AE81F       287          LD      HL,(HLTEMP)        ;restore HL register
0081  ED73D01F     288          LD      (USERSP),SP        ;set user's SP to current SP
0085  31D01F       289          LD      SP,USERIY+2        ;save other registers by
0088  FDE5         290          PUSH    IY                 ;continously pushing them
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

008A  DDE5       291          PUSH    IX              ;onto stack
008C  D9         292          EXX
008D  E5         293          PUSH    HL
008E  D5         294          PUSH    DE
008F  C5         295          PUSH    BC
0090  D9         296          EXX
0091  08         297          EX      AF,AF'
0092  F5         298          PUSH    AF
0093  08         299          EX      AF,AF'
0094  E5         300          PUSH    HL
0095  D5         301          PUSH    DE
0096  C5         302          PUSH    BC
0097  F5         303          PUSH    AF
                 304
                 305  ; The next two instructions save I register.
                 306  ; The interrupt flip-flop (IFF2) is copied into
                 307  ; parity flag (P/V) by instruction LD  A,I.
                 308  ; The interrupt status (enabled or disabled)
                 309  ; can be determined by testing parity flag.
                 310
0098  ED57       311          LD      A,I
009A  32D31F     312          LD      (USERIF+1),A
                 313
                 314  ; The next four instructions save IFF2 into
                 315  ; user's IFF.
                 316
009D  3E00       317          LD      A,0
009F  E2A400     318          JP      PO,SETIF        ;PO -- P/V = 0
00A2  3E01       319          LD      A,1
00A4  32D21F     320  SETIF   LD      (USERIF),A
                 321  ;
00A7  31AF1F     322          LD      SP,SYSSTK       ;set SP to system stack
                 323
                 324  ; The next 8 instructions check user's SP.
                 325  ; If the user's SP points to a location not
                 326  ; in RAM, display ERR-SP.
                 327
00AA  2AD01F     328          LD      HL,(USERSP)
00AD  DD21B507   329          LD      IX,ERR_SP
00B1  2B         330          DEC     HL
00B2  CDF605     331          CALL    RAMCHK
00B5  2019       332          JR      NZ,SETSTO
00B7  2B         333          DEC     HL
00B8  CDF605     334          CALL    RAMCHK
00BB  2013       335          JR      NZ,SETSTO
                 336
                 337  ; If the user's stack and system stack are
                 338  ; overlayed, display SYS-SP.  This checking
                 339  ; is done by the following  instructions.
                 340
00BD  DD21AF07   341          LD      IX,SYS_SP
00C1  00         342          NOP
00C2  00         343          NOP
                 344
00C3  1162E0     345          LD      DE,-USERSTK+1
00C6  19         346          ADD     HL,DE
00C7  3807       347          JR      C,SETSTO
00C9  DD21B61F   348          LD      IX,DISPBF
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

00CD  37         349          SCF             ;set carry flag to indicate
                 350                          ;the user's SP is legal.
00CE  1804       351          JR      BRRSTO
                 352  ;
                 353  SETSTO:
                 354  ; STATE is a memory location contains the monitor status.
                 355  ; It will be described in detail later.  STATE 0 stands
                 356  ; for fixed display pattern.  The initial pattern 'uPF--1'
                 357  ; or message 'SYS-SP'... belong to this category.  The next
                 358  ; two instruction set STATE to zero.
                 359
00D0  AF         360          XOR     A       ;set A to 0, also clear Carry flag
00D1  32E41F     361          LD      (STATE),A
00D4  3AE21F     362  BRRSTO  LD      A,(BRDA) ;restore the data at
                 363                           ;break address
00D7  2AE01F     364          LD      HL,(BRAD)
00DA  77         365          LD      (HL),A
                 366
                 367  ; If the user's SP is legal (carry set),
                 368  ; display user's PC and the content at PC.
                 369  ; Otherwise, display fixed message (ERR-SP
                 370  ; or SYS-SP or uPF--1)
00DB  DC0B04     371          CALL    C,MEMDP2
                 372  ;
                 373  ;
                 374  ;*************************************************************
                 375  ; Scan the display and keyboard. When a key is
                 376  ; detected, take proper action  according to the
                 377  ; key pressed.
                 378
                 379  MAIN:
00DE  31AF1F     380          LD      SP,SYSSTK       ;Initial system stack.
00E1  CDFE05     381          CALL    SCAN    ;Scan display and input keys.
                 382                          ;Routine SCAN will not return until
                 383                          ;any key is pressed.
00E4  CDCB06     384          CALL    BEEP ;After a key is detected, there
                 385                       ;will be accompanied with a beep
                 386                       ;sound.
00E7  18F5       387          JR      MAIN    ;Back to MAIN, get more keys and
                 388                          ;execute them.
                 389  ;
                 390  ;
                 391  ;*************************************************************
                 392  KEYEXEC:
                 393
                 394  ; Input key dispatch routine.
                 395  ; This routine uses the key code returned by subroutine
                 396  ; SCAN, which is one byte stored in A register.  The
                 397  ; range of key code is from 00 to 1FH.
                 398
                 399  ; (i) key code = 00 ¢ 0FH :
                 400  ;     These are hexadecimal keys.  Branch to routine KHEX.
                 401
00E9  FE10       402          CP      10H
00EB  3824       403          JR      C,KHEX
                 404
                 405  ; If the key entered is not hexadecimal, it must be a
                 406  ; function or subfunction key.  This means the previous
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                    407  ; numeric entry has terminated.  Bit 0 of TEST flag
                    408  ; must be set at the beginning of a new numeric entry.
                    409  ; This is done by the next two instructions.  (If bit 0
                    410  ; of TEST is set, the data buffer will be automatically
                    411  ; cleared when a hexadecimal key is entered.)
                    412
00ED  21E61F        413          LD      HL,TEST
00F0  CBC6          414          SET     0,(HL)
                    415
                    416  ; (ii) key code = 10H ¢ 17H :
                    417  ;      (+, -, GO, STEP, DATA, SBR, INS, DEL)
                    418  ;      There is no state corresponding to these keys.
                    419  ;      The response of them depends on the current
                    420  ;      state and minor-state. (E.g., the response of '+'
                    421  ;      key depends on the current function.  It is illegal
                    422  ;      when the display is 'uPF--1', but is legal when the
                    423  ;      display is of 'address-data' form.)  In this
                    424  ;      documentation, they are named 'sub-function key'.
                    425  ;      They are all branched by table KSUBFUN and routin
                    426  ;      BRANCH.
                    427
00F2  D610          428          SUB     10H
00F4  FE08          429          CP      8
00F6  213707        430          LD      HL,KSUBFUN
00F9  DAB003        431          JP      C,BRANCH
                    432
                    433 ;(iii) key code = 18H ¢ 1FH
                    434 ;      (PC, Addr, CBr, Reg, Move, Rela, WRtape, RDtape)
                    435 ;      These keys are named 'function key'.  They are
                    436 ;      acceptable at any time.  When they are hit, the
                    437 ;      monitor will unconditionally enter a new state.
                    438 ;      STMINOR contains the minor-state, which is required
                    439 ;      to dispatch some sub-function keys (e.g. +, - ).
                    440
00FC  DD21B61F      441          LD      IX,DISPBF
0100  D608          442          SUB     8
0102  21E41F        443          LD      HL,STATE
0105  77            444          LD      (HL),A   ;set STATE to key-code minus 18H
                    445                           ;The STATE is update here.  It will
                    446                           ;be modified later by local service
                    447                           ;routines if the function-key is PC,
                    448                           ;Addr or CBr.  For other function-
                    449                           ;keys, STATE will not be modified
                    450                           ;later.
0106  21E31F        451          LD      HL,STMINOR
0109  3600          452          LD      (HL),0   ;set STMINOR to 0
010B  214107        453          LD      HL,KFUN  ;KFUN is the base of branch table
                    454                           ;the offset is stored in A
010E  C3B003        455          JP      BRANCH
                    456
                    457 ;
                    458 ;*************************************************************
                    459 ;STATE:
                    460 ;    0=FIX          ;Display fixed pattern, e.g. 'uPF--1'.
                    461 ;    1=AD           ;The hex key entered is interpreted as
                    462                     ;memory address.
                    463 ;    2=DA           ;The hex key entered is interpreted as
                    464                     ;memory data.
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                  465  ;    3=RGFIX      ;Display fixed pattern: 'Reg-  ' and
                  466                    ;expect register name to be entered.
                  467  ;    4=MV         ;Expect parameters for 'Move' function.
                  468  ;    5=RL         ;Expect parameters for 'Rela' function.
                  469  ;    6=WT         ;Expect parameters for 'WRtape' func.
                  470  ;    7=RT         ;Expect parameters for 'RDtape' func.
                  471  ;    8=RGAD       ;Hex-key entered will be interpreted as
                  472                    ;address name for registers.
                  473  ;    9=RGDA       ;Hex-key entered will be interpreted as
                  474                    ;data for registers.
                  475  ;
                  476  ; Subroutine name conventions:
                  477  ;     (i) K???? -- K stands for key, ???? is the key name,
                  478  ;                  e.g. KINS  corresponds to key 'INS'.  Each
                  479  ;                  time a key ???? is entered, the routine
                  480  ;                  with name K???? will be executed.  All of
                  481  ;                  them are branched by table KFUN or KSUBFUN.
                  482  ;    (ii) H???? -- H stands for hexadecimal, ???? is the
                  483  ;                  current STATE.  For example, routine
                  484  ;                  HDA will be executed if the entered
                  485  ;                  key is hexadecimal and STATE is DA now.
                  486  ;                  These routines are branched by table
                  487  ;                  HTAB.
                  488  ;   (iii) I???? -- I stands for increment (+ key), ???? is
                  489  ;                  the current STATE.  E.g. IMV will be
                  490  ;                  executed when STATE is MV and '+' key
                  491  ;                  is entered.  These routines are branched
                  492  ;                  by table ITAB
                  493  ;    (iv) D???? -- D stands for decrement (- key), ???? is
                  494  ;                  the current STATE.  These routines are
                  495  ;                  branched using table DTAB.
                  496  ;     (v) G???? -- G stands for 'GO' key, ???? is the current
                  497  ;                  STATE.  These routines are branched using
                  498  ;                  table GTAB.
                  499
                  500  ;*********************************************************
                  501
                  502  ; Hexadecimal, '+', '-' and 'GO' key may be entered after
                  503  ; different function keys.  The monitor uses branch tables
                  504  ; and STATE to determine the current function and branch
                  505  ; to the proper entry point.
                  506  ;
                  507  KHEX:
                  508  ;Executed when hexadecimal keys are pressed.
                  509  ;Use HTAB and STATE for further branch.
                  510
0111  4F           511          LD      C,A      ;save A register in C
                  512                           ;which is the hex key-code.
0112  214B07       513          LD      HL,HTAB
0115  3AE41F       514  BR1     LD      A,(STATE)
0118  C3B003       515          JP      BRANCH
                  516  ;
                  517
                  518  KINC:
                  519  ;Branched by KSUBFUN table.
                  520  ;Executed when '+' key is pressed.
                  521  ;Use ITAB and STATE for further branch.
                  522  ;STATE is will be stored in A register at BR1.
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                  523
011B  215707      524          LD       HL,ITAB
011E  18F5        525          JR       BR1
                  526
                  527  ;
                  528  KDEC:
                  529  ;Branched by KSUBFUN table.  Executed
                  530  ;when '-' key is pressed.  Use DTAB and
                  531  ;STATE for further branch.  STATE will be
                  532  ;stored in A register at BR1.
                  533
0120  216307      534          LD       HL,DTAB
0123  18F0        535          JR       BR1
                  536
                  537  ;
                  538  KGO:
                  539  ;Branched by KSUBFUN table.  Executed
                  540  ;when 'GO' key is pressed.  Use GTAB and
                  541  ;STATE for further branch.  STATE will be
                  542  ;stored in A register at BR1.
                  543
0125  216F07      544          LD       HL,GTAB
0128  18EB        545          JR       BR1
                  546
                  547  ;
                  548  KSTEP:
                  549  ;Branched by table KSUBFUN.  Executed
                  550  ;when 'STEP' key is pressed.
                  551
012A  CDE503      552          CALL     TESTM    ;Check if the left 4 digits
                  553                            ;of the display are memory address.
                  554                            ;If not, disable all LED's as
                  555                            ;a warning to the user.  This
                  556                            ;is done by routine IGNORE.
012D  C2BB03      557          JP       NZ,IGNORE
0130  3E80        558          LD       A,10000000B      ;This data will be output
                  559                                   ;to port B to enable
                  560                                   ;BREAK.  It is done by
                  561                                   ;routine PREOUT.
0132  C3A302      562          JP       PREOUT
                  563
                  564  ;
                  565  KDATA:
                  566  ;Branched by table KSUBFUN.  Executed
                  567  ;when 'DATA' key is pressed.
                  568
0135  CDE503      569          CALL     TESTM    ;Check if the left 4 digits
                  570                            ;of the display are memory address.
0138  2004        571          JR       NZ,TESTRG ;If not, branch to TESTRG
                  572                             ;to check whether the display
                  573                             ;is register or not.
013A  CD0B04      574          CALL     MEMDP2   ;If yes, display the data of
                  575                            ;that address and set STATE
                  576                            ;to 2.
013D  C9          577          RET
013E  FE08        578  TESTRG  CP       8        ;check if the status is 8 or 9
                  579                            ;(RGAD or RGDA).
0140  DABB03      580          JP       C,IGNORE ;If not, ignore this key and
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                    581                             ;send out a warning message.
0143  CD7704        582             CALL    REGDP9  ;If yes, display register and
                    583                             ;set status to 9 (RGDA).
0146  C9            584             RET
                    585
                    586 ;
                    587 KSBR:
                    588 ;Branched by table KSUBFUN.  Executed
                    589 ;when 'SBr' key (set break point) is
                    590 ;pressed.
                    591
0147  CDE503        592             CALL    TESTM   ;Check if the display is of
                    593                             ;'address-data' form.
014A  C2BB03        594             JP      NZ,IGNORE ;If not, ignore this key and
                    595                             ;send out a warning message.
014D  2ADE1F        596             LD      HL,(ADSAVE) ;If yes, get the address
                    597                               ;being display now.
0150  CDF605        598             CALL    RAMCHK  ;Check if this address is
                    599                             ;in RAM.
0153  C2BB03        600             JP      NZ,IGNORE ;If not, ignore the 'SBR' key
                    601                             ;and send out a warning message.
0156  22E01F        602             LD      (BRAD),HL ;If yes, set this address as
                    603                             ;a break point.
0159  CD0B04        604             CALL    MEMDP2  ;Display the data of break
                    605                             ;address and set STATE to
                    606                             ;2 (DA).
015C  C9            607             RET
                    608
                    609 ;
                    610 KINS:
                    611 ;Branched by table KSUBFUN.  Executed
                    612 ;when 'Ins' key (insert) is pressed.
                    613
015D  CDE503        614             CALL    TESTM   ;Check if the display is of
                    615                             ;'address-data' form now.
0160  C2BB03        616             JP      NZ,IGNORE ;If not, ignore the 'INS' key
                    617                             ;and send out a warning message.
0163  2ADE1F        618             LD      HL,(ADSAVE) ;If yes, get the address being
                    619                               ;displayed now.
                    620
0166  00            621             NOP
                    622
0167  22AF1F        623             LD      (STEPBF),HL ; Store this address in
                    624                               ;STEPBF and the next address
                    625                               ;in STEPBF+4 for later use.
016A  23            626             INC     HL
016B  22B31F        627             LD      (STEPBF+4),HL
016E  CDF605        628             CALL    RAMCHK  ;Check if the address to be
                    629                             ;inserted is in RAM.
0171  C2BB03        630             JP      NZ,IGNORE ;If not, ignore the 'INS' key
                    631                               ;and send out a warning message.
                    632                             ;If the address to be inserted
                    633                             ;is in 1800-1DFF,store 1DFE into
                    634                             ;STEPBF+2
                    635                             ;Otherwise, ignore the 'INS' key.
                    636                             ;This is done by the following
                    637                             ;instructions.
0174  11FE1D        638             LD      DE,1DFEH
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

0177  7C          639          LD      A,H
0178  FE1E        640          CP      1EH
017A  3807        641          JR      C,SKIPH1
017C  FE20        642          CP      20H
017E  DABB03      643          JP      C,IGNORE
0181  1627        644          LD      D,27H
0183  ED53B11F    645  SKIPH1  LD      (STEPBF+2),DE
                  646
                  647  ;When one byte is inserted at some
                  648  ;address, all data below this address
                  649  ;will be shifted down one position.
                  650  ;The last location will be shifted out
                  651  ;and therefore lost.
                  652  ;The RAM is divided into 3 blocks as
                  653  ;insert is  concerned.  They are:
                  654  ;1800-1DFF,1E00-1FFF and 2000-27FF
                  655  ;The 2 nd block cannot be inserted and
                  656  ;is usually used as data bank.  System
                  657  ;data that of course cannot be shifted
                  658  ;are also stored in this bank.  Each
                  659  ;block is independent of the other when
                  660  ;shift is  performed, i.e. the data
                  661  ;shifted out of the first block will not
                  662  ;be propagated to next block.
                  663  ;The shift is accomplished by block
                  664  ;transfer, i.e. MOVE.  This is the
                  665  ;job of subroutine GMV.
                  666  ;Routine GMV needs 3 parameters which
                  667  ;are stored in step-buffer (STEPBF):
                  668  ;STEPBF: starting address (2 bytes);
                  669  ;STETBF+2: ending address (2 bytes);
                  670  ;STEPBF+4: destination address (2 bytes).
                  671
0187  CDE402      672  DOMV    CALL    GMV
018A  AF          673          XOR     A    ;After the RAM has been shifted down,
                  674                       ;the data of the address to be inserted
                  675                       ;is cleared to zero.  This is done by
                  676                       ;the next two instructions.  Register
                  677                       ;DE contain inserted address after GMV
                  678                       ;is performed.
018B  12          679          LD      (DE),A
018C  2AB31F      680          LD      HL,(STEPBF+4) ;Store the data in (STEPBF+4)
018F  22DE1F      681          LD      (ADSAVE),HL   ;into (ADSAVE).
0192  CD0B04      682          CALL    MEMDP2  ;Display the address and data, also
                  683                          ;set STATE to 2.
0195  C9          684          RET
                  685  ;
                  686  KDEL:
                  687  ;Branched by table KSUBFUN.  Executed
                  688  ;when 'Del' (delete) key is pressed.
                  689
0196  CDE503      690          CALL    TESTM   ;Check if the display is of
                  691                          ;'address-data' form.
0199  C2BB03      692          JP      NZ,IGNORE ;If not, ingore the 'Del' key and
                  693                          ;send out a warning message.
                  694                          ;'Delete' is quite similar to
                  695                          ;'Insert', except that the memory
                  696                          ;is shifted up instead of shifted
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                  697                           ;down.  See the comments on
                  698                           ;routine KINS for detail.
019C  2ADE1F      699           LD      HL,(ADSAVE) ;Get the address being displayed
                  700                             ;now.  This is the address to
                  701                             ;be deleted.
                  702
                  703
019F  00          704           NOP
                  705
01A0  22B31F      706           LD      (STEPBF+4),HL
01A3  CDF605      707           CALL    RAMCHK  ;Check if the address is in RAM.
01A6  C2BB03      708           JP      NZ,IGNORE ;If not, ignore this key and
                  709                             ;send out a warning message.
                  710                           ;Following instructions prepare the
                  711                           ;parameters for routine GMV in step-
                  712                           ;buffer.  Refer to routine KINS for
                  713                           ;detail.
01A9  11001E      714           LD      DE,1E00H
01AC  7C          715           LD      A,H
01AD  FE1E        716           CP      1EH
01AF  3807        717           JR      C,SKIPH2
01B1  FE20        718           CP      20H
01B3  DABB03      719           JP      C,IGNORE
01B6  1628        720           LD      D,28H
01B8  ED53B11F    721  SKIPH2   LD      (STEPBF+2),DE
01BC  23          722           INC     HL
01BD  22AF1F      723           LD      (STEPBF),HL
01C0  18C5        724           JR      DOMV
                  725  ;
                  726  ;*************************************************************
                  727  KPC:
                  728  ; Branched by table KFUN.  Executed when
                  729  ; 'PC' key is pressed.
                  730
01C2  2ADC1F      731           LD      HL,(USERPC) ;Store the user's program
01C5  22DE1F      732           LD      (ADSAVE),HL ;counter into (ADSAVE)
01C8  CD0B04      733           CALL    MEMDP2  ;Routine MEMDP2 displays the address
                  734                           ;in (ADSAVE) and its data.  It also
                  735                           ;set the STATE to 2.
01CB  C9          736           RET
                  737  ;
                  738  KCBR:
                  739  ; Branched by table KFUN.  Executed when
                  740  ; 'CBr' (clear break point) key is pressed.
                  741
01CC  CDDE03      742           CALL    CLRBR   ;Call subroutine CBRBR to clear
                  743                           ;break point.  When returned, the HL
                  744                           ;register will contain FFFF.
01CF  22DE1F      745           LD      (ADSAVE),HL ;Store FFFF into (ADSAVE)
01D2  CD0B04      746           CALL    MEMDP2  ;Display address and its data.  Also
                  747                           ;set STATE to 2.
01D5  C9          748           RET
                  749  ;
                  750  KREG:
                  751  ; Branched by table KFUN.  Executed when
                  752  ; 'Reg' key is pressed.
01D6  DD21CA07    753           LD      IX,REG_ ;Routine SCAN uses IX as a pointer
                  754                           ;for display buffer.  Set IX to REG
```

```
 LOC    OBJ CODE M STMT SOURCE STATEMENT

                   755                          ;will make SCAN displays 'Reg-
01DA    CDC404     756          CALL    FCONV   ;Decode user's flag F and F' to
                   757                          ;binary display format.  This
                   758                          ;format will be used later, when
                   759                          ;user requires the monitor to
                   760                          ;display decoded flag by pressing
                   761                          ;keys 'SZXH', 'XPNC',...
01DD    C9         762          RET
                   763  ;
                   764  KADDR:
                   765  ; Branched by KFUN table.  Executed when
                   766  ; 'Addr' key is pressed.
                   767
01DE    CD0204     768          CALL    MEMDP1  ;Display the address stored in
                   769                          ;(ADSAVE) and its data.  Set STATE
                   770                          ;to 1 (AD).
01E1    C9         771          RET
                   772  ;
                   773  ; Function Move, Relative, Read-tape and
                   774  ; Write-tape require from one to three
                   775  ; parameters.  They are stored in STEPBF
                   776  ; (step buffer).  STMINOR (minor status)
                   777  ; contains the number of parameters has been
                   778  ; entered.  For Move and Relative, the
                   779  ; default value of the first parameter is
                   780  ; the address stored in (ADSAVE).  There
                   781  ; is no default value for the first parameter
                   782  ; (filename) of Read- and Write-tape.  When the
                   783  ; function keys are pressed, STMINOR is automatically
                   784  ; reset to 0.
                   785  ;
                   786
                   787  KMV:
                   788  ; Branched by table KFUN.  Executed when
                   789  ; 'Move' key is pressed.
                   790  KRL:
                   791  ; Branched by table KFUN.  Executed when
                   792  ; 'Rela' (relative) key is pressed.
01E2    2ADE1F     793          LD      HL,(ADSAVE) ;Store the contents of ADSAVE
                   794                              ;into STEPBF as default value
                   795                              ;of first parameter.
01E5    22AF1F     796          LD      (STEPBF),HL
                   797  KWT:
                   798  ; Branched by table KFUN.  Executed
                   799  ; when 'WRtape' key is pressed.
                   800
                   801  KRT:
                   802  ; Branched by table KFUN.  Executed when
                   803  ; 'RDtape' key is pressed.
                   804
01E8    CD3A04     805          CALL    STEPDP  ;Display the parameter that
                   806                          ;is being entered now by calling
                   807                          ;subroutine STEPDP.
01EB    C9         808          RET
                   809  ;
                   810  ;*************************************************************
                   811  ; The following subroutines with name H???
                   812  ; are the service routine for hexadecimal
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                  813  ; keys corresponding to each STATE.  They
                  814  ;  are all branched by table HTAB and STATE.
                  815
01EC  C3BB03      816  HFIX    JP      IGNORE ;When the display is fixed pattern
                  817                         ;hexadecimal keys are illegal.
                  818                         ;Disable all LED's as a warning
                  819                         ;message to the user.  This is what
                  820                         ;routine IGNORE does.
                  821  ;
01EF  2ADE1F      822  HDA     LD      HL,(ADSAVE) ;Get the address being displayed
                  823                              ;now from (ADSAVE)
01F2  CDF605      824          CALL    RAMCHK  ;Check if it is in RAM.
01F5  C2BB03      825          JP      NZ,IGNORE ;If not, ignore this key and
                  826                          ;send out a warning message.
01F8  CDEE03      827          CALL    PRECL1  ;If this is the first hexadecimal
                  828                          ;key entered after function or sub-
                  829                          ;function key,reset the data of that
                  830                          ;address to 0. (by routine PERCL1)
01FB  79          831          LD      A,C     ;The key-code is saved in C at
                  832                          ;routine KHEX.  Restore it to A.
01FC  ED6F        833          RLD             ;Rotate the key-code (4 bits) into
                  834                          ;the address obtained above. (in HL)
01FE  CD0B04      835          CALL    MEMDP2  ;Display the address and data,
                  836                          ;then set STATE to 2 (DA).
0201  C9          837          RET
                  838  ;
0202  21DE1F      839  HAD:    LD      HL,ADSAVE
0205  CDFA03      840          CALL    PRECL2  ;If this is the first hezdecimal
                  841                          ;key after function key is entered,
                  842                          ;set the contents of ADSAVE to 0.
0208  79          843          LD      A,C     ;The key-code is saved in C
                  844                          ;by routine KHEX.
                  845                          ;The next three instructions shift
                  846                          ;the address being displayed  by
                  847                          ;one digit.
0209  ED6F        848          RLD
020B  23          849          INC     HL
020C  ED6F        850          RLD
020E  CD0204      851          CALL    MEMDP1  ;Display the address and its
                  852                          ;data.  Also, set STATE to 1.
0211  C9          853          RET
                  854  ;
                  855  HRGAD:
                  856  HRGFIX:
0212  79          857          LD      A,C
0213  DD21B61F    858          LD      IX,DISPBF
0217  21E31F      859          LD      HL,STMINOR
021A  87          860          ADD     A,A     ;The key-code is the register
                  861                          ;name.  Double it and store it
                  862                          ;into STMINOR.
021B  77          863          LD      (HL),A
021C  CD7304      864          CALL    REGDP8  ;Display register and set
                  865                          ;STATE to 8. (RGAD)
021F  C9          866          RET
                  867  ;
                  868  HRT:
                  869  HWT:
                  870  HRL:
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

0220  CD5504       871  HMV:    CALL    LOCSTBF ;Use STMINOR and STEPBF
                   872                         ;to calculate the address
                   873                         ;of current parameter in
                   874                         ;step buffer.
0223  CDFA03       875          CALL    PRECL2  ;If this is the first hex
                   876                         ;key entered, cleared the
                   877                         ;parameter (2 bytes) by
                   878                         ;PRECL2.
0226  79           879          LD      A,C     ;C contains the key-code.
                   880                         ;Rotate the parameter (2 bytes)
                   881                         ;1 digit left with the key-code.
0227  ED6F         882          RLD
0229  23           883          INC     HL
022A  ED6F         884          RLD
022C  CD3A04       885          CALL    STEPDP  ;Display the parameter.
022F  C9           886          RET
                   887  ;
0230  CDBB04       888  HRGDA   CALL    LOCRGBF ;Calculate the address of
                   889                         ;the register being modified.
0233  CDEE03       890          CALL    PRECL1  ;If this is the first hex
                   891                         ;key entered.  Clear the register
                   892                         ;(1 byte) by PRECL1.
0236  79           893          LD      A,C     ;Rotate user's register (1 byte)
                   894                         ;1 digit left with the key-code
                   895                         ;stored in C.
0237  ED6F         896          RLD
0239  CD7704       897          CALL    REGDP9  ;Display the register and set
                   898                         ;STATE to 9 (RGDA).
023C  C9           899          RET
                   900  ;
                   901  ;*************************************************************
                   902   ;The following routines with name
                   903   ;I???? are the service routines for
                   904   ;'+' key corresponding to each STATE.
                   905   ;They are all branched by table ITAB
                   906   ;and STATE.
                   907
                   908  IFIX:
                   909  IRGFIX:
023D  C3BB03       910          JP      IGNORE ;'+' key is illeagl for state
                   911                         ;FIX or RGFIX, ignore it.
                   912  ;
                   913  IAD:
0240  2ADE1F       914  IDA:    LD      HL,(ADSAVE) ;Increase the address being
                   915                          ;displayed now (in ADSAVE)
                   916                          ;by 1.
0243  23           917          INC     HL
0244  22DE1F       918          LD      (ADSAVE),HL
0247  CD0B04       919          CALL    MEMDP2  ;Display the address and data,
                   920                         ;then set the STATE to 2.
024A  C9           921          RET
                   922  ;
                   923  IRT:
                   924  IWT:
                   925  IRL:
024B  21E31F       926  IMV:    LD      HL,STMINOR ;STMINOR contains the
                   927                          ;parameter count, increment
                   928                          ;it by one.
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

024E  34          929          INC     (HL)
024F  CD5F04      930          CALL    LOCSTNA ;Check if the count is
                  931                          ;overflowed.
0252  2004        932          JR      NZ,ISTEP ;If not overflowed, continue
                  933                           ;at ISTEP.
0254  35          934          DEC     (HL)    ;Otherwise, restore the count
                  935                          ;and ignore the '+' key.
0255  C3BB03      936          JP      IGNORE
0258  CD3A04      937  ISTEP   CALL    STEPDP  ;Display the parameter at
                  938                          ;step buffer.
025B  C9          939          RET
                  940  ;
                  941  IRGAD:
025C  21E31F      942  IRGDA:  LD      HL,STMINOR ;In these states, the STMINOR
                  943                             ;contains the register name.
                  944                             ;Increase it by 1.  If it
                  945                             ;reaches the last one, reset
                  946                             ;it to the first one (0).
025F  34          947          INC     (HL)
0260  3E1F        948          LD      A,1FH
0262  BE          949          CP      (HL)
0263  3002        950          JR      NC,IRGNA
0265  3600        951          LD      (HL),0
0267  CD7704      952  IRGNA   CALL    REGDP9  ;Display the register and
                  953                          ;set STATE to 9.
026A  C9          954          RET
                  955  ;
                  956  ;*************************************************************
                  957   ;The following routines with name
                  958   ;D???? are the service routines for
                  959   ;'-' key corresponding to each state.
                  960   ;They are all branched by table DTAB
                  961   ;and STATE.
                  962
                  963  DFIX:
                  964  DRGFIX:
026B  C3BB03      965          JP      IGNORE ;'-' key is illegal for
                  966                         ;these states.  Ignore it.
                  967  ;
                  968  DAD:
026E  2ADE1F      969  DDA:    LD      HL,(ADSAVE) ;Decrease the address being
                  970                              ;displayed now (in ADSAVE)
                  971                              ;by one.
0271  2B          972          DEC     HL
0272  22DE1F      973          LD      (ADSAVE),HL
0275  CD0B04      974          CALL    MEMDP2  ;Display the address and data,
                  975                          ;set STATE to 2 (DA).
0278  C9          976          RET
                  977  ;
                  978  DRT:
                  979  DWT:
                  980  DRL:
0279  21E31F      981  DMV:    LD      HL,STMINOR ;In these states, STMINOR
                  982                             ;contains the parameter count.
                  983                             ;Decrease it by one.  If overflow
                  984                             ;occurs, restore STMINOR and
                  985                             ;ignore the '-' key.  Otherwise
                  986                             ;continue at DSTEP.
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

027C  35          987          DEC     (HL)
027D  CD5F04      988          CALL    LOCSTNA
0280  2004        989          JR      NZ,DSTEP
0282  34          990          INC     (HL)
0283  C3BB03      991          JP      IGNORE
0286  CD3A04      992  DSTEP   CALL    STEPDP  ;Display the parameter.
0289  C9          993          RET
                  994  ;
                  995  DRGAD:
028A  21E31F      996  DRGDA:  LD      HL,STMINOR ;In these states, STMINOR
                  997                         ;contains the register name.
                  998                         ;Decrease it by one.' If it
                  999                         ;goes below zero, set it to
                 1000                         ;the highest value (1F).
028D  35         1001          DEC     (HL)
028E  3E1F       1002          LD      A,01FH
0290  BE         1003          CP      (HL)
0291  3002       1004          JR      NC,DRGNA
0293  361F       1005          LD      (HL),1FH
0295  CD7704     1006  DRGNA   CALL    REGDP9  ;Display the register and
                 1007                         ;set STATE to 9.
0298  C9         1008          RET
                 1009  ;
                 1010  ;*************************************************************
                 1011   ;The following routines with name
                 1012   ;G???? are the service routines for
                 1013   ;'GO' key  corresponding to each
                 1014   ;state.  They are all branched by
                 1015   ;table GTAB and STATE.
                 1016
                 1017  GFIX:
                 1018  GRGFIX:
                 1019  GRGAD:
0299  C3BB03     1020  GRGDA:  JP      IGNORE ;'GO' key is illegal for
                 1021                         ;these states.  Ignore it.
                 1022  ;
                 1023  GAD:
029C  2AE01F     1024  GDA:    LD      HL,(BRAD) ;Get the address of break
                 1025                         ;point.
029F  36EF       1026          LD      (HL),0EFH ;Instruction RST 28H.
                 1027                         ;The content of break address
                 1028                         ;is changed to RST 28H before
                 1029                         ;the control is transfered to
                 1030                         ;user's program.  This
                 1031                         ;will cause a trap when user's
                 1032                         ;PC passes this point.
02A1  3EFF       1033          LD      A,0FFH  ;Save FF into TEMP.  This data
                 1034                         ;will be output to port B later.
                 1035                         ;FF is used to disable break point.
02A3  32EA1F     1036  PREOUT  LD      (TEMP),A   ;Store A into TEMP.
02A6  3AD21F     1037          LD      A,(USERIF) ;Save two instructions into
                 1038                         ;TEMP and TEMP+1.  These two
                 1039                         ;instructions will be executed
                 1040                         ;later.  If the user's IFF
                 1041                         ;(interrupt flip-flop) is 1,
                 1042                         ;the instructions are 'EI RET'.
                 1043                         ;Otherwise, they are 'DI RET'.
02A9  CB47       1044          BIT     0,A
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

02AB  21FBC9     1045          LD      HL,0C9FBH       ;'EI','RET'
02AE  2002       1046          JR      NZ,EIDI
02B0  2EF3       1047          LD      L,0F3H          ;'DI'
02B2  22EB1F     1048  EIDI    LD      (TEMP+1),HL
02B5  31BC1F     1049          LD      SP,REGBF ;Restore user's registers by
                 1050                   ;setting SP to REGBF (register
                 1051                   ;buffer) and continuously popping
                 1052                   ;the stack.
02B8  F1         1053          POP     AF
02B9  C1         1054          POP     BC
02BA  D1         1055          POP     DE
02BB  E1         1056          POP     HL
02BC  08         1057          EX      AF,AF'
02BD  F1         1058          POP     AF
02BE  08         1059          EX      AF,AF'
02BF  D9         1060          EXX
02C0  C1         1061          POP     BC
02C1  D1         1062          POP     DE
02C2  E1         1063          POP     HL
02C3  D9         1064          EXX
02C4  DDE1       1065          POP     IX
02C6  FDE1       1066          POP     IY
02C8  ED7BD01F   1067          LD      SP,(USERSP) ;Restore user's SP.
02CC  32BD1F     1068          LD      (USERAF+1),A ;Temporarily save A
02CF  3AD31F     1069          LD      A,(USERIF+1) ;Restore user's I
02D2  ED47       1070          LD      I,A
02D4  E5         1071          PUSH    HL      ;The next 3 instructions
                 1072                   ;push the address being
                 1073                   ;displayed now (in ADSAVE)
                 1074                   ;onto stack without changing
                 1075                   ;HL register.  This address will be
                 1076                   ;treated as user's new PC.
02D5  2ADE1F     1077          LD      HL,(ADSAVE)
02D8  E3         1078          EX      (SP),HL
02D9  3AEA1F     1079          LD      A,(TEMP) ;Output the data stored in
                 1080                   ;TEMP to port B of 8255.
                 1081                   ;This data is prepared by
                 1082                   ;routine KSTEP or GAD or
                 1083                   ;GDA.  In first case, it is
                 1084                   ;10111111 and will enable
                 1085                   ;break point.  In other
                 1086                   ;cases, it is FF and will
                 1087                   ;disable break point.
                 1088                   ;If break is enabled, non-
                 1089                   ;maskable interrupt will occur
                 1090                   ;5 M1's after the OUT instruction.
02DC  D302       1091          OUT     (DIGIT),A
02DE  3ABD1F     1092          LD      A,(USERAF+1) ;1st M1,
                 1093                       ;Restore A register.
02E1  C3EB1F     1094          JP      TEMP+1  ;2nd M1,
                 1095                   ;Execute the two instructions
                 1096                   ;stored in RAM.  They are:
                 1097                   ;    EI (or DI)     ;3rd M1
                 1098                   ;    RET            ;4th M1
                 1099                   ;The starting address of user's
                 1100                   ;program has been pushed onto
                 1101                   ;the top of the stack.  RET pops
                 1102                   ;out this address and transfers
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                1103                            ;control to it.  The first M1
                1104                            ;of user's program will be the
                1105                            ;5th M1 after OUT. If break point
                1106                            ;is enabled, NMI will occur after
                1107                            ;this instruction is completed.
                1108                            ;This is the mechanism of single
                1109                            ;step.
                1110  ;
                1111  ;*********************************************************
02E4  21AF1F    1112  GMV       LD      HL,STEPBF
02E7  CD3D05    1113            CALL    GETP    ;Load parameters from
                1114                            ;step buffer into registers.
                1115                            ;Also check if the parameters
                1116                            ;are legal.  After GETP,
                1117                            ;HL = start address of source
                1118                            ;BC = length to MOVE.
02EA  3867      1119            JR      C,ERROR ;Jump to ERROR if the
                1120                            ;parameters are illegal. (I.e., Ending
                1121                            ;address < starting address.)
02EC  ED5BB31F  1122            LD      DE,(STEPBF+4) ;Load destination
                1123                                  ;address into DE.
02F0  ED52      1124            SBC     HL,DE   ;Compare HL and DE to
                1125                            ;determine move up or down.
02F2  300C      1126            JR      NC,MVUP
                1127                            ;Move down:
02F4  EB        1128            EX      DE,HL   ;HL = destination address
02F5  09        1129            ADD     HL,BC   ;HL = dest. address + length
02F6  2B        1130            DEC     HL      ;HL = end address of dest.
02F7  EB        1131            EX      DE,HL   ;DE = end address of dest.
02F8  2AB11F    1132            LD      HL,(STEPBF+2) ;HL = end address of source
02FB  EDB8      1133            LDDR            ;block transfer instruction
02FD  13        1134            INC     DE      ;DE = last address moved
02FE  181C      1135            JR      ENDFUN  ;Continue at ENDFUN.
                1136  MVUP:                     ;Move up:
0300  19        1137            ADD     HL,DE   ;HL is destroyed by
                1138                            ;SBC HL,DE.  Restore HL.
0301  EDB0      1139            LDIR            ;block transfer
0303  1B        1140            DEC     DE      ;DE = last address moved
0304  1816      1141            JR      ENDFUN  ;Continue at ENDFUN.
                1142  ;
                1143  ;*********************************************************
0306  ED5BAF1F  1144  GRL       LD      DE,(STEPBF) ;Load starting address
                1145                                ;into DE.
030A  13        1146            INC     DE      ;Increase this address by 2.
                1147                            ;Relative address is used in
                1148                            ;instruction JR or DJNZ.
                1149                            ;The codes for them are 2 bytes.
                1150                            ;The PC is increased by 2 after
                1151                            ;opcode is fetched.
030B  13        1152            INC     DE
030C  2AB11F    1153            LD      HL,(STEPBF+2) ;Load destionation
                1154                                  ;address into HL.
030F  B7        1155            OR      A
0310  ED52      1156            SBC     HL,DE   ;Calculate difference.
0312  7D        1157            LD      A,L     ;Check if the offset is between
                1158                            ;+127 (007FH) and -128 (FF80H).
                1159                            ;If the offset is positive, both H
                1160                            ;and bit 7 of L must be zero; if it
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                 1161                          ;is negative, H and bit 7 of L must
                 1162                          ;be FF and 1.  In both cases, adding
                 1163                          ;H with bit 7 of L results in 0.
0313   17        1164          RLA             ;Rotate bit 7 of L into carry flag.
0314   7C        1165          LD      A,H
0315   CE00      1166          ADC     A,0     ;ADD H and bit 7 of L.
0317   203A      1167          JR      NZ,ERROR ;Branch to ERROR if
                 1168                          ;the result is nonzero.
0319   7D        1169          LD      A,L
031A   1B        1170          DEC     DE
031B   12        1171          LD      (DE),A  ;Save the offset into
                 1172                          ;the next byte of opcode.
                 1173                          ;(DJNZ or JR)
                 1174  ;
                 1175  ENDFUN:
031C   ED53DE1F  1176          LD      (ADSAVE),DE ;Save DE into ADSAVE.
0320   CD0B04    1177          CALL    MEMDP2  ;Display this address and
                 1178                          ;its data.  Set STATE to 2.
0323   C9        1179          RET
                 1180  ;
                 1181  ;*************************************************************
                 1182  GWT:
0324   CD2D05    1183          CALL    SUM1    ;Load parameters from
                 1184                          ;step buffer into registers.
                 1185                          ;Check if the parameters
                 1186                          ;are legal.  If legal, calculate
                 1187                          ;the sum of all data to be output
                 1188                          ;to tape.
0327   382A      1189          JR      C,ERROR ;Branch to ERROR if the
                 1190                          ;parameters are illegal. (lenght is
                 1191                          ;negative)
0329   32B51F    1192          LD      (STEPBF+6),A ;Store the checksum into
                 1193                               ;STEPBF+6.
032C   21A00F    1194          LD      HL,4000 ;Output 1k Hz square
                 1195                          ;wave for 4000 cycles.
                 1196                          ;Leading sync. signal.
032F   CDDE05    1197          CALL    TONE1K
0332   21AF1F    1198          LD      HL,STEPBF ;Output 7 bytes starting
                 1199                            ;at STEPBF. (Include:
                 1200                            ;filename, starting, ending
                 1201                            ;address and checksum)
0335   010700    1202          LD      BC,7
0338   CDA705    1203          CALL    TAPEOUT
033B   21A00F    1204          LD      HL,4000 ;Output 2k Hz square
                 1205                          ;wave for 4000 cycles.
                 1206                          ;Middle sync.  The file name of the
                 1207                          ;file being read will be displayed
                 1208                          ;in this interval.
033E   CDE205    1209          CALL    TONE2K
0341   CD3A05    1210          CALL    GETPTR  ;Load parameters into
                 1211                          ;registers. (Starting, ending and
                 1212                          ;length).
0344   CDA705    1213          CALL    TAPEOUT ;Output user's data.
0347   21A00F    1214          LD      HL,4000 ;Output 4000 cycles of
                 1215                          ;2k Hz squire wave.
                 1216                          ;(Tail sync.)
034A   CDE205    1217          CALL    TONE2K
034D   ED5BB31F  1218  ENDTAPE LD      DE,(STEPBF+4) ;DE = last address
```

```
                                MPF-I
 LOC   OBJ CODE M STMT SOURCE STATEMENT

0351   18C9        1219          JR      ENDFUN  ;Continue at ENDFUN.
                   1220  ;
0353   DD21A907    1221  ERROR   LD      IX,ERR_ ;IX points to '-Err
0357   C3D000      1222          JP      SETST0  ;Set STATE to 0 by
                   1223                          ;branching to SETST0.
                   1224  ;
                   1225  ;*****************************************************
                   1226  GRT:
035A   2AAF1F      1227          LD      HL,(STEPBF) ;Temporarily save filename.
035D   22EA1F      1228          LD      (TEMP),HL
0360   3E40        1229  LEAD    LD      A,01000000B     ;decimal point
0362   D301        1230          OUT     (SEG7),A ;When searching for filename,
                   1231                          ;the display is blank  initially.
                   1232                          ;If the data read from MIC is
                   1233                          ;acceptable 0 or 1, the display
                   1234                          ;becomes '......'.
0364   21E803      1235          LD      HL,1000
0367   CD8C05      1236  LEAD1   CALL    PERIOD ;The return of PERIOD
                   1237                         ;is in flag:
                   1238                         ;  NC -- tape input is 1k Hz;
                   1239                         ;   C -- otherwise.
036A   38F4        1240          JR      C,LEAD ;Loop until leading sync.
                   1241                         ;is detected.
036C   2B          1242          DEC     HL      ;Decrease HL by one when
                   1243                          ;one period is detected.
036D   7C          1244          LD      A,H
036E   B5          1245          OR      L       ;Check if both H and L are 0.
036F   20F6        1246          JR      NZ,LEAD1 ;Wait for 1000 periods.
                   1247                          ;The leading sync. is accepted
                   1248                          ;if it is longer than 1000
                   1249                          ;cycles (1 second).
0371   CD8C05      1250  LEAD2   CALL    PERIOD
0374   30FB        1251          JR      NC,LEAD2 ;Wait all leading sync. to
                   1252                          ;pass over.
                   1253  ;
0376   21AF1F      1254          LD      HL,STEPBF ;Load 7 bytes from
                   1255                           ;tape into STEPBF.
0379   010700      1256          LD      BC,7
037C   CD4D05      1257          CALL    TAPEIN
037F   38DF        1258          JR      C,LEAD  ;Jump to LEAD if input
                   1259                          ;is not successful.
0381   ED5BAF1F    1260          LD      DE,(STEPBF) ;Get filename from
                   1261                            ;step buffer.
0385   CD6506      1262          CALL    ADDRDP  ;Convert it to display
                   1263                          ;format.
0388   0696        1264          LD      B,150   ;Display it for 1.5 sec.
038A   CD2406      1265  FILEDP  CALL    SCAN1
038D   10FB        1266          DJNZ    FILEDP
038F   2AEA1F      1267          LD      HL,(TEMP) ;Check if the input
                   1268                          ;filename equals to the
                   1269                          ;specified filename.
0392   B7          1270          OR      A
0393   ED52        1271          SBC     HL,DE
0395   20C9        1272          JR      NZ,LEAD ;If not, find the leading
                   1273                          ;sync. of next file.
                   1274
                   1275                          ;If filename is found,
0397   3E02        1276          LD      A,00000010B     ;segament '-'
```

```
                                MPF-I
 LOC    OBJ CODE M STMT SOURCE STATEMENT

0399    D301        1277            OUT     (SEG7),A ;Display '------'.
039B    CD3A05      1278            CALL    GETPTR  ;The parameters (starting
                    1279                            ;ending address and check-
                    1280                            ;sum) have been load into
                    1281                            ;STEPBF.  Load them into
                    1282                            ;registers, calculate the block
                    1283                            ;length and check if they are
                    1284                            ;legal.
039E    38B3        1285            JR      C,ERROR ;Jump to ERROR if the
                    1286                            ;parameters are illegal.
03A0    CD4D05      1287            CALL    TAPEIN  ;Input user's data.
03A3    38AE        1288            JR      C,ERROR ;Jump to ERROR if input
                    1289                            ;is not successful.
03A5    CD2D05      1290            CALL    SUM1    ;Calculate the sum of all
                    1291                            ;input data.
03A8    21B51F      1292            LD      HL,STEPBF+6
03AB    BE          1293            CP      (HL)    ;Compare it with the
                    1294                            ;checksum calculated by and stored
                    1295                            ;'WRtape'.
03AC    20A5        1296            JR      NZ,ERROR ;Jump to ERROR if not
                    1297                             ;matched.
03AE    189D        1298            JR      ENDTAPE ;Continue at ENDTAPE.
                    1299    ;
                    1300    ;*********************************************************
                    1301    BRANCH:
                    1302    ;Branch table format:
                    1303    ;   byte 1,2 : address of the 1st routine in
                    1304    ;              each group.
                    1305    ;   byte 3   : difference between the address
                    1306    ;              of 1st and 1st routine, which is
                    1307    ;              of course 0.
                    1308    ;   byte 4   : difference between the address
                    1309    ;              of 2nd and 1st routine
                    1310    ;   byte 5   : difference between the address
                    1311    ;              of 3rd and 1st routine
                    1312    ;    ...
                    1313    ;    ...
                    1314    ;    ...
                    1315    ; HL : address of branch table
                    1316    ; A  : the routine number in its group
                    1317    ; Such branch table can save table length and avoid page
                    1318    ; (256 bytes) boundary problem.
                    1319
03B0    5E          1320            LD      E,(HL)  ;Load the address of 1st
                    1321                            ;routine in the group into
                    1322                            ;DE register.
03B1    23          1323            INC     HL
03B2    56          1324            LD      D,(HL)
03B3    23          1325            INC     HL      ;Locate the pointer of difference
                    1326                            ;table.
03B4    85          1327            ADD     A,L
03B5    6F          1328            LD      L,A
03B6    6E          1329            LD      L,(HL)  ;Load the address
                    1330                            ;difference into L.
03B7    2600        1331            LD      H,0
03B9    19          1332            ADD     HL,DE   ;Get routine's real address
03BA    E9          1333            JP      (HL)    ;Jump to it.
                    1334
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

                1335  ;*********************************************************
                1336  IGNORE:
03BB   21E61F   1337          LD      HL,TEST
03BE   CBFE     1338          SET     7,(HL)  ;Routine SCAN will check bit
                1339                          ;7 of TEST.  If it is set,
                1340                          ;all LEDs will be disabled.
                1341                          ;This is a warning message to
                1342                          ;the user when a illegal key
                1343                          ;is entered.
03C0   C9       1344          RET
                1345  ;
                1346  ;*********************************************************
                1347  INI:
                1348  ; Power-up initialization.
03C1   DD21A507 1349          LD      IX,BLANK ;BLANK is the initial pattern
                1350
                1351                           ;Display the following
                1352                           ;patterns sequence, each 0.16
                1353                           ;seconds:
                1354                           ;     '      '
                1355                           ;     '     u'
                1356                           ;     '    uP'
                1357                           ;     '   uPF'
                1358                           ;     '  uPF-'
                1359                           ;     ' uPF--'
                1360                           ;     'uPF--1'
                1361
03C5   0E07     1362          LD      C,7     ;pattern count
03C7   0610     1363  INI1    LD      B,10H   ;Display 0.16 second.
03C9   CD2406   1364  INI2    CALL    SCAN1
03CC   10FB     1365          DJNZ    INI2
03CE   DD2B     1366          DEC     IX      ;next pattern
03D0   0D       1367          DEC     C
03D1   20F4     1368          JR      NZ,INI1
                1369  ;
03D3   3EA5     1370          LD      A,PWCODE
03D5   C3B306   1371          JP      INI3
03D8   216600   1372  INI4    LD      HL,NMI
03DB   22EE1F   1373          LD      (IM1AD),HL ;Set the service routine
                1374                             ;of RST 38H to NMI, which is the
                1375                             ;nonmaskable interrupt service
                1376                             ;routine for break point and
                1377                             ;single step.
                1378  CLRBR:
                1379  ; Clear break point by setting
                1380  ; the break point address to
                1381  ; FFFF.  This is a non-existant
                1382  ; address, so break can never
                1383  ; happen.
                1384
03DE   21FFFF   1385          LD      HL,0FFFFH
03E1   22E01F   1386          LD      (BRAD),HL
03E4   C9       1387          RET
                1388  ;
                1389  TESTM:
                1390  ; Check if the display is of 'address-data'
                1391  ; form, i.e. STATE 1 or 2.
                1392  ; The result is stored in zero flag.
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                1393 ;    Z:  yes
                1394 ;   NZ:  no
                1395
03E5  3AE41F    1396          LD      A,(STATE)
03E8  FE01      1397          CP      1
03EA  C8        1398          RET     Z
03EB  FE02      1399          CP      2
03ED  C9        1400          RET
                1401 ;
                1402 PRECL1:
                1403 ; Pre-clear 1 byte.
                1404 ; If bit 0 of TEST is not 0, load 0 into (HL).  Bit 0 of
                1405 ; TEST is cleared after check.
                1406 ; Only AF register are destroyed.
                1407
03EE  3AE61F    1408          LD      A,(TEST)
03F1  B7        1409          OR      A       ;Is bit 0 of TEST zero?
03F2  C8        1410          RET     Z
03F3  3E00      1411          LD      A,0
03F5  77        1412          LD      (HL),A  ;Clear (HL)
03F6  32E61F    1413          LD      (TEST),A ;Clear TEST too.
03F9  C9        1414          RET
                1415 ;
                1416 PRECL2:
                1417 ; Pre-clear 2 bytes.
                1418 ; If bit 0 of TEST is nonzero, clear (HL)
                1419 ; and (HL+1).
                1420 ; Only AF register are destroyed.
                1421
03FA  CDEE03    1422          CALL    PRECL1
03FD  C8        1423          RET     Z
03FE  23        1424          INC     HL
03FF  77        1425          LD      (HL),A
0400  2B        1426          DEC     HL
0401  C9        1427          RET
                1428 ;
                1429 ;*******************************************************
                1430 ; Memory display format: (address-data)
                1431
                1432 ;      i) A.A.A.A. D D -- State is AD.  four decimal points
                1433 ;                         under the address field indicate
                1434 ;                         that the numeric key entered will
                1435 ;                         be interpreted as memory address.
                1436 ;     ii) A A A A  D.D.-- State is DA.  Two decimal points
                1437 ;                         under the data field indicate
                1438 ;                         the monitor is expecting user to
                1439 ;                         enter memory data.
                1440 ;    iii) A.A.A.A. D.D.-- Six decimal points indicate the
                1441 ;                         address being displayed is set
                1442 ;                         as a break point.
                1443
                1444 MEMDP1:
0402  3E01      1445          LD      A,1     ;Next STATE =1
0404  0604      1446          LD      B,4     ;4 decimal points active
0406  21B81F    1447          LD      HL,DISPBF+2 ;The first active decimal
                1448                          ;point is in DISPBF+2, the
                1449                          ;last in DESPBF+5.
0409  1807      1450          JR      SAV12   ;Continue at SAV12.
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                1451  MEMDP2:
040B  3E02      1452          LD      A,2       ;Next STATE = 2
040D  0602      1453          LD      B,2       ;2 active decimal points
040F  21B61F    1454          LD      HL,DISPBF ;1st decimal point is in
                1455                            ;DISPBF, 2nd in DISPBF+1.
0412  32E41F    1456  SAV12   LD      (STATE),A ;Update STATE
0415  D9        1457          EXX               ;Save register HL,BC,DE.
0416  ED5BDE1F  1458          LD      DE,(ADSAVE) ;The address to be
                1459                              ;displayed is stored in
                1460                              ;(ADSAVE).  Load it into
                1461                              ;DE register.
041A  CD6506    1462          CALL    ADDRDP    ;Convert this address to
                1463                            ;display format and store it
                1464                            ;into DISPBF+2 ¢ DISPBF+5.
041D  1A        1465          LD      A,(DE)    ;Load the data of this
                1466                            ;address into A register.
041E  CD7106    1467          CALL    DATADP    ;Convert this data to
                1468                            ;display format and store it
                1469                            ;into DISPBF ¢ DISPBF+1.
                1470  BRTEST:
                1471  ; The next 3 instructions serve to refresh the
                1472  ; data at break address every time memory is
                1473  ; displayed.
0421  2AE01F    1474          LD      HL,(BRAD) ;Get break point address.
0424  7E        1475          LD      A,(HL)    ;Get the data of this
                1476                            ;address into A register.
0425  32E21F    1477          LD      (BRDA),A  ;Store it into BRDA (break data).
0428  B7        1478          OR      A
0429  ED52      1479          SBC     HL,DE     ;Check if the address to
                1480                            ;be displayed is break point.
042B  2006      1481          JR      NZ,SETPT1 ;If not, jump to SETPT1.
042D  0606      1482          LD      B,6       ;6 active decimal points.
042F  21B61F    1483          LD      HL,DISPBF ;1st decimal point is in
                1484                              ;DISPBF; 6th in DISPBF+5.
0432  D9        1485          EXX
0433  D9        1486  SETPT1  EXX               ;Restore HL,BC,DE.
0434  CBF6      1487  SETPT   SET     6,(HL)    ;Set decimal points.
                1488                            ;Count in B, first address
                1489                            ;in HL register.
0436  23        1490          INC     HL
0437  10FB      1491          DJNZ    SETPT
0439  C9        1492          RET
                1493  ;
                1494  ;*************************************************************
                1495  ; Step display format: (this format is used when user is
                1496  ; entering parameters for Move, Rela, WRtape, RDtape.)
                1497
                1498  ;          P.P.P.P. - N
                1499
                1500  ; 'P' is the digit of parameter.  Four decimal points
                1501  ; indicate P's are being modified now.  N is the mnemonic of
                1502  ; the parameter:
                1503  ;       i) Move    S -- starting address
                1504  ;                  E -- ending address
                1505  ;                  D -- destination address
                1506  ;      ii) Rela    S -- source address
                1507  ;                  D -- destination address
                1508  ;     iii) WRtape  F -- file name
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                 1509 ;                S -- starting address
                 1510 ;                E -- ending address
                 1511 ;     iv) RDtape F -- file name
                 1512 ;
                 1513 STEPDP:
                 1514 ;Display step buffer and its parameter name.
                 1515 ;Input: STATE
                 1516 ;       STMIONR (parameter count)
                 1517 ;register destroyed: AF,BC,DE,HL
                 1518
043A  CD5504     1519         CALL    LOCSTBF ;Get parameter address
043D  5E         1520         LD      E,(HL)  ;Load parameter into DE
043E  23         1521         INC     HL
043F  56         1522         LD      D,(HL)
0440  CD6506     1523         CALL    ADDRDP  ;Convert this parameter to
                 1524                         ;display format (4 digits)
                 1525                         ;and store it into DISPBF+2
                 1526                         ; ¢ DISPBF+5.
0443  21B81F     1527         LD      HL,DISPBF+2 ;Set 4 decimal points.
                 1528                             ;From DISPBF+2 to DISPBF+5.
0446  0604       1529         LD      B,4
0448  CD3404     1530         CALL    SETPT
044B  CD5F04     1531         CALL    LOCSTNA ;Get parameter name.
044E  6F         1532         LD      L,A
044F  2602       1533         LD      H,2      ;Pattern '-' for 2nd rightmost
                 1534                          ;digit.
0451  22B61F     1535         LD      (DISPBF),HL
0454  C9         1536         RET
                 1537 ;
                 1538 LOCSTBF:
                 1539 ;Get the location of parameter.
                 1540 ; address = STEPBF + STMINOR*2
                 1541 ;register destroyed: AF,HL
                 1542
0455  3AE31F     1543         LD      A,(STMINOR) ;Get parameter count.
0458  87         1544         ADD     A,A         ;Each parameter has 2 bytes.
0459  21AF1F     1545         LD      HL,STEPBF   ;Get base address.
045C  85         1546         ADD     A,L
045D  6F         1547         LD      L,A
045E  C9         1548         RET
                 1549 ;
                 1550 LOCSTNA:
                 1551  ;Get parameter name.
                 1552  ;Input: STATE, STMINOR
                 1553  ;Output: parameter name in A, and Z flag.
                 1554
                 1555  ;register destroyed: AF,DE
045F  3AE41F     1556         LD      A,(STATE) ;Get STATE.
                 1557                           ;Possible states are:
                 1558                           ;4,5,6,7. (Move, Rel,
                 1559                           ;WRtape, RDtape)
0462  D604       1560         SUB     4         ;Change 4,5,6,7 to
                 1561                           ;0,1,2,3.
0464  87         1562         ADD     A,A     ;Each state has 4 bytes for names.
0465  87         1563         ADD     A,A
0466  11BC07     1564         LD      DE,STEPTAB
0469  83         1565         ADD     A,E
046A  5F         1566         LD      E,A     ;Now, DE contains the
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                1567                            ;address of 1st name
                1568                            ;for each state.
046B  3AE31F    1569            LD      A,(STMINOR) ;Get parameter count
046E  83        1570            ADD     A,E         ;DE <--- DE + A
046F  5F        1571            LD      E,A
0470  1A        1572            LD      A,(DE)  ;Get parameter name.
0471  B7        1573            OR      A       ;Change zero flag.  If the
                1574                            ;returned pattern (in A) is
                1575                            ;zero,  the '+' or '-' must
                1576                            ;have been pressed beyond legal
                1577                            ; parameter boundary. (Check if
                1578                            ;parameter name got from STEPTAB
                1579                            ;is zero)
0472  C9        1580            RET
                1581    ;
                1582    ;*********************************************************
                1583    ; Register display format:
                1584
                1585    ;       i)  X X X X  Y Y -- State is REGAD. The numeric data
                1586    ;                           entered is interpreted as
                1587    ;                           register name.
                1588    ;                           YY is the register name, the
                1589    ;                           data of that register pair is
                1590    ;                           XXXX.
                1591
                1592    ;      ii)  X X X.X. Y Y  or
                1593    ;     iii)  X.X.X X  Y Y -- State is REGDA.  The unit of
                1594    ;                           register modification is byte.
                1595    ;                           The numeric data entered will
                1596    ;                           change the byte with decimal
                1597    ;                           points under it.  Decimal points
                1598    ;                           can be moved by '+' of '-' keys.
                1599
                1600    REGDP8:
                1601    ; Display register and set STATE to 8.
                1602
0473  3E08      1603            LD      A,8     ;Next state = 8
0475  1802      1604            JR      RGSTIN
                1605
                1606    REGDP9:
                1607    ; Display register and set STATE to 9.
                1608
0477  3E09      1609            LD      A,9     ;Next state = 9
                1610
                1611    RGSTIN:
                1612    ; Update STATE by register A.
                1613    ; Display user's register (count
                1614    ; contained in STMINOR).
                1615    ; register destroyed: AF,BC,DE,HL
                1616
0479  32E41F    1617            LD      (STATE),A ;Update STATE.
047C  3AE31F    1618            LD      A,(STMINOR) ;Get register count.
047F  CB87      1619            RES     0,A     ;Registers are displayed by
                1620                            ;pair.  Find the count
                1621                            ;of pair leader.  (count of
                1622                            ;the lower one)
0481  47        1623            LD      B,A     ;Temporarily save A.
0482  CDAE04    1624            CALL    RGNADP  ;Find register count.
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

                  1625                          ;Store them into DISPBF
                  1626                          ;and DISPBF+1.
0485   78         1627          LD      A,B     ;Restore A (register pair leader).
0486   CDBE04     1628          CALL    LOCRG   ;Get the address of
                  1629                          ;user's register.
0489   5E         1630          LD      E,(HL)  ;Get register data. (2 bytes)
048A   23         1631          INC     HL
048B   56         1632          LD      D,(HL)
048C   ED53DE1F   1633          LD      (ADSAVE),DE ;Convert them to display
                  1634                          ;format and store into
                  1635                          ;display buffer.
0490   CD6506     1636          CALL    ADDRDP
0493   3AE41F     1637          LD      A,(STATE)
0496   FE09       1638          CP      9       ;If STATE equals to 9 (RGDA),
                  1639                          ;set 2 decimal points.
                  1640                          ;Otherwise return here.
0498   C0         1641          RET     NZ
0499   21B81F     1642          LD      HL,DISPBF+2
049C   3AE31F     1643          LD      A,(STMINOR) ;Get register name.
049F   CB47       1644          BIT     0,A     ;If this register is
                  1645                          ;group leader, set decimal
                  1646                          ;points of two central digits.
                  1647                          ;Otherwise set two left digits.
04A1   2802       1648          JR      Z,LOCPT
04A3   23         1649          INC     HL
04A4   23         1650          INC     HL
04A5   CBF6       1651  LOCPT   SET     6,(HL)  ;Set decimal points of
                  1652                          ;(HL) and (HL+1)
04A7   23         1653          INC     HL
04A8   CBF6       1654          SET     6,(HL)
04AA   CDC404     1655          CALL    FCONV   ;Convert user's flag (F,F')
                  1656                          ;to binary display format.
04AD   C9         1657          RET
                  1658  ;
                  1659  RGNADP:
                  1660  ; Get the patterns of register names and
                  1661  ; store them into DISPBF and DISPBF+1.
                  1662  ; Input: A contains register count of
                  1663  ;        pair leader.
                  1664  ; register destroyed: AF,DE,HL
                  1665
04AE   21D007     1666          LD      HL,RGTAB ;Get address of pattern
                  1667                          ;table.
04B1   85         1668          ADD     A,L
04B2   6F         1669          LD      L,A
04B3   5E         1670          LD      E,(HL)  ;Get first pattern.
04B4   23         1671          INC     HL
04B5   56         1672          LD      D,(HL)  ;Get 2nd pattern.
04B6   ED53B61F   1673          LD      (DISPBF),DE
04BA   C9         1674          RET
                  1675  ;
                  1676  LOCRGBF:
                  1677  ; Get the address of user's register.
                  1678  ; Register name contained in STMINOR.
                  1679  ; Destroys HL, AF.
                  1680
04BB   3AE31F     1681          LD      A,(STMINOR)
04BE   21BC1F     1682  LOCRG   LD      HL,REGBF
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

04C1   85          1683          ADD     A,L
04C2   6F          1684          LD      L,A
04C3   C9          1685          RET
                   1686  ;
                   1687  FCONV:
                   1688  ; Encode or decode user's flag register.
                   1689  ; STMINOR contains the name of the flag
                   1690  ; being displayed now.
                   1691  ; register destroyed: AF,BC,HL.
                   1692
04C4   3AE31F      1693          LD      A,(STMINOR) ;Get register name.
04C7   B7          1694          OR      A           ;Clear carry flag.
04C8   1F          1695          RRA     ;name of I register: 17H,
                   1696                  ;name of IFF: 16H.
                   1697                  ;Rotate right one bit, both
                   1698                  ;become 0BH.
04C9   FE0B        1699          CP      0BH
04CB   2809        1700          JR      Z,FLAGX ;Jump to FLAGX if
                   1701                          ;I or IFF is being
                   1702                          ;displayed now.
04CD   4F          1703          LD      C,A     ;Otherwise, mask out bit
                   1704                          ;1 to bit 7 of user's IFF.
                   1705                          ;IFF is only 1 bit, monitor
                   1706                          ;use one byte to store it,
                   1707                          ;masking out bit 1¢7 is to
                   1708                          ;ignore the useless bits.
                   1709                          ;This is done only when the
                   1710                          ;user is not modifying IFF.
                   1711                          ;If user is modifying IFF,
                   1712                          ;monitor will display whatever
                   1713                          ;he enters, even if bit 1¢7
                   1714                          ;are not all zero.
                   1715                          ;A register is not changed
                   1716                          ;after doing this.
04CE   21D21F      1717          LD      HL,USERIF
04D1   7E          1718          LD      A,(HL)
04D2   E601        1719          AND     00000001B
04D4   77          1720          LD      (HL),A
04D5   79          1721          LD      A,C
04D6   FE0C        1722  FLAGX   CP      0CH     ;If STMINOR contains
                   1723                          ;the name of SZXH, XPNC,
                   1724                          ;SZXH' or XPNC', after
                   1725                          ;rotating right one bit
                   1726                          ;it will be greater than
                   1727                          ;or equal to 0CH.
                   1728                          ;Decode user's flag if it
                   1729                          ;is not being modified now,
                   1730                          ;encode it otherwise.
04D8   301F        1731          JR      NC,FCONV2
04DA   3ABC1F      1732  FCONV1  LD      A,(USERAF) ;Get user's F register.
04DD   CD1805      1733          CALL    DECODE  ;Decode upper 4 bits.
04E0   22D41F      1734          LD      (FLAGH),HL
04E3   CD1805      1735          CALL    DECODE  ;Decode lower 4 bits.
04E6   22D61F      1736          LD      (FLAGL),HL
04E9   3AC41F      1737          LD      A,(UAFP) ;Get user's F' register.
04EC   CD1805      1738          CALL    DECODE
04EF   22D81F      1739          LD      (FLAGHP),HL
04F2   CD1805      1740          CALL    DECODE
```

```
                        MPF-I
 LOC    OBJ CODE M STMT SOURCE STATEMENT

04F5    22DA1F      1741          LD      (FLAGLP),HL
04F8    C9          1742          RET
04F9    2AD41F      1743  FCONV2  LD      HL,(FLAGH) ;Get the binary form
                    1744                             ;of 4 upper bits of
                    1745                             ;user's F register.
04FC    CD2305      1746          CALL    ENCODE     ;Encode it.
04FF    2AD61F      1747          LD      HL,(FLAGL) ;Encode 4 lower bits.
0502    CD2305      1748          CALL    ENCODE
0505    32BC1F      1749          LD      (USERAF),A ;Save the encoded
                    1750                             ;result into USERAF.
0508    2AD81F      1751          LD      HL,(FLAGHP) ;Encode F' register.
050B    CD2305      1752          CALL    ENCODE
050E    2ADA1F      1753          LD      HL,(FLAGLP)
0511    CD2305      1754          CALL    ENCODE
0514    32C41F      1755          LD      (UAFP),A
0517    C9          1756          RET
                    1757  ;
                    1758  DECODE:
                    1759  ; Decode bit 7¢4 of A register.
                    1760  ; Each bit is extented to 4 bits.
                    1761  ; 0 becomes 0000, 1 becomes 0001.
                    1762  ; The output is stored in HL, which
                    1763  ; is 16 bits in length.  Also, after
                    1764  ; execution, bit 7¢4 of A register are
                    1765  ; bit 3¢0 of A before execution.
                    1766  ; Register AF,B,HL are destroyed.
                    1767
0518    0604        1768          LD      B,4        ;Loop 4 times.
051A    29          1769  DRL4    ADD     HL,HL      ;Clear rightmost 3
                    1770                             ;bits of HL.
051B    29          1771          ADD     HL,HL
051C    29          1772          ADD     HL,HL
051D    07          1773          RLCA
051E    ED6A        1774          ADC     HL,HL      ;The 4th bit of HL
                    1775                             ;is determined by carry
                    1776                             ;flag, which is the MSB
                    1777                             ;of A register.
0520    10F8        1778          DJNZ    DRL4
0522    C9          1779          RET
                    1780  ;
                    1781  ENCODE:
                    1782  ; Encode HL register.  Each 4 bits of HL
                    1783  ; are encoded to 1 bit. 0000 become 0,
                    1784  ; 0001 become 1.  The result is stored
                    1785  ; in bit 3¢0 of A register.  Also, after
                    1786  ; execution, bit 7¢4 of A are bit 3¢0
                    1787  ; before execution.
                    1788  ; Registers AF,B,HL are destroyed.
                    1789
0523    0604        1790          LD      B,4        ;Loop 4 times.
0525    29          1791  ERL4    ADD     HL,HL      ;Shift HL left 4 bits.
                    1792                             ;Bit 12 of HL will be
                    1793                             ;shifted into carry flag.
0526    29          1794          ADD     HL,HL
0527    29          1795          ADD     HL,HL
0528    29          1796          ADD     HL,HL
0529    17          1797          RLA                ;Rotate carry flag into
                    1798                             ;A register.
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

052A   10F9       1799          DJNZ    ERL4
052C   C9         1800          RET
                  1801  ;
                  1802  ;*********************************************************
                  1803  SUM1:
                  1804  ; Calculate the sum of the data in a memory
                  1805  ; block.  The starting and ending address
                  1806  ; of this block are stored in STEPBF+2 ¢ STEPBF+4.
                  1807  ;   Registers AF,BC,DE,HL are destroyed.
                  1808
052D   CD3A05     1809          CALL    GETPTR  ;Get parameters from
                  1810                          ;step buffer.
0530   D8         1811          RET     C       ;Return if the parameters
                  1812                          ;are illegal.
                  1813  SUM:
                  1814  ; Calculate the sum of a memory block.
                  1815  ; HL contains the starting address of
                  1816  ; this block, BC contains the length.
                  1817  ; The result is stored in A.  Registers
                  1818  ; AF,BC,HL are destroyed.
                  1819
0531   AF         1820          XOR     A       ;Clear A.
0532   86         1821  SUMCAL  ADD     A,(HL)  ;Add
0533   EDA1       1822          CPI
0535   EA3205     1823          JP      PE,SUMCAL
0538   B7         1824          OR      A       ;Clear flags.
0539   C9         1825          RET
                  1826  ;
                  1827  GETPTR:
                  1828  ; Get parameters from step buffer.
                  1829  ; Input: (STEPBF+2) and (STEPBF+3) contain
                  1830  ;        starting address.
                  1831  ;        (STEPBF+4) and (STEPBF+5) contain
                  1832  ;        ending address.
                  1833  ; Output: HL register contains the starting
                  1834  ;         address.
                  1835  ;          BC register contains the length.
                  1836  ;          Carry flay  0 -- BC positive
                  1837  ;                      1 -- BC negative
                  1838  ; Destroyed reg.: AF,BC,DE,HL.
                  1839
053A   21B11F     1840          LD      HL,STEPBF+2
053D   5E         1841  GETP    LD      E,(HL)  ;Load starting address
                  1842                          ;into DE.
053E   23         1843          INC     HL
053F   56         1844          LD      D,(HL)
0540   23         1845          INC     HL
0541   4E         1846          LD      C,(HL)
0542   23         1847          INC     HL      ;Load ending address
                  1848                          ;into HL.
0543   66         1849          LD      H,(HL)
0544   69         1850          LD      L,C
0545   B7         1851          OR      A       ;Clear carry flag.
0546   ED52       1852          SBC     HL,DE   ;Find difference.
                  1853                          ;Carry flag is changed here.
0548   4D         1854          LD      C,L
0549   44         1855          LD      B,H
054A   03         1856          INC     BC      ;Now BC contains the
```

```
                                MPF-I
 LOC   OBJ CODE M STMT SOURCE STATEMENT

                1857                            ;length.
054B   EB       1858            EX      DE,HL   ;Now HL contains the
                1859                            ;starting address.
054C   C9       1860            RET
                1861  ;
                1862  TAPEIN:
                1863  ; Load a memory block from tape.
                1864  ; Input: HL -- starting address of the block
                1865  ;        BC -- length of the block
                1866  ; Output: Carry flag,1 -- reading error
                1867  ;                  0 -- no error
                1868  ; Destroyed reg. -- AF,BC,DE,HL,AF',BC',DE',HL'
                1869
054D   AF       1870            XOR     A       ;Clear carry flag.
                1871                            ;At beginning, the reading is
                1872                            ;no error.
054E   08       1873            EX      AF,AF'
054F   CD5A05   1874  TLOOP    CALL    GETBYTE ;Read 1 byte from tape.
0552   73       1875            LD      (HL),E  ;Store it into memory.
0553   EDA1     1876            CPI
0555   EA4F05   1877            JP      PE,TLOOP ;Loop until length
                1878                            ;is zero.
0558   08       1879            EX      AF,AF'
0559   C9       1880            RET
                1881  ;
                1882  GETBYTE:
                1883  ; Read one byte from tape.
                1884  ; Output: E -- data read
                1885  ;         Carry of F',1 -- reading error
                1886  ;                    0 -- no error
                1887  ; Destroy reg. -- AF,DE,AF',BC',DE',HL'
                1888  ; Byte format:
                1889
                1890  ; start bit bit bit bit bit bit bit bit stop
                1891  ;  bit   0   1   2   3   4   5   6   7   bit
                1892
055A   CD6B05   1893            CALL    GETBIT  ;Get start bit.
055D   1608     1894            LD      D,8     ;Loop 8 times.
055F   CD6B05   1895  BLOOP    CALL    GETBIT  ;Get one data bit.
                1896                            ;Result in carry flag.
0562   CB1B     1897            RR      E       ;Rotate it into E.
0564   15       1898            DEC     D
0565   20F8     1899            JR      NZ,BLOOP
0567   CD6B05   1900            CALL    GETBIT  ;Get stop bit.
056A   C9       1901            RET
                1902  ;
                1903  ;
                1904  GETBIT:
                1905  ; Read one bit from tape.
                1906  ; Output: Carry of F,0 -- this bit is 0
                1907  ;                  1 -- this bit is 1
                1908  ;         Carry of F',1 -- reading error
                1909  ;                  0 -- no error
                1910  ; Destroyed reg. -- AF,AF',BC',DE',HL'
                1911  ; Bit format:
                1912
                1913  ;   0 -- 2K Hz 8 cycles + 1K Hz 2 cycles.
                1914  ;   1 -- 2K Hz 4 cycles + 1K Hz 4 cycles.
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                1915
056B  D9        1916          EXX     ;Save HL,BC,DE registers
                1917
                1918  ; The tape-bit format of both 0 and 1 are
                1919  ; of the same form:  high freq part + low freq part.
                1920  ; The difference between 0 and 1 is the
                1921  ; number high freq cycles and low freq
                1922  ; cycles.  Thus, a high freq period may has
                1923  ; two meanings:
                1924  ;  i) It is used to count the number of high
                1925  ;     freq cycles of the current tape-bit;
                1926  ; ii) If a high freq period is detected
                1927  ;     immediately after a low freq period, then
                1928  ;     this period is the first cycle of next
                1929  ;     tape-bit and is used as a terminator of the
                1930  ;     last tape-bit.
                1931
                1932  ; Bit 0 of H register is used to indicate the usage
                1933  ; of a high freq period.  If this bit is zero, high
                1934  ; freq period causes counter increment for the current
                1935  ; tape-bit.  If the high freq part has passed, bit 0
                1936  ; of H is set and the next high freq period will be used
                1937  ; as a terminator.
                1938  ; L register is used to up/down count the number of periods.
                1939  ; when a high freq period is read, L is increased by
                1940  ; 1; when a low freq period is read, L is decreased
                1941  ; by 2. (The time duration for each count is 0.5 ms.)
                1942  ; At the end of a tape-bit, positive and negative L
                1943  ; stand for 0 and 1 respectively.
                1944
056C  210000    1945          LD      HL,0    ;Clear bit 0 of H,
                1946                          ;Set L to 0.
056F  CD8C05    1947  COUNT   CALL    PERIOD  ;Read one period.
0572  14        1948          INC     D       ;The next 2 instructions
                1949                          ;check if D is zero.  Carry
                1950                          ;flag is not affected.
0573  15        1951          DEC     D
0574  2011      1952          JR      NZ,TERR ;If D is not zero, jump
                1953                          ;to error routine TERR.
                1954                          ;(Because the period is too
                1955                          ;much longer than that of 1K Hz.)
0576  3806      1956          JR      C,SHORTP ;If the period is short
                1957                          ;(2K Hz), jump to SHORTP.
0578  2D        1958          DEC     L       ;The period is 1K Hz,
                1959                          ;decrease L by 2.  And set
                1960                          ;bit 0 of H to indicate this
                1961                          ;tape-bit has passed high freq
                1962                          ;part and reaches its low freq part.
0579  2D        1963          DEC     L
057A  CBC4      1964          SET     0,H
057C  18F1      1965          JR      COUNT
057E  2C        1966  SHORTP  INC     L       ;The period is 2 K Hz,
                1967                          ;increase L by 1.
057F  CB44      1968          BIT     0,H     ;If the tape-bit has passed
                1969                          ;its high freq part, high frquency
                1970                          ;means this bit is all over and
                1971                          ;next bit has started.
0581  28EC      1972          JR      Z,COUNT
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                 1973                    ;L = (# of 2K period) - 2*(# of 1K period)
0583  CB15       1974          RL        L
                 1975                            ; 0 --- NCarry (L positive)
                 1976                            ; 1 ---  Carry (L negative)
                 1977                            ;The positive or negative sign of
                 1978                            ;L corresponds to the tape-bit data.
                 1979                            ;'RL  L' will shift the sign bit of
                 1980                            ;L into carry flag.  After this
                 1981                            ;instruction, the carry flag
                 1982                            ;contains the tape-bit.
0585  D9         1983          EXX       ;Restore BC',DE',HL'
0586  C9         1984          RET
0587  08         1985  TERR    EX        AF,AF'
0588  37         1986          SCF       ;Set carry flag of F' to indicate error.
0589  08         1987          EX        AF,AF'
058A  D9         1988          EXX
058B  C9         1989          RET
                 1990  ;
                 1991  PERIOD:
                 1992  ; Wait the tape to pass one period.
                 1993  ; The time duration is stored in DE.  The
                 1994  ; unit is loop count.  Typical value for
                 1995  ; 2K Hz is 28, for 1K Hz is 56.
                 1996  ; Use (56+28)/2 as threshold.  The returned
                 1997  ; result is in carry flag. (1K -- NC, 2K -- C)
                 1998  ; Register AF and DE are destroyed.
                 1999
058C  110000     2000          LD        DE,0
058F  DB00       2001  LOOPH   IN        A,(KIN) ;Bit 7 of port A is Tapein.
0591  13         2002          INC       DE
0592  17         2003          RLA
0593  38FA       2004          JR        C,LOOPH ;Loop until input goes low.
0595  3EFF       2005          LD        A,11111111B ;Echo the tape input to
                 2006                              ;speaker on  MPF-I.
0597  D302       2007          OUT       (DIGIT),A
0599  DB00       2008  LOOPL   IN        A,(KIN)
059B  13         2009          INC       DE
059C  17         2010          RLA
059D  30FA       2011          JR        NC,LOOPL ;Loop until input goes high.
059F  3E7F       2012          LD        A,01111111B ;Echo the tape input to
                 2013                              ;speaker on MPF-I.
05A1  D302       2014          OUT       (DIGIT),A
05A3  7B         2015          LD        A,E       ;Compare the result with
                 2016                              ;the threshold.
05A4  FE2A       2017          CP        MPERIOD
05A6  C9         2018          RET
                 2019  ;
                 2020  ;*******************************************************
                 2021  TAPEOUT:
                 2022  ; Output a memory block to tape.
                 2023  ; Input: HL -- starting address of the block
                 2024  ;        BC -- length of the block
                 2025  ; Destroyed reg. -- AF,BC,DE,HL,BC',DE',HL'
                 2026
05A7  5E         2027          LD        E,(HL)  ;Get the data.
05A8  CDB105     2028          CALL      OUTBYTE ;Output to tape.
05AB  EDA1       2029          CPI
05AD  EAA705     2030          JP        PE,TAPEOUT ;Loop until finished.
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

05B0   C9          2031          RET
                   2032  ;
                   2033  OUTBYTE:
                   2034  ; Output one byte to tape.  For tape-byte
                   2035  ; format, see comments on GETBYTE.
                   2036  ; Input: E -- data
                   2037  ; Destroyed reg. -- AF,DE,BC',DE',HL'
                   2038
05B1   1608        2039          LD      D,8     ;Loop 8 times.
05B3   B7          2040          OR      A       ;Clear carry flag.
05B4   CDC405      2041          CALL    OUTBIT  ;Output start bit.
05B7   CB1B        2042  OLOOP   RR      E       ;Rotate data into carry
05B9   CDC405      2043          CALL    OUTBIT  ;Output the carry
05BC   15          2044          DEC     D
05BD   20F8        2045          JR      NZ,OLOOP
05BF   37          2046          SCF             ;Set carry flag.
05C0   CDC405      2047          CALL    OUTBIT  ;Output stop bit
05C3   C9          2048          RET
                   2049  ;
                   2050  OUTBIT:
                   2051  ; Output one bit to tape.
                   2052  ; Input: data in carry flag.
                   2053  ; Destroyed reg. -- AF,BC',DE',HL'
05C4   D9          2054          EXX             ;Save BC,DE,HL.
05C5   2600        2055          LD      H,0
05C7   3809        2056          JR      C,OUT1  ;If data=1, output 1.
                   2057  OUT0:   ;2K 8 cycles, 1K 2 cycles.
05C9   2E08        2058          LD      L,ZERO_2K
05CB   CDE205      2059          CALL    TONE2K
05CE   2E02        2060          LD      L,ZERO_1K
05D0   1807        2061          JR      BITEND
                   2062  ;
                   2063  OUT1:   ;2K 4 cycles, 1K 4 cycles.
05D2   2E04        2064          LD      L,ONE_2K
05D4   CDE205      2065          CALL    TONE2K
05D7   2E04        2066          LD      L,ONE_1K
05D9   CDDE05      2067  BITEND  CALL    TONE1K
05DC   D9          2068          EXX             ;Restore registers.
05DD   C9          2069          RET
                   2070  ;
                   2071  ;*************************************************************
                   2072  ;
                   2073  ;         UTILITY SUBROUTINE
                   2074  ;
                   2075  ;*************************************************************
                   2076  ;
                   2077  ; Function: Generate square wave to the MIC & speaker
                   2078  ;           on MPF--I
                   2079  ; Input :  C -- period = 2*(44+13*C) clock states.
                   2080  ;         HL -- number of periods.
                   2081  ; Output: none.
                   2082  ; Destroyed reg.: AF, B(C), DE, HL.
                   2083  ; Call: none.
                   2084
                   2085  TONE1K:
05DE   0E41        2086          LD      C,F1KHZ
05E0   1802        2087          JR      TONE
                   2088  TONE2K:
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

05E2   0E1F        2089          LD      C,F2KHZ
                   2090  TONE:                           ;Half period: 44+13*C states
05E4   29          2091          ADD     HL;HL           ;Double for half-cycle count
05E5   110100      2092          LD      DE,1
05E8   3EFF        2093          LD      A,0FFH
05EA   D302        2094  SQWAVE  OUT     (DIGIT),A       ;Bit-7 tapeout
05EC   41          2095          LD      B,C
05ED   10FE        2096          DJNZ    $               ;Half period delay
05EF   EE80        2097          XOR     80H             ;Toggle output
05F1   ED52        2098          SBC     HL,DE           ;Decrement one count
05F3   20F5        2099          JR      NZ,SQWAVE
05F5   C9          2100          RET
                   2101  ;
                   2102  ;*************************************************************
                   2103  ; Function: check if a memory address is in RAM.
                   2104  ; Input: HL -- address to be check.
                   2105  ; Output: Zero flag -- 0, ROM or nonexistant;
                   2106  ;                      1, RAM.
                   2107  ; Destroyed reg.: AF.
                   2108  ; Call: none
                   2109
                   2110  RAMCHK:
05F6   7E          2111          LD      A,(HL)
05F7   2F          2112          CPL
05F8   77          2113          LD      (HL),A
05F9   7E          2114          LD      A,(HL)
05FA   2F          2115          CPL
05FB   77          2116          LD      (HL),A
05FC   BE          2117          CP      (HL)
05FD   C9          2118          RET
                   2119  ;
                   2120  ;*************************************************************
                   2121  ; Function: Scan the keyboard and display.  Loop until
                   2122  ;           a key is detected.  If the some key is already
                   2123  ;           pressed when this routine starts execution,
                   2124  ;           return when next key is entered.
                   2125  ; Input: IX points to the buffer contains display patterns.
                   2126  ;        6 LEDs require 6 byte data.  (IX) contains the
                   2127  ;        pattern for rightmost LED, (IX+5) contains the
                   2128  ;        pattern for leftmost LED.
                   2129  ; Output: internal code of the key pressed.
                   2130  ; Destroyed reg. : AF, B, HL, AF', BC', DE'.
                   2131  ;                   All other registers except IY are also
                   2132  ;                   changed during execution, but they are
                   2133  ;                   restored before return.
                   2134  ; Call: SCAN1
                   2135
                   2136  SCAN:
05FE   DDE5        2137          PUSH    IX      ;Save IX.
0600   21E61F      2138          LD      HL,TEST
0603   CB7E        2139          BIT     7,(HL)  ;This bit is sert if the use
                   2140                          ;has entered illegal key.  The
                   2141                          ;display will be disabled as
                   2142                          ;a warning to the user.  This
                   2143                          ;is done by replacing the display
                   2144                          ;buffer pointer IX by BLANK.
0605   2804        2145          JR      Z,SCPRE
0607   DD21A507    2146          LD      IX,BLANK
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                2147
                2148  ; Wait until all keys are released for 40 ms.
                2149  ; (The execution time of SCAN1 is 10 ms,
                2150  ; 40 = 10 * 4.)
                2151
060B  0604      2152  SCPRE   LD      B,4
060D  CD2406    2153  SCNX    CALL    SCAN1
0610  30F9      2154          JR      NC,SCPRE ;If any key is pressed, re-load
                2155                           ;the debounce counter B by 4.
0612  10F9      2156          DJNZ    SCNX
0614  CBBE      2157          RES     7,(HL)   ;Clear error-flag.
0616  DDE1      2158          POP     IX       ;Restore original IX.
                2159
                2160  ; Loop until any key is pressed.
                2161
0618  CD2406    2162  SCLOOP  CALL    SCAN1
061B  38FB      2163          JR      C,SCLOOP
                2164
                2165  ; Convert the key-position-code returned by SCAN1 to
                2166  ; key-internal-code.  This is done by table-lookup.
                2167  ; The table used is KEYTAB.
                2168
061D  217B07    2169  KEYMAP  LD      HL,KEYTAB
0620  85        2170          ADD     A,L
0621  6F        2171          LD      L,A
0622  7E        2172          LD      A,(HL)
0623  C9        2173          RET
                2174  ;
                2175  ;*************************************************************
                2176  ; Function: Scan keyboard and display one cycle.
                2177  ;           Total execution time is about 10 ms (exactly
                2178  ;           9.95 ms, 17812 clock states @ 1.79 MHz).
                2179  ; Input: Same as SCAN.
                2180  ; Output:  i) no key during one scan
                2181  ;                  Carry flay -- 1
                2182  ;         ii) key pressed during one scan
                2183  ;                  Carry flag -- 0,
                2184  ;                  A -- position code of the key pressed.
                2185  ;                       If more than one key is pressed, A
                2186  ;                       contains the largest position-code.
                2187  ;                       (This key is the last key scanned.)
                2188  ; Destroyed reg: AF, AF', BC', DE'. (see comments on SCAN)
                2189  ; Call: none.
                2190
                2191  SCAN1:
                2192  ;In hardware, the display and keyboard are
                2193  ;arranged as a 6 by 6 matrix.  Each cloumn
                2194  ;corresponds to one LED and six key buttons.
                2195  ;In normal operation, at most one column is
                2196  ;active.  The pattern of the active LED is the
                2197  ;data output on port C of 8255 I.  The data input
                2198  ;from bit 0¢5 of port A are the status of key
                2199  ;buttons in the active column.  All signals on
                2200  ;I/O port are active low.
                2201
0624  37        2202          SCF             ;Set carry flag.
0625  08        2203          EX      AF,AF'
0626  D9        2204          EXX
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

                2205
                2206  ;Carry flag of F' is used to return the status of
                2207  ;the keyboard.  If any key is pressed during one
                2208  ;scan, the flag is reset; otherwise, it is set.
                2209  ;Initially, this flag is set.  A' register is used
                2210  ;to store the position-code of the key pressed.
                2211  ;In this routine, 36 key positions are checked one
                2212  ;by one.  C register contains the code of the key
                2213  ;being checked.  The value of C is 0 at the beginning,
                2214  ;and is increased by 1 after each check.  So the code
                2215  ;ranges from 0 to 23H (total 36 positions).  On each
                2216  ;check, if the input bit is 0 (key pressed), C register
                2217  ;is copied into A'.  The carry flag of F' is set also.
                2218  ;When some key is detected, the key positions after
                2219  ;this key will still be checked.  So if more than
                2220  ;one key are pressed during one scan, the code of the
                2221  ;last one will be returned.
                2222
0627  0E00      2223          LD      C,0       ;Initial position code
0629  1EC1      2224          LD      E,11000001B ;Scan from rightmost digit.
062B  2606      2225          LD      H,6
                2226                            ;to the active columm.
062D  7B        2227  KCOL    LD      A,E
062E  D302      2228          OUT     (DIGIT),A       ;Activate one columm.
0630  DD7E00    2229          LD      A,(IX)
0633  D301      2230          OUT     (SEG7),A
0635  06C9      2231          LD      B,COLDEL
0637  10FE      2232          DJNZ    $         ;Delay 1.5 ms per digit.
0639  AF        2233          XOR     A         ;Deactivate all display segments
063A  D301      2234          OUT     (SEG7),A
063C  7B        2235          LD      A,E
063D  2F        2236          CPL
063E  F6C0      2237          OR      11000000B
0640  D302      2238          OUT     (DIGIT),A
0642  0606      2239          LD      B,6       ;Each column has 6 keys.
0644  DB00      2240          IN      A,(KIN)   ;Now, bit 0c5 of A contain
                2241                            ;the status of the 6 keys
                2242                            ;in the active column.
0646  57        2243          LD      D,A       ;Store A into D.
0647  CB1A      2244  KROW    RR      D         ;Rotate D 1 bit right, bit 0
                2245                            ; of D will be rotated into
                2246                            ;carry flag.
0649  3802      2247          JR      C,NOKEY   ;Skip next 2 instructions
                2248                            ;if the key is not pressed.
                2249                            ;The next 2 instructions
                2250                            ;store the current position-code
                2251                            ;into A' and reset carry flag
                2252                            ;of F' register.
064B  79        2253          LD      A,C       ;Key-in, get key position.
064C  08        2254          EX      AF,AF'    ;Save A & Carry in AF'.
064D  0C        2255  NOKEY   INC     C         ;Increase current key-code by 1.
064E  10F7      2256          DJNZ    KROW      ;Loop until 6 keys in the
                2257                            ;active colums are all checked.
0650  DD23      2258          INC     IX
0652  7B        2259          LD      A,E
0653  E63F      2260          AND     00111111B
0655  CB07      2261          RLC     A
0657  F6C0      2262          OR      11000000B
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

0659   5F          2263          LD      E,A
065A   25          2264          DEC     H
065B   20D0        2265          JR      NZ,KCOL
065D   11FAFF      2266          LD      DE,-6
0660   DD19        2267          ADD     IX,DE    ;Get original IX.
0662   D9          2268          EXX
0663   08          2269          EX      AF,AF'
0664   C9          2270          RET
                   2271  ;
                   2272  ;*************************************************************
                   2273  ; Function: Convert the 2 byte data stored in DE to
                   2274  ;           7-segament display format.  The output is stored
                   2275  ;           in the address field of DISPBF (display buffer),
                   2276  ;           most significiant digit in DISPBF+5.
                   2277  ;           This routine is usually used by monitor only.
                   2278  ; Destroyed reg: AF, HL.
                   2279  ; Call: HEX7SG
                   2280
                   2281  ADDRDP:
0665   21B81F      2282          LD      HL,DISPBF+2
0668   7B          2283          LD      A,E
0669   CD7806      2284          CALL    HEX7SG
066C   7A          2285          LD      A,D
066D   CD7806      2286          CALL    HEX7SG
0670   C9          2287          RET
                   2288  ;
                   2289  ;*************************************************************
                   2290  ; Function: Convert the data stored in A to 7-segament
                   2291  ;           display format.  1 byte is converted to 2
                   2292  ;           digits.  The result is stored in the data
                   2293  ;           field of display buffer (DISPBF).
                   2294  ;           This routine is usually used by monitor only.
                   2295  ; Destroyed reg: AF, HL.
                   2296  ; Call: HEX7SG
                   2297
                   2298  DATADP:
0671   21B61F      2299          LD      HL,DISPBF
0674   CD7806      2300          CALL    HEX7SG
0677   C9          2301          RET
                   2302  ;
                   2303  ;*************************************************************
                   2304  ; Function: Convert binary data to 7-segament display
                   2305  ;           format.
                   2306  ; Input: 1 byte in A register.
                   2307  ;        HL points to the result buffer.
                   2308  ; Output: Pattern for 2 digits.  Low order digit in (HL),
                   2309  ;         high order digit in (HL+1).
                   2310  ;         HL becomes HL+2.
                   2311  ; Destory reg: AF, HL.
                   2312  ; Call: HEX7
                   2313
                   2314  HEX7SG:
0678   F5          2315          PUSH    AF
0679   CD8906      2316          CALL    HEX7
067C   77          2317          LD      (HL),A
067D   23          2318          INC     HL
067E   F1          2319          POP     AF
067F   0F          2320          RRCA
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

0680   0F          2321          RRCA
0681   0F          2322          RRCA
0682   0F          2323          RRCA
0683   CD8906      2324          CALL    HEX7
0686   77          2325          LD      (HL),A
0687   23          2326          INC     HL
0688   C9          2327          RET
                   2328  ;
                   2329  ;*************************************************************
                   2330  ; Function: Convert binary data to 7-segament display
                   2331  ;           format.
                   2332  ; Input: A -- LSB 4 bits contains the binary data
                   2333  ; Output: A -- display pattern for 1 digit.
                   2334  ; Destroyed reg: AF
                   2335  ; Call: none
                   2336
                   2337  HEX7:
0689   E5          2338          PUSH    HL
068A   21F007      2339          LD      HL,SEGTAB
068D   E60F        2340          AND     0FH
068F   85          2341          ADD     A,L
0690   6F          2342          LD      L,A
0691   7E          2343          LD      A,(HL)
0692   E1          2344          POP     HL
0693   C9          2345          RET
                   2346  ;
                   2347  ;
                   2348  ;*************************************************************
                   2349  ; Function: RAM 1800-1FFF self-check.
                   2350  ; Input: none
                   2351  ; Output: none
                   2352  ; Destroyed reg: AF, BC, HL
                   2353  ; Call: RAMCHK
                   2354
                   2355  RAMTEST:
0694   210018      2356          LD      HL,1800H
0697   010008      2357          LD      BC,800H
069A   CDF605      2358  RAMT    CALL    RAMCHK
069D   2801        2359          JR      Z,TNEXT
069F   76          2360          HALT            ;If error.
06A0   EDA1        2361  TNEXT   CPI
06A2   EA9A06      2362          JP      PE,RAMT
06A5   C7          2363          RST     0       ;Display 'uPF--1'.
                   2364  ;
                   2365  ;*************************************************************
                   2366  ;Monitor ROM self-check.  Add the data of address
                   2367  ;0000 ¢ 0800.  If the sum equals to 0.  Reset the monitor
                   2368  ;and display 'uPF--1'.  If the sum is not 0, which
                   2369  ;indicates error, HALT.
                   2370  ;Input: none.
                   2371  ;Output: none.
                   2372  ;Destroyed registers: AF, BC, HL.
                   2373  ;Call: SUM.
                   2374
                   2375  ROMTEST:
06A6   210000      2376          LD      HL,0
06A9   010008      2377          LD      BC,800H
06AC   CD3105      2378          CALL    SUM
```

```
                                MPF-I
 LOC   OBJ CODE M STMT SOURCE STATEMENT

06AF   2801        2379          JR      Z,SUMOK
06B1   76          2380          HALT            ;If error.
06B2   C7          2381  SUMOK   RST     0       ;Display 'uPF--1'.
06B3   32E51F      2382  INI3    LD      (POWERUP);A ;Load power-code into
                   2383                            ;(POWERUP).  The monitor
                   2384                            ;uses the location to decide
                   2385                            ;whether a reset signal is
                   2386                            ;on power-up.
06B6   3E55        2387          LD      A,55H
06B8   32F01F      2388          LD      (BEEPSET),A
06BB   3E44        2389          LD      A,44H
06BD   32F11F      2390          LD      (FBEEP),A   ;Beep frequency when key is
                   2391                            ;pressed.
06C0   21F21F      2392          LD      HL,TBEEP
06C3   362F        2393          LD      (HL),2FH    ;Time duration of beep when
06C5   23          2394          INC     HL
06C6   3600        2395          LD      (HL),0
                   2396                            ;key is pressed.
06C8   C3D803      2397          JP      INI4
                   2398
06CB   F5          2399  BEEP    PUSH    AF
06CC   21F11F      2400          LD      HL,FBEEP
06CF   4E          2401          LD      C,(HL)
06D0   2AF21F      2402          LD      HL,(TBEEP)
06D3   3AF01F      2403          LD      A,(BEEPSET)
06D6   FE55        2404          CP      55H
06D8   2003        2405          JR      NZ,NOTONE   ;There is no beep sound when
                   2406                            ;the key is pressed if data
                   2407                            ;of (BEEPSET) is not 55H
06DA   CDE405      2408          CALL    TONE
                   2409  NOTONE:
06DD   F1          2410          POP     AF
06DE   C3E900      2411          JP      KEYEXEC ;After a key is detected,determine
                   2412                          ;what action should the monitor take.
                   2413                          ;KEYEXEC uses the next 3 factors
                   2414                          ;to get the entry point of proper
                   2415                          ;service routine :key-code, STATE
                   2416                          ;and STMINOR (Minor-State).
                   2417  ; Below are the branch tables for each key and
                   2418  ; state.  The first entry of each table is
                   2419  ; a base address, other entrys are the offset to
                   2420  ; this address.  Offset is only one byte long,
                   2421  ; which is much shorter than the 2-byte address.
                   2422  ; This can save the monitor code space.
                   2423
0737               2424  KSUBFUN ORG     0737H
0737   1B01        2425          DEFW    KINC
0739   00          2426          DEFB    -KINC+KINC
073A   05          2427          DEFB    -KINC+KDEC
073B   0A          2428          DEFB    -KINC+KGO
073C   0F          2429          DEFB    -KINC+KSTEP
073D   1A          2430          DEFB    -KINC+KDATA
073E   2C          2431          DEFB    -KINC+KSBR
073F   42          2432          DEFB    -KINC+KINS
0740   7B          2433          DEFB    -KINC+KDEL
0741   C201        2434  KFUN    DEFW    KPC
0743   00          2435          DEFB    -KPC+KPC
0744   1C          2436          DEFB    -KPC+KADDR
```

```
                                    MPF-I
 LOC   OBJ CODE M STMT SOURCE STATEMENT

0745   0A          2437           DEFB    -KPC+KCBR
0746   14          2438           DEFB    -KPC+KREG
0747   20          2439           DEFB    -KPC+KMV
0748   20          2440           DEFB    -KPC+KRL
0749   26          2441           DEFB    -KPC+KWT
074A   26          2442           DEFB    -KPC+KRT
074B   EC01        2443  HTAB     DEFW    HFIX
074D   00          2444           DEFB    -HFIX+HFIX
074E   16          2445           DEFB    -HFIX+HAD
074F   03          2446           DEFB    -HFIX+HDA
0750   26          2447           DEFB    -HFIX+HRGFIX
0751   34          2448           DEFB    -HFIX+HMV
0752   34          2449           DEFB    -HFIX+HRL
0753   34          2450           DEFB    -HFIX+HWT
0754   34          2451           DEFB    -HFIX+HRT
0755   26          2452           DEFB    -HFIX+HRGAD
0756   44          2453           DEFB    -HFIX+HRGDA
0757   3D02        2454  ITAB     DEFW    IFIX
0759   00          2455           DEFB    -IFIX+IFIX
075A   03          2456           DEFB    -IFIX+IAD
075B   03          2457           DEFB    -IFIX+IDA
075C   00          2458           DEFB    -IFIX+IRGFIX
075D   0E          2459           DEFB    -IFIX+IMV
075E   0E          2460           DEFB    -IFIX+IRL
075F   0E          2461           DEFB    -IFIX+IWT
0760   0E          2462           DEFB    -IFIX+IRT
0761   1F          2463           DEFB    -IFIX+IRGAD
0762   1F          2464           DEFB    -IFIX+IRGDA
0763   6B02        2465  DTAB     DEFW    DFIX
0765   00          2466           DEFB    -DFIX+DFIX
0766   03          2467           DEFB    -DFIX+DAD
0767   03          2468           DEFB    -DFIX+DDA
0768   00          2469           DEFB    -DFIX+DRGFIX
0769   0E          2470           DEFB    -DFIX+DMV
076A   0E          2471           DEFB    -DFIX+DRL
076B   0E          2472           DEFB    -DFIX+DWT
076C   0E          2473           DEFB    -DFIX+DRT
076D   1F          2474           DEFB    -DFIX+DRGAD
076E   1F          2475           DEFB    -DFIX+DRGDA
076F   9902        2476  GTAB     DEFW    GFIX
0771   00          2477           DEFB    -GFIX+GFIX
0772   03          2478           DEFB    -GFIX+GAD
0773   03          2479           DEFB    -GFIX+GDA
0774   00          2480           DEFB    -GFIX+GRGFIX
0775   4B          2481           DEFB    -GFIX+GMV
0776   6D          2482           DEFB    -GFIX+GRL
0777   8B          2483           DEFB    -GFIX+GWT
0778   C1          2484           DEFB    -GFIX+GRT
0779   00          2485           DEFB    -GFIX+GRGAD
077A   00          2486           DEFB    -GFIX+GRGDA
               2487
               2488  ; Key-position-code to key-internal-code conversion table.
               2489
               2490  KEYTAB:
077B   03          2491  K0       DEFB    03H     ;HEX_3
077C   07          2492  K1       DEFB    07H     ;HEX_7
077D   0B          2493  K2       DEFB    0BH     ;HEX_B
077E   0F          2494  K3       DEFB    0FH     ;HEX_F
```

```
LOC   OBJ CODE M STMT SOURCE STATEMENT

077F  20         2495  K4       DEFB    20H     ;NOT USED
0780  21         2496  K5       DEFB    21H     ;NOT USED
0781  02         2497  K6       DEFB    02H     ;HEX_2
0782  06         2498  K7       DEFB    06H     ;HEX_6
0783  0A         2499  K8       DEFB    0AH     ;HEX_A
0784  0E         2500  K9       DEFB    0EH     ;HEX_E
0785  22         2501  K0A      DEFB    22H     ;NOT USED
0786  23         2502  K0B      DEFB    23H     ;NOT USED
0787  01         2503  K0C      DEFB    01H     ;HEX_1
0788  05         2504  K0D      DEFB    05H     ;HEX_5
0789  09         2505  K0E      DEFB    09H     ;HEX_9
078A  0D         2506  K0F      DEFB    0DH     ;HEX_D
078B  13         2507  K10      DEFB    13H     ;STEP
078C  1F         2508  K11      DEFB    1FH     ;TAPERD
078D  00         2509  K12      DEFB    00H     ;HEX_0
078E  04         2510  K13      DEFB    04H     ;HEX_4
078F  08         2511  K14      DEFB    08H     ;HEX_8
0790  0C         2512  K15      DEFB    0CH     ;HEX_C
0791  12         2513  K16      DEFB    12H     ;GO
0792  1E         2514  K17      DEFB    1EH     ;TAPEWR
0793  1A         2515  K18      DEFB    1AH     ;CBR
0794  18         2516  K19      DEFB    18H     ;PC
0795  1B         2517  K1A      DEFB    1BH     ;REG
0796  19         2518  K1B      DEFB    19H     ;ADDR
0797  17         2519  K1C      DEFB    17H     ;DEL
0798  1D         2520  K1D      DEFB    1DH     ;RELA
0799  15         2521  K1E      DEFB    15H     ;SBR
079A  11         2522  K1F      DEFB    11H     ;-
079B  14         2523  K20      DEFB    14H     ;DATA
079C  10         2524  K21      DEFB    10H     ;+
079D  16         2525  K22      DEFB    16H     ;INS
079E  1C         2526  K23      DEFB    1CH     ;MOVE
                 2527  ;
                 2528  ;
                 2529  ;
                 2530  ;
079F  30         2531  MPF_I    DEFB    030H    ;'1'
07A0  02         2532           DEFB    002H    ;'-'
07A1  02         2533           DEFB    002H    ;'-'
07A2  0F         2534           DEFB    0FH     ;'F'
07A3  1F         2535           DEFB    1FH     ;'P'
07A4  A1         2536           DEFB    0A1H    ;'u'
07A5  00         2537  BLANK    DEFB    0
07A6  00         2538           DEFB    0
07A7  00         2539           DEFB    0
07A8  00         2540           DEFB    0
07A9  00         2541  ERR_     DEFB    0
07AA  00         2542           DEFB    0
07AB  03         2543           DEFB    3       ;'R'
07AC  03         2544           DEFB    3       ;'R'
07AD  8F         2545           DEFB    8FH     ;'E'
07AE  02         2546           DEFB    2       ;'-'
07AF  1F         2547  SYS_SP   DEFB    1FH     ;'P'
07B0  AE         2548           DEFB    0AEH    ;'S'
07B1  02         2549           DEFB    02H     ;'-'
07B2  AE         2550           DEFB    0AEH    ;'S'
07B3  B6         2551           DEFB    0B6H    ;'Y'
07B4  AE         2552           DEFB    0AEH    ;'S'
```

```
 LOC   OBJ CODE M STMT SOURCE STATEMENT

07B5   1F          2553  ERR_SP  DEFB    1FH     ;'P'
07B6   AE          2554          DEFB    0AEH    ;'S'
07B7   02          2555          DEFB    02      ;'-'
07B8   03          2556          DEFB    03      ;'R'
07B9   03          2557          DEFB    03      ;'R'
07BA   8F          2558          DEFB    8FH     ;'E'
07BB   00          2559          DEFB    0
07BC   AE          2560  STEPTAB DEFB    0AEH    ;'S'
07BD   8F          2561          DEFB    08FH    ;'E'
07BE   B3          2562          DEFB    0B3H    ;'D'
07BF   00          2563          DEFB    0
07C0   AE          2564          DEFB    0AEH    ;'S'
07C1   B3          2565          DEFB    0B3H    ;'D'
07C2   00          2566          DEFB    0
07C3   00          2567          DEFB    0
07C4   0F          2568          DEFB    0FH     ;'F'
07C5   AE          2569          DEFB    0AEH    ;'S'
07C6   8F          2570          DEFB    08FH    ;'E'
07C7   00          2571          DEFB    0
07C8   0F          2572          DEFB    0FH     ;'F'
07C9   00          2573          DEFB    0
07CA   00          2574  REG_    DEFB    0
07CB   00          2575          DEFB    0
07CC   02          2576          DEFB    02H     ;'-'
07CD   BE          2577          DEFB    0BEH    ;'G'
07CE   8F          2578          DEFB    08FH    ;'E'
07CF   03          2579          DEFB    03H     ;'R'
07D0   0F3F        2580  RGTAB   DEFW    3F0FH   ;'AF'
07D2   8DA7        2581          DEFW    0A78DH  ;'BC'
07D4   8FB3        2582          DEFW    0B38FH  ;'DE'
07D6   8537        2583          DEFW    3785H   ;'HL'
07D8   4F3F        2584          DEFW    3F4FH   ;'AF.'
07DA   CDA7        2585          DEFW    0A7CDH  ;'BC.'
07DC   CFB3        2586          DEFW    0B3CFH  ;'DE.'
07DE   C537        2587          DEFW    37C5H   ;'HL.'
07E0   0730        2588          DEFW    3007H   ;'IX'
07E2   B630        2589          DEFW    30B6H   ;'IY'
07E4   1FAE        2590          DEFW    0AE1FH  ;'SP'
07E6   0F30        2591          DEFW    300FH   ;'IF'
07E8   370F        2592          DEFW    0F37H   ;'FH'
07EA   850F        2593          DEFW    0F85H   ;'FL'
07EC   770F        2594          DEFW    0F77H   ;'FH.'
07EE   C50F        2595          DEFW    0FC5H   ;'FL.'
07F0   BD          2596  SEGTAB  DEFB    0BDH    ;'0'
07F1   30          2597          DEFB    30H     ;'1'
07F2   9B          2598          DEFB    09BH    ;'2'
07F3   BA          2599          DEFB    0BAH    ;'3'
07F4   36          2600          DEFB    36H     ;'4'
07F5   AE          2601          DEFB    0AEH    ;'5'
07F6   AF          2602          DEFB    0AFH    ;'6'
07F7   38          2603          DEFB    38H     ;'7'
07F8   BF          2604          DEFB    0BFH    ;'8'
07F9   BE          2605          DEFB    0BEH    ;'9'
07FA   3F          2606          DEFB    3FH     ;'A'
07FB   A7          2607          DEFB    0A7H    ;'B'
07FC   8D          2608          DEFB    08DH    ;'C'
07FD   B3          2609          DEFB    0B3H    ;'D'
07FE   8F          2610          DEFB    08FH    ;'E'
```

```
                                MPF-I
 LOC   OBJ CODE M STMT SOURCE STATEMENT

07FF   0F          2611          DEFB    0FH     ;'F'
                   2612  ;
                   2613  ;*************************************************************
                   2614  ;SYSTEM RAM AREA:
1F9F               2615  USERSTK ORG     1F9FH
1F9F               2616          DEFS    16
1FAF               2617  SYSSTK: ORG     1FAFH
1FAF               2618  STEPBF  DEFS    7
1FB6               2619  DISPBF  DEFS    6
                   2620  REGBF:
1FBC               2621  USERAF  DEFS    2
1FBE               2622  USERBC  DEFS    2
1FC0               2623  USERDE  DEFS    2
1FC2               2624  USERHL  DEFS    2
1FC4               2625  UAFP    DEFS    2
1FC6               2626  UBCP    DEFS    2
1FC8               2627  UDEP    DEFS    2
1FCA               2628  UHLP    DEFS    2
1FCC               2629  USERIX  DEFS    2
1FCE               2630  USERIY  DEFS    2
1FD0               2631  USERSP  DEFS    2
1FD2               2632  USERIF  DEFS    2
1FD4               2633  FLAGH   DEFS    2
1FD6               2634  FLAGL   DEFS    2
1FD8               2635  FLAGHP  DEFS    2
1FDA               2636  FLAGLP  DEFS    2
1FDC               2637  USERPC  DEFS    2
                   2638  ;
1FDE               2639  ADSAVE  DEFS    2       ;Contains the address being
                   2640                          ;displayed now.
1FE0               2641  BRAD    DEFS    2       ;Break point address
1FE2               2642  BRDA    DEFS    1       ;Data of break point address
1FE3               2643  STMINOR DEFS    1       ;Minor state
1FE4               2644  STATE   DEFS    1       ;State
1FE5               2645  POWERUP DEFS    1       ;Power-up initialization
1FE6               2646  TEST    DEFS    1       ;Flag, bit 0 -- set when function
                   2647                          ;      or subfunction key is hit.
                   2648                          ;      bit 7 -- set when illegal key
                   2649                          ;            is entered.
1FE7               2650  ATEMP   DEFS    1       ;Temporary storage
1FE8               2651  HLTEMP  DEFS    2       ;Temporary storage
1FEA               2652  TEMP    DEFS    4       ;See comments on routine GDA.
1FEE               2653  IM1AD   DEFS    2       ;Contains the address of Opcode 'FF'
                   2654                          ;service routine. (RST 38H, mode
                   2655                          ;1 interrupt, etc.)
1FF0               2656  BEEPSET DEFS    1       ;Default value is 55H
1FF1               2657  FBEEP   DEFS    1       ;Beep frequency
1FF2               2658  TBEEP   DEFS    2       ;Time duration of beep
                   2659          END
```

```
CROSS REFERENCE                MPF-I
SYMBOL   VAL  M DEFN REFS

ADDRDP 0665    2281 1262 1462 1523 1636
ADSAVE 1FDE    2639  283  596  618  681  699  732  745  793  822  839
                      914  918  969  973 1077 1176 1458 1633
ATEMP  1FE7    2650  221  238  274  280
BEEP   06CB    2399  384
BEEPSE 1FF0    2656 2388 2403
BITEND 05D9    2067 2061
BLANK  07A5    2537 1349 2146
BLOOP  055F    1895 1899
BR1    0115     514  525  535  545
BRAD   1FE0    2641  229  364  602 1024 1386 1474
BRANCH 03B0    1301  431  455  515
BRDA   1FE2    2642  230  362 1477
BRRSTO 00D4     362  351
BRTEST 0421    1470
CLRBR  03DE    1378  742
COLDEL 00C9      27 2231
CONT28 003E     217  163
COUNT  056F    1947 1965 1972
DAD    026E     968 2467
DATADP 0671    2298 1467
DDA    026E     969 2468
DECODE 0518    1758 1733 1735 1738 1740
DFIX   026B     963 2465 2466 2467 2468 2469 2470 2471 2472 2473
                    2474 2475
DIGIT  0002      16  115  237  279 1091 2007 2014 2094 2230 2238
DISPBF 1FB6    2619  348  441  858 1447 1454 1483 1527 1535 1642 1673
                    2282 2299
DMV    0279     981 2470
DOMV   0187     672  724
DRGAD  028A     995 2474
DRGDA  028A     996 2475
DRGFIX 026B     964 2469
DRGNA  0295    1006 1004
DRL    0279     980 2471
DRL4   051A    1769 1778
DRT    0279     978 2473
DSTEP  0286     992  989
DTAB   0763    2465  534
DWT    0279     979 2472
EIDI   02B2    1048 1046
ENCODE 0523    1781 1746 1748 1752 1754
ENDFUN 031C    1175 1135 1141 1219
ENDTAP 034D    1218 1298
ERL4   0525    1791 1799
ERROR  0353    1221 1119 1167 1189 1285 1288 1296
ERR    07A9    2541 1221
ERR_SP 07B5    2553  329
F1KHZ  0041      29 2086
F2KHZ  001F      31 2089
FBEEP  1FF1    2657 2390 2400
FCONV  04C4    1687  756 1655
FCONV1 04DA    1732
FCONV2 04F9    1743 1731
FILEDP 038A    1265 1266
FLAGH  1FD4    2633 1734 1743
FLAGHP 1FD8    2635 1739 1751
FLAGL  1FD6    2634 1736 1747
```

```
CROSS REFERENCE                    MPF-I
SYMBOL   VAL  M DEFN REFS

FLAGLP 1FDA    2636 1741 1753
FLAGX  04D6    1722 1700
GAD    029C    1023 2478
GDA    029C    1024 2479
GETBIT 056B    1904 1893 1895 1900
GETBYT 055A    1882 1874
GETP   053D    1841 1113
GETPTR 053A    1827 1210 1278 1809
GFIX   0299    1017 2476 2477 2477 2478 2479 2480 2481 2482 2483 2484
                    2485 2486
GMV    02E4    1112  672 2481
GRGAD  0299    1019 2485
GRGDA  0299    1020 2486
GRGFIX 0299    1018 2480
GRL    0306    1144 2482
GRT    035A    1226 2484
GTAB   076F    2476  544
GWT    0324    1182 2483
HAD    0202     839 2445
HDA    01EF     822 2446
HEX7   0689    2337 2316 2324
HEX7SG 0678    2314 2284 2286 2300
HFIX   01EC     816 2443 2444 2444 2445 2446 2447 2448 2449 2450 2451
                    2452 2453
HLTEMP 1FE8    2651  162  239  281  287
HMV    0220     871 2448
HRGAD  0212     855 2452
HRGDA  0230     888 2453
HRGFIX 0212     856 2447
HRL    0220     870 2449
HRT    0220     868 2451
HTAB   074B    2443  513
HWT    0220     869 2450
IAD    0240     913 2456
IDA    0240     914 2457
IFIX   023D     908 2454 2455 2455 2456 2457 2458 2459 2460 2461 2462
                    2463 2464
IGNORE 03BB    1336  557  580  594  600  616  630  643  692  708  719
                     816  825  910  936  965  991 1020
IM1AD  1FEE    2653  207 1373
IMV    024B     926 2459
INI    03C1    1347  123
INI1   03C7    1363 1368
INI2   03C9    1364 1365
INI3   06B3    2382 1371
INI4   03D8    1372 2397
IRGAD  025C     941 2463
IRGDA  025C     942 2464
IRGFIX 023D     909 2458
IRGNA  0267     952  950
IRL    024B     925 2460
IRT    024B     923 2462
ISTEP  0258     937  932
ITAB   0757    2454  524
IWT    024B     924 2461
K0     077B    2491
K0A    0785    2501
K0B    0786    2502
```

```
CROSS REFERENCE                    MPF-I
SYMBOL   VAL M DEFN REFS

K0C     0787    2503
K0D     0788    2504
K0E     0789    2505
K0F     078A    2506
K1      077C    2492
K10     078B    2507
K11     078C    2508
K12     078D    2509
K13     078E    2510
K14     078F    2511
K15     0790    2512
K16     0791    2513
K17     0792    2514
K18     0793    2515
K19     0794    2516
K1A     0795    2517
K1B     0796    2518
K1C     0797    2519
K1D     0798    2520
K1E     0799    2521
K1F     079A    2522
K2      077D    2493
K20     079B    2523
K21     079C    2524
K22     079D    2525
K23     079E    2526
K3      077E    2494
K4      077F    2495
K5      0780    2496
K6      0781    2497
K7      0782    2498
K8      0783    2499
K9      0784    2500
KADDR   01DE     764 2436
KCBR    01CC     738 2437
KCOL    062D    2227 2265
KDATA   0135     565 2430
KDEC    0120     528 2427
KDEL    0196     686 2433
KEYEXE  00E9     392 2411
KEYMAP  061D    2169
KEYTAB  077B    2490 2169
KFUN    0741    2434  453
KGO     0125     538 2428
KHEX    0111     507  403
KIN     0000      18 2001 2008 2240
KINC    011B     518 2425 2426 2426 2427 2428 2429 2430 2431 2432 2433
KINS    015D     610 2432
KMV     01E2     787 2439
KPC     01C2     727 2434 2435 2435 2436 2437 2438 2439 2440 2441 2442
KREG    01D6     750 2438
KRL     01E2     790 2440
KROW    0647    2244 2256
KRT     01E8     801 2442
KSBR    0147     587 2431
KSTEP   012A     548 2429
KSUBFU  0737    2424  430
KWT     01E8     797 2441
```

```
CROSS REFERENCE                    MPF-I
SYMBOL   VAL M DEFN REFS

LEAD    0360    1229 1240 1258 1272
LEAD1   0367    1236 1246
LEAD2   0371    1250 1251
LOCPT   04A5    1651 1648
LOCRG   04BE    1682 1628
LOCRGB  04BB    1676  888
LOCSTB  0455    1538  871 1519
LOCSTN  045F    1550  930  988 1531
LOOPH   058F    2001 2004
LOOPL   0599    2008 2011
MAIN    00DE     379  387
MEMDP1  0402    1444  768  851
MEMDP2  040B    1451  371  574  604  682  733  746  835  919  974 1177
MPERIO  002A      33 2017
MPF_I   079F    2531  258
MVUP    0300    1136 1126
NMI     0066     266  174 1372
NOKEY   064D    2255 2247
NOTONE  06DD    2409 2405
OLOOP   05B7    2042 2045
ONE_1K  0004      47 2066
ONE_2K  0004      48 2064
OUT0    05C9    2057
OUT1    05D2    2063 2056
OUTBIT  05C4    2050 2041 2043 2047
OUTBYT  05B1    2033 2028
P8255   0003      15  107  276
PERIOD  058C    1991 1236 1250 1947
POWERU  1FE5    2645  121 2382
PRECL1  03EE    1402  827  890 1422
PRECL2  03FA    1416  840  875
PREOUT  02A3    1036  562
PREPC   0021     133  131
PWCODE  00A5      19  122 1370
RAMCHK  05F6    2110  130  331  334  598  628  707  824 2358
RAMT    069A    2358 2362
RAMTES  0694    2355
REGBF   1FBC    2620 1049 1682
REGDP8  0473    1600  864
REGDP9  0477    1606  582  897  952 1006
REG     07CA    2574  753
RESET1  0032     181  140
RESET2  0054     248  183
RGNADP  04AE    1659 1624
RGSAVE  0074     281
RGSTIN  0479    1611 1604
RGTAB   07D0    2580 1666
ROMTES  06A6    2375
RST28   0028     143
RST30   0030     166
RST38   0038     194
SAV12   0412    1456 1450
SCAN    05FE    2136  381
SCAN1   0624    2191 1265 1364 2153 2162
SCLOOP  0618    2162 2163
SCNX    060D    2153 2156
SCPRE   060B    2152 2145 2154
SEG7    0001      17 1230 1277 2228 2234
```

```
CROSS REFERENCE                    MPF-I
SYMBOL   VAL M DEFN REFS

SEGTAB  07F0    2596 2339
SETIF   00A4     320  318
SETPT   0434    1487 1491 1530
SETPT1  0433    1486 1481
SETST0  00D0     353  263  332  335  347 1222
SHORTP  057E    1966 1956
SKIPH1  0183     645  641
SKIPH2  01B8     721  717
SQWAVE  05EA    2094 2099
STATE   1FE4    2644  361  443  514 1396 1456 1556 1617 1637
STEPBF  1FAF    2618  623  627  645  680  706  721  723  796 1112 1122
                     1132 1144 1153 1192 1198 1218 1227 1254 1260 1292
                     1545 1840
STEPDP  043A    1513  805  885  937  992
STEPTA  07BC    2560 1564
STMINO  1FE3    2643  451  859  926  942  981  996 1543 1569 1618 1643
                     1681 1693
SUM     0531    1813 2378
SUM1    052D    1803 1183 1290
SUMCAL  0532    1821 1823
SUMOK   06B2    2381 2379
SYSSTK  1FAF    2617  116  322  380
SYS_SP  07AF    2547  341
TAPEIN  054D    1862 1257 1287
TAPEOU  05A7    2021 1203 1213 2030
TBEEP   1FF2    2658 2392 2402
TEMP    1FEA    2652 1036 1048 1079 1094 1228 1267
TERR    0587    1985 1952
TEST    1FE6    2646  252  413 1337 1408 1413 2138
TESTM   03E5    1389  552  569  592  614  690
TESTRG  013E     578  571
TLOOP   054F    1874 1877
TNEXT   06A0    2361 2359
TONE    05E4    2090 2087 2408
TONE1K  05DE    2085 1197 2067
TONE2K  05E2    2088 1209 1217 2059 2065
UAFP    1FC4    2625 1737 1755
UBCP    1FC6    2626
UDEP    1FC8    2627
UHLP    1FCA    2628
USERAF  1FBC    2621 1068 1092 1732 1749
USERBC  1FBE    2622
USERDE  1FC0    2623
USERHL  1FC2    2624
USERIF  1FD2    2632  181  312  320 1037 1069 1717
USERIX  1FCC    2629
USERIY  1FCE    2630  289
USERPC  1FDC    2637  133  285  731
USERSP  1FD0    2631  250  288  328 1067
USERST  1F9F    2615  249  345
ZERO_1  0002      49 2060
ZERO_2  0008      50 2058
ZSUM_   0071      20  191
```