```
                        ;------------------------------------------------------------
                        ; C8085.ASM
                        ; monitor program for MTK-85 8085 MICROPROCESSOT TRAINING KIT
                        ; COPYTIGHT (C) 2007-2018 BY WICHIT SIRICHOTE, wichit.sirichote@gmai
                        ;
                        ; source file was assembled with C32 Cross Assembler V3.0
                        ;
                        ; 18 May 2007 add insert byte, ALT E
                        ;                delete byte, ALT D
                        ;                click sound when key pressed
                        ;
                        ; 8 March 2015 remove repeat key
                        ;              modified address and data entry mode
                        ; 3 April 2016 replace buzzer with small speaker for tone experiment
                        ;              add beep/no beep with ALT F press
                        ; 16 April 2016 add delay after no beep

                        ; 18 May 2018  fixed RST 7, service routine, save CPU HL register to

                        ; 11 July 2021 skip NVRAM booting
                        ;
                        ;------------------------------------------------------------

0000                            CPU     "8085.TBL"    ;CPU TABLE
0000                            HOF     "INT8"        ;HEX FORMAT


                        ; ---------- onboard GPIO ----------------------------------------

0000 =                  gpio    equ 0 ; D0-D3 is 4-bit output port, D4-D7 is 4-bit input port

                        ;---------- 8255 PPI system port I/O address ----------------------

0010 =                  system_port_a:  equ 10h
0011 =                  system_port_b:  equ 11h
0012 =                  system_port_c:  equ 12h
0013 =                  system_port_control: equ 13h

                        ;-------------------- 8254 counter/timer -------------------------

0020 =                  counter0_8254  equ 20h
0021 =                  counter1_8254  equ 21h
0022 =                  counter2_8254  equ 22h
0023 =                  control_8254 equ 23h

0034 =                  control_word_8254 equ 00110100B  ; mode 0, counter0

                        ;---------- 8255 PPI user port I/O address ----------------------

0030 =                  user_port_a:  equ 30h
0031 =                  user_port_b:  equ 31h
0032 =                  user_port_c:  equ 32h
0033 =                  user_port_control: equ 33h


                        ;---------- 16C550 compatible UART I/O address -------------------
                        ; e.g., UM8250B, 16C450, 16C550

0040 =                  uart_buffer: equ 40h
0045 =                  uart_line_status: equ 45h
0042 =                  uart_fifo:        equ 42h
0043 =                  uart_lcr:         equ 43h
0040 =                  uart_divisor_lsb: equ 40h
0041 =                  uart_divisor_msb: equ 41h
```

```
0047 =          uart_scr:         equ 47h


                ;------------------- onboard LCD registers -----------------------

0050 =          command_write  equ 50h
0052 =          command_read   equ 52h
0051 =          data_write     equ 51h
0053 =          data_read      equ 53h
0080 =          busy           equ 80h

0009 =          TAB    EQU     9             ; ASCII TAB
0000 =          RS     EQU     0               ; terminator

000D =          cr: equ 0dh
000A =          lf: equ 0ah
0020 =          sp: equ 20h


F000 =          system_ram equ 0f000h

                ;system_stack equ 0ffffh

8100 =          home_address  equ 8100h

0000 =          rom      equ  0 ;8000h           ; change to 8000 for testing under R
                                                 ; change to 0000 for rom programming

8000 =          my_rom   equ  8000h

0000                     ORG rom
0000 C30001              JMP START           ; reset vector

0008                     ORG rom+8           ; RST 1  opcode is CF
0008 C36B02              jmp monitor_call

0010                     ORG rom+10h         ; RST 2  used for testing RST 7
                        ;  jmp service_rst2

0010 C31080             jmp my_rom+10h

0018                     ORG rom+18h         ; DF RST 3 for testing monitor call func
0018 C31880              jmp my_rom+18h

                         ; jmp monitor_call

0020                     ORG rom+20h         ; RST 4
0020 C32080              jmp my_rom+20h

0024                     ORG rom+24h
                        ; jmp my_rom+24h
0024 C3ED02              jmp service_trap    ; sing step running service routine

0028                     ORG rom+28h         ; RST 5
0028 C32880              jmp my_rom+28h

002C                     ORG rom+2ch         ; relocate RST5.5 to external ram
002C C32C80              jmp my_rom+2ch

0030                     ORG rom+30h         ; relocate RST 6
0030 C33080              jmp my_rom+30h

0034                     ORG rom+34h         ; relocate RST6.5 to external ram
0034 C33480              jmp my_rom+34h
```

```
0038                    ORG ROM+38H
0038 C39D02             jmp service_rst7    ; RST 7 service jump back to monitor

003C                    ORG rom+3ch         ; relocate RST7.5 to external ram
003C C33C80             jmp my_rom+3ch


0100                    ORG rom+100h


0100 F3         START   di
0101 3E90               MVI A,90H
0103 D313               OUT system_port_control

0105 3EF0               mvi a,0f0h      ; disable trap
0107 D312               out system_port_c

0109 3179F0       lxi sp,system_stack+32  ; point to top of system stack
010C 2199F0             lxi h,user_stack+32     ; point to top of user stack
010F 2234F0             shld user_SP



0112 CD1612             call init_uart

0115 DB52               in command_read
0117 E680               ani 80h
0119 C23101             jnz  skip_lcd

011C CD1202       call init_lcd
011F 21A41D             lxi h,prompt2
0122 CD2402             call put_str_lcd
0125 210100             lxi h,01
0128 CD3402             call goto_xy
012B 21B51D             lxi h,text3
012E CD2402             call put_str_lcd

0131            skip_lcd
0131 CD9901             call init_8254

0134 C34B01       jmp skip_boot

                  ; NVRAM booting
                  ; if location 8000H has C3 opcode then jump to 8000H
                  ;       if user press USER1 with RESET put 00 to 8000H
                  ;       ans skip booting

0137 DB10               in system_port_a
0139 E680               ani 80h
013B CA4B01             jz skip_boot

013E 3A0080             lda 8000H
0141 FEC3               cpi 0c3h
0143 C24B01             jnz skip_boot
0146 210080             lxi h,8000h
0149 E5                 push h
014A C9                 ret      ; jump to NVRAM


014B AF         skip_boot: xra a        ; write 00 to 8000H
014C 320080             sta 8000H
014F 3227F0       sta counter1   ; clear counter1

0152 3A29F0             lda warm_code
0155 FE24               cpi "$"
```

```
196     0157 CA6E01             jz  skip_cold_boot
197
198     015A 3E24         mvi a,"$"
199     015C 3229F0             sta warm_code
200
201     015F CD3D06             call test_buzzer
202
203     0162 3EFF               mvi a,0ffh
204     0164 D300               out gpio       ; make GPIO LED on
205
206     0166 CD860B             call cold_boot
207
208     0169 3E00         mvi a, 0
209     016B 3224F0       sta beep_flag
210
211
212     016E            skip_cold_boot:
213
214
215     016E 210081             lxi h,home_address
216     0171 222AF0             shld user_PC
217     0174 223CF0             shld pointer
218
219     0177 3A25F0             lda uart_found
220     017A FE00               cpi 0
221     017C CA8501             jz skip_send_prompt
222
223     017F CDBC12             call send_prompt1
224
225     0182 CD4A0D             call send_prompt
226
227
228     0185            skip_send_prompt:
229
230     0185 AF                 xra a
231     0186 D300               out gpio         ; turn LED off
232
233     0188 3E00               mvi a,0
234     018A 3226F0             sta entry_mode         ; set data entry mode
235     018D CDCE0A             call read_memory
236
237     0190 CD8E0C     main:   call scan_key
238     0193 CD9105             call key_execute
239     0196 F29001             jp main
240
241
242                     ;---------------- initialize counter0 for RST7.5 interrupt ----------
243
244     0199 3E34       init_8254:  mvi a, control_word_8254
245     019B D323                   out control_8254
246     019D AF                     xra a
247     019E D320                   out counter0_8254
248     01A0 D320                   out counter0_8254
249     01A2 C9                     ret
250
251                     ; convert 8-bit unsigned in A to ASCII string in line_buffer
252                     ; entry: A
253
254     01A3            bin2ascii:
255     01A3 1E00                   mvi e,0
256
257     01A5 FE64       bin1        cpi 100
258     01A7 DAB001                 jc  bin2
259     01AA D664                   sui 100
260     01AC 1C                     inr e
```

```
01AD C3A501               jmp bin1

01B0 57        bin2:      mov d,a
01B1 7B                   mov a,e
01B2 C630                 adi "0"
01B4 3247F0               sta line_buffer
01B7 7A                   mov a,d
01B8 1E00                 mvi e,0

01BA FE0A      bin3:      cpi 10
01BC DAC501               jc  bin4
01BF D60A                 sui 10
01C1 1C                   inr e
01C2 C3BA01               jmp bin3

01C5 57        bin4:      mov d,a
01C6 7B                   mov a,e
01C7 C630                 adi "0"
01C9 3248F0               sta line_buffer+1
01CC 7A                   mov a,d
01CD C630                 adi "0"
01CF 3249F0               sta line_buffer+2
01D2 C9                   ret


                ; print 8-bit unsigned decimal to terminal
                ; entry: A

01D3 CDA301    pint8u:    call bin2ascii
01D6 3A47F0               lda line_buffer
01D9 FE30                 cpi "0"
01DB CAEE01               jz  pint1
01DE CD3F12               call cout
01E1 3A48F0               lda line_buffer+1
01E4 CD3F12               call cout
01E7 3A49F0               lda line_buffer+2
01EA CD3F12               call cout
01ED C9                   ret

01EE 3A48F0    pint1:     lda line_buffer+1
01F1 FE30                 cpi "0"
01F3 CAF901               jz  pint2
01F6 CD3F12               call cout

01F9 3A49F0    pint2:     lda line_buffer+2
01FC CD3F12               call cout
01FF C9                   ret


                ; convert 16-bit unsigned integer to ASCII code stored in line_buffe
                ; entry: HL









                ;---------------------- LCD driver routines -------------------------

0200 F5        lcd_ready:  push psw

```

```
0201 DB52      lcd_ready1: in command_read
0203 E680                  ani 80h
0205 C20102                jnz  lcd_ready1  ; wait until lcd ready
0208 F1                    pop psw

0209 C9                    ret

020A CD0002    clear_lcd:  call lcd_ready
020D 3E01                  mvi a,1
020F D350                  out command_write
0211 C9        exit_clear: ret

0212 CD0002    init_lcd: call lcd_ready
0215 3E38                mvi a,38h
0217 D350                out command_write
0219 CD0002              call lcd_ready
021C 3E0C                mvi a, 0ch
021E D350                out command_write
0220 CD0A02              call clear_lcd

0223 C9                  ret

                 ; print ASCII text on LCD
                 ; entry: HL pointer with 0 for end of string

0224 7E        put_str_lcd:  mov a,m    ; get A from [HL]
0225 FE00                  cpi 0
0227 C22B02              jnz put_str_lcd1
022A C9                  ret

022B           put_str_lcd1:

022B CD0002              call lcd_ready
022E D351                out data_write
0230 23                  inx h
0231 F22402              jp put_str_lcd

                 ; goto_xy set cursor location on lcd
                 ; entry: HL: H = x, L = y

0234 CD0002    goto_xy:  call lcd_ready
0237 7D                  mov a,l
0238 FE00                cpi 0
023A C24302              jnz goto_xy1
023D 7C                  mov a,h
023E C680                adi 80h
0240 D350                out command_write
0242 C9                  ret

0243 FE01      goto_xy1: cpi 1
0245 C24E02              jnz goto_xy2
0248 7C                  mov a,h
0249 C6C0                adi 0c0h
024B D350                out command_write
024D C9                  ret

024E FE02      goto_xy2: cpi 2
0250 C25902              jnz goto_xy3
0253 7C                  mov a,h
0254 C694                adi 094h
0256 D350                out command_write
0258 C9                  ret

0259 FE03      goto_xy3: cpi 3
025B C26402              jnz goto_xy4
```

```
391     025E 7C                 mov a,h
392     025F C6D4               adi 0d4h
393     0261 D350               out command_write
394     0263 C9                 ret
395
396     0264 C9         goto_xy4: ret
397
398                     ; put_ch_lcd put character to lcd
399                     ; entry: A
400
401     0265 CD0002     put_ch_lcd: call lcd_ready
402     0268 D351                   out data_write
403     026A C9                     ret
404
405                     ;-------------------------------------------------------------------
406
407                     ; monitor call entry
408                     ; entry: E = monitor call number 0-255
409                     ; calling monitor function is made with RST 1 command after loading t
410                     ; regsiter E with call number
411                     ; destroy: BC user must save it in stack memory
412
413     026B E5         monitor_call:  push h
414     026C F5                        push psw
415     026D D5                        push d
416
417     026E 7B                        mov a,e    ; get call number
418     026F 07                        rlc        ; x2
419     0270 5F                        mov e,a    ; put it back
420
421     0271 217F02                    lxi h,vector_table
422     0274 1600                      mvi d,0
423     0276 19                        dad d      ; get location in jump table
424     0277 4E                        mov c,m
425     0278 23                        inx h
426     0279 46                        mov b,m
427
428     027A D1                        pop d
429     027B F1                        pop psw
430     027C E1                        pop h
431
432     027D C5                        push b     ; push address into top of stack
433
434     027E C9                        ret      ; jump to monitor call function
435
436     027F            vector_table:
437     027F B00B                      dwl demo  ; #0 running LED with HL pointer
438     0281 1006                      dwl delay ; #1 simple delay routine
439     0283 860B                      dwl cold_boot ; #2 show 8085 running
440     0285 BE0C                      dwl scan  ; #3 scan display one cycle
441     0287 4B12                      dwl cin   ; #4 get byte from console
442     0289 3F12                      dwl cout  ; #5 print byte to console
443     028B 6812                      dwl put_str ; #6 print string with 0 terminator to co
444     028D 1202                      dwl init_lcd ; #7 initialize lcd
445     028F 0002                      dwl lcd_ready ; #8 wait until lcd is ready
446     0291 0A02                      dwl clear_lcd ; #9 clear lcd display
447     0293 3402                      dwl goto_xy   ; #10 set lcd cursor position
448     0295 2402                      dwl put_str_lcd ; #11 print ASCII string on lcd
449     0297 6502                      dwl put_ch_lcd  ; #12 print ASCII letter on lcd
450     0299 0206                      dwl test_led    ; #13 run LED onboard
451     029B D301                      dwl pint8u     ; #14 print 8-bit unsigned to terminal
452
453
454
455                     ; save CPU registers to stack and write them to user registers
```

```
                    ;
                    
    029D            service_rst7:
    029D F5                     push psw
    029E C5                     push b
    029F D5                     push d
                    
                    ; save HL beforehand
                    
    02A0 2232F0                 shld user_HL
                    
    02A3 E1                     pop h
    02A4 2230F0                 shld user_DE
    02A7 E1                     pop h
    02A8 222EF0                 shld user_BC
    02AB E1                     pop h
    02AC 222CF0                 shld user_AF
    02AF E1                     pop h
                    
    02B0 222AF0                 shld user_PC   ; store next PC
                    
    02B3 210000                 lxi h,0000h
    02B6 39                     dad sp         ; get SP
    02B7 2234F0                 shld user_SP  ; save user SP
                    
    02BA CDCE0A                 call read_memory
    02BD CD640F                 call register_display1
                    
    02C0 2A57F0                 lhld save_stack
                    
    02C3 F9                     sphl            ; restore system stack
                    
                    
    02C4 C9                     ret
                    
                    
                    ; test diplay register after break
                    ; RST 2 opcode is D7
                    ; later will be changed to RST 7
                    
    02C5            service_rst2:
    02C5 F5                     push psw
    02C6 C5                     push b
    02C7 D5                     push d
                    
    02C8 2232F0           shld user_HL
    02CB E1                     pop h
    02CC 2230F0                 shld user_DE
    02CF E1                     pop h
    02D0 222EF0                 shld user_BC
    02D3 E1                     pop h
    02D4 222CF0                 shld user_AF
    02D7 E1                     pop h
                    
    02D8 222AF0                 shld user_PC   ; store next PC
                    
    02DB 210000                 lxi h,0000h
    02DE 39                     dad sp         ; get content of SP
                    
    02DF 2234F0                 shld user_SP   ; save user SP
                    
    02E2 CDCE0A                 call read_memory
    02E5 CD640F                 call register_display1
                    
    02E8 2A57F0                 lhld save_stack
```

```
521
522     02EB F9                     sphl            ; restore system stack
523
524
525     02EC C9                     ret
526
527
528                     ; service trap for single step running
529                     ; disable trap input by setting system port c.6
530                     ; save CPU registers to user registers
531
532     02ED            service_trap:
533     02ED F5                     push psw        ; save A and Flag
534
535     02EE 3EFF                   mvi a,0ffh
536     02F0 D312                   out system_port_c ; turn trap off by clearing shift reg
537
538     02F2 C5                     push b
539     02F3 D5                     push d
540     02F4 E5                     push h
541
542     02F5 E1                     pop h
543     02F6 2232F0                 shld user_HL   ; save HL
544     02F9 E1                     pop h
545     02FA 2230F0                 shld user_DE
546     02FD E1                     pop h
547     02FE 222EF0                 shld user_BC
548     0301 E1                     pop h
549     0302 222CF0                 shld user_AF
550
551     0305 E1                     pop h           ; store next PC
552     0306 222AF0                 shld user_PC
553
554     0309 210000                 lxi h,0
555     030C 39                     dad sp
556     030D 2234F0                 shld user_SP   ; save user SP
557
558     0310 CDCE0A                 call read_memory
559
560     0313 3A25F0                 lda uart_found
561     0316 FE00                   cpi 0
562     0318 CA1E03                 jz skip1
563     031B CD640F                 call register_display1
564
565     031E            skip1:
566     031E 2A57F0                 lhld save_stack
567
568     0321 F9                     sphl            ; restore system stack
569
570     0322 C9                     ret             ; jump back to main body
571
572
573                     ; disassemble machine code into mnemonic
574
575     0323            disassemble1:
576     0323 3A22F0                     lda command
577     0326 FE64                       cpi "d"
578     0328 C24903                     jnz exit_disassemble
579
580     032B 219A1E                     lxi h,disassemble_text
581     032E CD6812                     call put_str
582
583     0331 CDEA12                     call new_line
584     0334 0E10                       mvi c,16       ; 16 lines
585
```

```
586
587     0336 C5        dis2:            push b
588
589     0337 CDEA12                     call new_line
590
591     033A CDC003                     call d_disassemble
592
593     033D C1                         pop b
594     033E 0D                         dcr c
595     033F C23603                     jnz dis2
596
597     0342 CDEA12                     call new_Line
598     0345 CD4A0D                     call send_prompt
599     0348 C9                         ret
600
601
602     0349 C9        exit_disassemble: ret
603
604
605                    ; disassemble opcode to mnemonic
606                    ; entry: user_PC
607                    ; exit: user_PC = next address
608
609     034A 2A2AF0    disassemble:  lhld user_PC
610
611     034D E5                      push h
612
613     034E 7C                      mov a,h
614     034F CDDB12                  call out2x
615     0352 7D                      mov a,l
616     0353 CDDB12                  call out2x
617     0356 CDF512                  call space
618
619     0359 7E                      mov a,m     ; get opcode
620     035A CD3604                  call get_number_of_byte
621     035D 4F                      mov c,a
622
623     035E 7E        disassem3:    mov a,m
624     035F CDDB12                  call out2x
625     0362 23                      inx h
626     0363 0D                      dcr c
627     0364 C25E03                  jnz disassem3
628
629     0367 E1                      pop h
630
631     0368 7E                      mov a,m
632     0369 CD3604                  call get_number_of_byte
633     036C FE01                    cpi 1
634     036E C27603                  jnz one_tab
635
636     0371 3E09                    mvi a,tab         ; print two tabs for one byte opcode
637     0373 CD3F12                  call cout
638
639     0376 3E09      one_tab:      mvi a,tab         ; else only one tab
640     0378 CD3F12                  call cout
641
642     037B E5                      push h
643
644     037C 7E                      mov a,m           ; get opcode
645
646     037D F5                      push psw
647
648     037E 210000                  lxi h,0000h       ; clear HL
649     0381 6F                      mov l,a
650
```

```
0382 29                     dad h           ; HL = HLx2

0383 5D                     mov e,l
0384 54                     mov d,h

0385 211B15                 lxi h,ins_table
0388 19                     dad d           ; ADD HL,DE
0389 5E                     mov e,m
038A 23                     inx h
038B 56                     mov d,m

038C 6B                     mov l,e
038D 62                     mov h,d

038E CD6812                 call put_str

0391 F1                     pop psw
0392 E1                     pop h

0393 CD3604                 call get_number_of_byte
0396 FE01                   cpi 1
0398 C2A003                 jnz disassem1
039B 23                     inx h
039C 222AF0                 shld user_PC
039F C9                     ret

03A0 FE02       disassem1:  cpi 2
03A2 C2AF03                 jnz disassem2
03A5 23                     inx h
03A6 7E                     mov a,m
03A7 CDDB12                 call out2x
03AA 23                     inx h
03AB 222AF0                 shld user_PC
03AE C9                     ret

03AF 23         disassem2:  inx h
03B0 23                     inx h
03B1 7E                     mov a,m
03B2 CDDB12                 call out2x
03B5 2B                     dcx h
03B6 7E                     mov a,m
03B7 CDDB12                 call out2x
03BA 23                     inx h
03BB 23                     inx h
03BC 222AF0                 shld user_PC
03BF C9                     ret

                    ; disassemble opcode to mnemonic with command 'd'
                    ; entry: pointer
                    ; exit: pointer = next address

03C0 2A3CF0     d_disassemble:  lhld pointer

03C3 E5                     push h

03C4 7C                     mov a,h
03C5 CDDB12                 call out2x
03C8 7D                     mov a,l
03C9 CDDB12                 call out2x
03CC CDF512                 call space

03CF 7E                     mov a,m     ; get opcode
03D0 CD3604                 call get_number_of_byte
03D3 4F                     mov c,a

```

```
716     03D4 7E        d_disassem3:    mov a,m
717     03D5 CDDB12                   call out2x
718     03D8 23                       inx h
719     03D9 0D                       dcr c
720     03DA C2D403                   jnz d_disassem3
721
722     03DD E1                       pop h
723
724     03DE 7E                       mov a,m
725     03DF CD3604                   call get_number_of_byte
726     03E2 FE01                     cpi 1
727     03E4 C2EC03                   jnz d_one_tab
728
729     03E7 3E09                     mvi a,tab         ; print two tabs for one byte opcode
730     03E9 CD3F12                   call cout
731
732     03EC 3E09      d_one_tab:      mvi a,tab         ; else only one tab
733     03EE CD3F12                   call cout
734
735     03F1 E5                       push h
736
737     03F2 7E                       mov a,m           ; get opcode
738
739     03F3 F5                       push psw
740
741     03F4 210000                   lxi h,0000h       ; clear HL
742     03F7 6F                       mov l,a
743
744     03F8 29                       dad h            ; HL = HLx2
745
746     03F9 5D                       mov e,l
747     03FA 54                       mov d,h
748
749     03FB 211B15                   lxi h,ins_table
750     03FE 19                       dad d            ; ADD HL,DE
751     03FF 5E                       mov e,m
752     0400 23                       inx h
753     0401 56                       mov d,m
754
755     0402 6B                       mov l,e
756     0403 62                       mov h,d
757
758     0404 CD6812                   call put_str
759
760     0407 F1                       pop psw
761     0408 E1                       pop h
762
763     0409 CD3604                   call get_number_of_byte
764     040C FE01                     cpi 1
765     040E C21604                   jnz d_disassem1
766     0411 23                       inx h
767     0412 223CF0                   shld pointer
768     0415 C9                       ret
769
770     0416 FE02      d_disassem1:    cpi 2
771     0418 C22504                   jnz d_disassem2
772     041B 23                       inx h
773     041C 7E                       mov a,m
774     041D CDDB12                   call out2x
775     0420 23                       inx h
776     0421 223CF0                   shld pointer
777     0424 C9                       ret
778
779     0425 23        d_disassem2:  inx h
780     0426 23                       inx h
```

```
781     0427 7E                     mov a,m
782     0428 CDDB12                 call out2x
783     042B 2B                     dcx h
784     042C 7E                     mov a,m
785     042D CDDB12                 call out2x
786     0430 23                     inx h
787     0431 23                     inx h
788     0432 223CF0                 shld pointer
789     0435 C9                     ret
790
791
792                     ; get number of byte
793                     ; entry: A = OPCODE
794                     ; exit: A = number of byte, 1,2,3
795                     ;           0 = undefined opcode
796
797     0436            get_number_of_byte:
798
799     0436 FE01                   cpi 01
800     0438 C23E04                 jnz number1
801     043B 3E03                   mvi a,3
802     043D C9                     ret
803
804     043E FE06       number1:    cpi 6
805     0440 C24604                 jnz number2
806     0443 3E02                   mvi a,2
807     0445 C9                     ret
808
809     0446 FE0E       number2:    cpi 0eh
810     0448 C24E04                 jnz number3
811     044B 3E02                   mvi a,2
812     044D C9                     ret
813
814     044E FE11       number3:    cpi 11h
815     0450 C25604                 jnz number4
816     0453 3E03                   mvi a,3
817     0455 C9                     ret
818
819     0456 FE16       number4:    cpi 16h
820     0458 C25E04                 jnz number5
821     045B 3E02                   mvi a,2
822     045D C9                     ret
823
824     045E FE1E       number5:    cpi 1eh
825     0460 C26604                 jnz number6
826     0463 3E02                   mvi a,2
827     0465 C9                     ret
828
829     0466 FE21       number6:    cpi 21h
830     0468 C26E04                 jnz number7
831     046B 3E03                   mvi a,3
832     046D C9                     ret
833
834     046E FE22       number7:    cpi 22h
835     0470 C27604                 jnz number8
836     0473 3E03                   mvi a,3
837     0475 C9                     ret
838
839     0476 FE26       number8:    cpi 26h
840     0478 C27E04                 jnz number9
841     047B 3E02                   mvi a,2
842     047D C9                     ret
843
844     047E FE2A       number9:    cpi 2ah
845     0480 C28604                 jnz number10
```

```
0483 3E03                   mvi a,3
0485 C9                     ret

0486 FE2E      number10:    cpi 2eh
0488 C28E04                 jnz number11
048B 3E02                   mvi a,2
048D C9                     ret

048E FE31      number11:    cpi 31h
0490 C29604                 jnz number12
0493 3E03                   mvi a,3
0495 C9                     ret

0496 FE32      number12:    cpi 32h
0498 C29E04                 jnz number13
049B 3E03                   mvi a,3
049D C9                     ret

049E FE36      number13:    cpi 36h
04A0 C2A604                 jnz number14
04A3 3E03                   mvi a,3
04A5 C9                     ret

04A6 FE3A      number14:    cpi 3ah
04A8 C2AE04                 jnz number15
04AB 3E03                   mvi a,3
04AD C9                     ret

04AE FE3E      number15:    cpi 3eh
04B0 C2B604                 jnz number16
04B3 3E02                   mvi a,2
04B5 C9                     ret

04B6 FEC2      number16:    cpi 0c2h
04B8 C2BE04                 jnz number17
04BB 3E03                   mvi a,3
04BD C9                     ret

04BE FEC3      number17:    cpi 0c3h
04C0 C2C604                 jnz number18
04C3 3E03                   mvi a,3
04C5 C9                     ret

04C6 FEC4      number18:    cpi 0c4h
04C8 C2CE04                 jnz number19
04CB 3E03                   mvi a,3
04CD C9                     ret

04CE FEC6      number19:    cpi 0c6h
04D0 C2D604                 jnz number20
04D3 3E02                   mvi a,2
04D5 C9                     ret

04D6 FECA      number20:    cpi 0cah
04D8 C2DE04                 jnz number21
04DB 3E03                   mvi a,3
04DD C9                     ret

04DE FECC      number21:    cpi 0cch
04E0 C2E604                 jnz number22
04E3 3E03                   mvi a,3
04E5 C9                     ret

04E6 FECD      number22:    cpi 0cdh
04E8 C2EE04                 jnz number23
```

```
911     04EB 3E03                   mvi a,3
912     04ED C9                     ret
913
914     04EE FECE       number23:   cpi 0ceh
915     04F0 C2F604                 jnz number24
916     04F3 3E02                   mvi a,2
917     04F5 C9                     ret
918
919     04F6 FED2       number24:   cpi 0d2h
920     04F8 C2FE04                 jnz number25
921     04FB 3E03                   mvi a,3
922     04FD C9                     ret
923
924     04FE FED3       number25:   cpi 0d3h
925     0500 C20605                 jnz number26
926     0503 3E02                   mvi a,2
927     0505 C9                     ret
928
929     0506 FED4       number26:   cpi 0d4h
930     0508 C20E05                 jnz number27
931     050B 3E03                   mvi a,3
932     050D C9                     ret
933
934     050E FED6       number27:   cpi 0d6h
935     0510 C21605                 jnz number28
936     0513 3E02                   mvi a,2
937     0515 C9                     ret
938
939     0516 FEDA       number28:   cpi 0dah
940     0518 C21E05                 jnz number29
941     051B 3E03                   mvi a,3
942     051D C9                     ret
943
944     051E FEDB       number29:   cpi 0dbh
945     0520 C22605                 jnz number30
946     0523 3E02                   mvi a,2
947     0525 C9                     ret
948
949     0526 FEDC       number30:   cpi 0dch
950     0528 C22E05                 jnz number31
951     052B 3E03                   mvi a,3
952     052D C9                     ret
953
954     052E FEE2       number31:   cpi 0e2h
955     0530 C23605                 jnz number32
956     0533 3E03                   mvi a,3
957     0535 C9                     ret
958
959     0536 FEE4       number32:   cpi 0e4h
960     0538 C23E05                 jnz number33
961     053B 3E03                   mvi a,3
962     053D C9                     ret
963
964     053E FEE6       number33:   cpi 0e6h
965     0540 C24605                 jnz number34
966     0543 3E02                   mvi a,2
967     0545 C9                     ret
968
969     0546 FEEA       number34:   cpi 0eah
970     0548 C24E05                 jnz number35
971     054B 3E03                   mvi a,3
972     054D C9                     ret
973
974     054E FEEC       number35:   cpi 0ech
975     0550 C25605                 jnz number36
```

```
 976     0553 3E03                  mvi a,3
 977     0555 C9                    ret
 978
 979     0556 FEEE      number36:   cpi 0eeh
 980     0558 C25E05                jnz number37
 981     055B 3E02                  mvi a,2
 982     055D C9                    ret
 983
 984     055E FEF2      number37:   cpi 0f2h
 985     0560 C26605                jnz number38
 986     0563 3E03                  mvi a,3
 987     0565 C9                    ret
 988
 989     0566 FEF4      number38:   cpi 0f4h
 990     0568 C26E05                jnz number39
 991     056B 3E03                  mvi a,3
 992     056D C9                    ret
 993
 994     056E FEF6      number39:   cpi 0f6h
 995     0570 C27605                jnz number40
 996     0573 3E02                  mvi a,2
 997     0575 C9                    ret
 998
 999     0576 FEFA      number40:   cpi 0fah
1000     0578 C27E05                jnz number41
1001     057B 3E03                  mvi a,3
1002     057D C9                    ret
1003
1004     057E FEFC      number41:   cpi 0fch
1005     0580 C28605                jnz number42
1006     0583 3E03                  mvi a,3
1007     0585 C9                    ret
1008
1009     0586 FEFE      number42:   cpi 0feh
1010     0588 C28E05                jnz number43
1011     058B 3E02                  mvi a,2
1012     058D C9                    ret
1013
1014     058E 3E01      number43:   mvi a,1
1015     0590 C9                    ret
1016
1017
1018
1019
1020
1021
1022
1023
1024                    ; execute key 0-F or 10H-19H
1025
1026     0591 FE10      key_execute: cpi 10h
1027     0593 D29D05                 jnc  function_key   ; 0-9 jump to data key
1028     0596 57                     mov d,a
1029     0597 CD4106                 call buzzer
1030     059A F21906                 jp   data_key
1031
1032     059D FE12      function_key: cpi 12h
1033     059F C2A805                  jnz function1
1034     05A2 CD4106                  call buzzer
1035     05A5 F2750A                  jp  increment
1036
1037     05A8 FE15      function1:    cpi 15h
1038     05AA C2B305                  jnz function2
1039     05AD CD4106                  call buzzer
1040     05B0 F2880A                  jp  decrement
```

```
1041
1042    05B3 FE10      function2:    cpi 10h
1043    05B5 C2BE05                  jnz function3
1044    05B8 CD4106                  call buzzer
1045    05BB F2110A                  jp  address_mode
1046
1047    05BE FE11      function3:    cpi 11h
1048    05C0 C2C905                  jnz function4
1049    05C3 CD4106                  call buzzer
1050    05C6 F21E0A                  jp  data_mode
1051
1052    05C9 FE13      function4:    cpi 13h
1053    05CB C2D505                  jnz function5
1054    05CE CD4106                  call buzzer
1055    05D1 F2AB09                  jp  go
1056    05D4 C9                      ret
1057
1058    05D5 FE14      function5:   cpi 14h
1059    05D7 C2E005                  jnz function6
1060    05DA CD4106                  call buzzer
1061    05DD F2290A                  jp function_2nd
1062
1063
1064    05E0 FE16      function6:   cpi 16h
1065    05E2 C2EB05                  jnz function7
1066    05E5 CD4106                  call buzzer
1067    05E8 F2CF09                  jp single_step
1068
1069    05EB FE17      function7:    cpi 17h
1070    05ED C2F605                  jnz function8
1071    05F0 CD4106                  call buzzer
1072    05F3 F29109                  jp home
1073
1074    05F6 FE18      function8:    cpi 18h
1075    05F8 C20106                  jnz function9
1076    05FB CD4106                  call buzzer
1077    05FE F24D0A                  jp modify_register
1078
1079    0601 C9        function9:    ret
1080
1081                   ; test running onboard led
1082
1083    0602 3E01      test_led:  mvi a,1
1084
1085    0604 D300      test_led1: out gpio
1086    0606 115010               lxi d,1050h
1087    0609 CD1006               call delay
1088    060C 07                   rlc
1089    060D C30406               jmp  test_led1
1090
1091
1092                   ; delay subroutine
1093                   ; entry: D= outer loop E=inner loop (should be 0 for long delay)
1094                   ; exit: none
1095
1096    0610 1D        delay:     dcr e
1097    0611 C21006               jnz delay
1098    0614 15                   dcr d
1099    0615 C21006               jnz delay
1100    0618 C9                   ret
1101
1102
1103    0619 3A26F0    data_key:     lda entry_mode
1104    061C FE00                    cpi 0
1105    061E C22406                  jnz data_key1
```

```
0621 F2DF06                 jp  enter_data

0624 FE01      data_key1:   cpi 1
0626 C22C06                 jnz data_key2
0629 F21907                 jp  enter_address

062C FE02      data_key2:   cpi 2
062E C23406                 jnz data_key3
0631 F25307                 jp select_register

0634 FE03      data_key3:   cpi 3
0636 C23C06                 jnz data_key4
0639 F27F06                 jp enter_register

063C           data_key4:

063C C9                     ret


063D CD5006    test_buzzer:  call beep_on
0640 C9              ret

                     ;mvi a,7fh
                           ;out system_port_c
                           ;lxi d,1000h
                           ;call delay
                           ;mvi a,0ffh
                           ;out system_port_c
                           ;ret


                   ; produce beep output at system port c.7
                   ; click when key pressed

0641 3A24F0    buzzer:      lda beep_flag
0644 E601                   ani 1
0646 CA5006          jz  beep_on

0649 0600            mvi b,0
064B 05        delay_nobeep: dcr b
064C C24B06                 jnz delay_nobeep
064F C9                     ret
0650           beep_on:
0650 0E20            mvi c,20h

0652 3E7F      buzzer1:     mvi a,7fh
0654 D312                   out system_port_c  ;nop             ;out system_port_c
0656 CD6906                 call delay_us
0659 3EFF                   mvi a,0ffh
065B D312                   out system_port_c  ;nop             ;out system_port_c
065D CD6906          call delay_us

0660 0D                     dcr c
0661 C25206                 jnz buzzer1

0664 3EFF                   mvi a,0ffh
0666 D312                   out system_port_c

0668 C9                     ret

0669 0660      delay_us:    mvi b,60h
066B 05        delay_us1:   dcr b
066C C26B06                 jnz delay_us1
066F C9                     ret
```

```
1171
1172                    ;-----  turn display off while key has been pressing ----------
1173                    ;       useful for no function accepted
1174
1175    0670 219E1D     display_off:  lxi h,off_display
1176    0673 CDBE0C     off_display1:  call scan
1177    0676 3A21F0                    lda key
1178    0679 FEFF                      cpi 0ffh
1179    067B C27306                    jnz  off_display1  ; loop if key still pressed
1180    067E C9                        ret
1181
1182                    ;************* modify current displayed register ******************
1183                    ; entry: current user register displayed
1184                    ;
1185
1186
1187    067F            enter_register:
1188
1189    067F 2A38F0                    lhld current_register
1190
1191    0682 4E                        mov c,m
1192    0683 23                        inx h
1193    0684 46                        mov b,m
1194    0685 210000                    lxi h,0
1195
1196    0688 09                        dad b           ; MOV HL,BC
1197
1198    0689 5A                        mov e,d         ; save key code to E
1199
1200    068A            shift_register:
1201    068A 7D                         mov a,l
1202    068B 07                         rlc
1203    068C 6F                         mov l,a
1204    068D 7C                         mov a,h
1205    068E 17                         ral
1206    068F 67                         mov h,a
1207
1208    0690 7D                         mov a,l
1209    0691 07                         rlc
1210    0692 6F                         mov l,a
1211    0693 7C                         mov a,h
1212    0694 17                         ral
1213    0695 67                         mov h,a
1214
1215    0696 7D                         mov a,l
1216    0697 07                         rlc
1217    0698 6F                         mov l,a
1218    0699 7C                         mov a,h
1219    069A 17                         ral
1220    069B 67                         mov h,a
1221
1222    069C 7D                         mov a,l
1223    069D 07                         rlc
1224    069E 6F                         mov l,a
1225    069F 7C                         mov a,h
1226    06A0 17                         ral
1227    06A1 67                         mov h,a
1228
1229    06A2 7D                         mov a,l
1230
1231    06A3 E6F0                       ani 0f0h
1232    06A5 82                         add d
1233    06A6 6F                         mov l,a
1234
1235    06A7 44                         mov b,h
```

```
06A8 4D                      mov c,l

06A9 2A38F0                  lhld current_register
06AC 71                      mov m,c
06AD 23                      inx h
06AE 70                      mov m,b

06AF 2A38F0                 lhld current_register

06B2 4E                     mov c,m
06B3 23                     inx h
06B4 46                     mov b,m
06B5 210000                 lxi h,0

06B8 09                     dad b           ; MOV HL,BC

06B9 CDA30A                 call read_register

06BC F5                    push psw

06BD 3A13F0          lda buffer+3
06C0 F680                   ori 80h
06C2 3213F0                 sta buffer+3

06C5 3A12F0          lda buffer+2
06C8 F680                   ori 80h
06CA 3212F0                 sta buffer+2

06CD 3A11F0          lda buffer+1
06D0 F680                   ori 80h
06D2 3211F0                 sta buffer+1

06D5 3A10F0          lda buffer
06D8 F680                   ori 80h
06DA 3210F0                 sta buffer




06DD F1              pop psw

06DE C9                      ret

                ; enter nibble into current location

06DF 2A2AF0    enter_data:   lhld user_PC

06E2 5A                      mov e,d        ; save key code to E

06E3 3A27F0                  lda counter1
06E6 FE00                    cpi 0
06E8 C2F206                  jnz shift_data
06EB 3C                      inr a
06EC 3227F0                  sta counter1
06EF 3E00                    mvi a,0
06F1 77                      mov m,a

06F2 7E        shift_data:   mov a,m

06F3 07                      rlc
06F4 07                      rlc
06F5 07                      rlc
06F6 07                      rlc
06F7 E6F0                    ani 0f0h       ; make low nibble to 0 before insert
```

```
1301    06F9 83                     add e          ; insert low nibble to A
1302    06FA 77                     mov m,a
1303    06FB 57                     mov d,a
1304    06FC 7E                     mov a,m        ; check if the space is ram or rom
1305    06FD BA                     cmp d
1306    06FE CA1507                 jz  it_is_ram
1307
1308                    ; if it was rom them turn of led while key has been pressed
1309
1310    0701 219E1D                 lxi h,off_display
1311    0704 CDBE0C     enter_data1:  call scan
1312    0707 3A21F0                 lda key
1313    070A FEFF                   cpi 0ffh
1314    070C C20407                 jnz  enter_data1  ; loop if key still pressed
1315
1316    070F CDB70C                 call debounce     ; debounce after key was released
1317
1318    0712 2110F0                 lxi h,buffer      ; back to show display again
1319
1320    0715            it_is_ram:
1321    0715 CDCE0A                 call read_memory
1322    0718 C9                     ret
1323
1324                    ; enter nibble into current pointer
1325
1326    0719 2A2AF0     enter_address: lhld user_PC
1327    071C 5A                         mov e,d        ; save key code to E
1328
1329    071D 3A28F0                 lda counter2
1330    0720 FE00                   cpi 0
1331    0722 C22F07                 jnz shift_address
1332    0725 3C                     inr a
1333    0726 3228F0                 sta counter2
1334    0729 210000                 lxi h,0
1335    072C 222AF0                 shld user_PC
1336
1337    072F            shift_address:
1338    072F 7D                      mov a,l
1339    0730 07                      rlc
1340    0731 6F                      mov l,a
1341    0732 7C                      mov a,h
1342    0733 17                      ral
1343    0734 67                      mov h,a
1344
1345    0735 7D                      mov a,l
1346    0736 07                      rlc
1347    0737 6F                      mov l,a
1348    0738 7C                      mov a,h
1349    0739 17                      ral
1350    073A 67                      mov h,a
1351
1352    073B 7D                      mov a,l
1353    073C 07                      rlc
1354    073D 6F                      mov l,a
1355    073E 7C                      mov a,h
1356    073F 17                      ral
1357    0740 67                      mov h,a
1358
1359    0741 7D                      mov a,l
1360    0742 07                      rlc
1361    0743 6F                      mov l,a
1362    0744 7C                      mov a,h
1363    0745 17                      ral
1364    0746 67                      mov h,a
1365
```

```
0747 7D                      mov a,l

0748 E6F0                    ani 0f0h
074A 82                      add d
074B 6F                      mov l,a

074C 222AF0                  shld user_PC   ; store new pointer

074F CDCE0A                  call read_memory

0752 C9                      ret


                    ;****************** ALT register display ***************************

0753 5A             select_register: mov e,d  ; save key for selecting user register

0754 7A                      mov a,d
0755 FE00                    cpi 0
0757 C27107                  jnz register1

075A 3E77                    mvi a,77h       ; AF register pair
075C 3214F0                  sta buffer+4
075F 3E71                    mvi a,71h
0761 3215F0                  sta buffer+5

0764 212CF0                  lxi h,user_AF
0767 2238F0                  shld current_register


076A 2A2CF0                  lhld user_AF
076D CDA30A                  call read_register
0770 C9                      ret

0771                register1:
0771 FE01                    cpi 1
0773 C28D07                  jnz register2

0776 3E7C                    mvi a,7ch       ; BC register pair
0778 3214F0                  sta buffer+4
077B 3E39                    mvi a,39h
077D 3215F0                  sta buffer+5

0780 212EF0                  lxi h,user_BC
0783 2238F0                  shld current_register

0786 2A2EF0                  lhld user_BC
0789 CDA30A                  call read_register

078C C9                      ret

078D                register2:
078D FE02                    cpi 2
078F C2A907                  jnz register3

0792 3E5E                    mvi a,5eh       ; DE register pair
0794 3214F0                  sta buffer+4
0797 3E79                    mvi a,79h
0799 3215F0                  sta buffer+5

079C 2130F0                  lxi h,user_DE
079F 2238F0                  shld current_register

07A2 2A30F0                  lhld user_DE
07A5 CDA30A                  call read_register
```

```
1431
1432    07A8 C9                     ret
1433
1434    07A9            register3:
1435    07A9 FE03                   cpi 3
1436    07AB C2C507                 jnz register4
1437
1438    07AE 3E76                   mvi a,76h       ; HL register pair
1439    07B0 3214F0                 sta buffer+4
1440    07B3 3E38                   mvi a,38h
1441    07B5 3215F0                 sta buffer+5
1442
1443    07B8 2132F0                 lxi h,user_HL
1444    07BB 2238F0                 shld current_register
1445
1446    07BE 2A32F0                 lhld user_HL
1447    07C1 CDA30A                 call read_register
1448
1449    07C4 C9                     ret
1450    07C5            register4:
1451    07C5 FE04                   cpi 4
1452    07C7 C2E107                 jnz register5
1453
1454    07CA 3E6D                   mvi a,6dh        ; user SP
1455    07CC 3214F0                 sta buffer+4
1456    07CF 3E73                   mvi a,73h
1457    07D1 3215F0                 sta buffer+5
1458
1459    07D4 2134F0                 lxi h,user_SP
1460    07D7 2238F0                 shld current_register
1461
1462    07DA 2A34F0                 lhld user_SP
1463    07DD CDA30A                 call read_register
1464
1465    07E0 C9                     ret
1466
1467    07E1            register5:
1468    07E1 FE05                   cpi 5
1469    07E3 C20808                 jnz register6
1470
1471    07E6 3E73                   mvi a,73h       ; user PC
1472    07E8 3214F0                 sta buffer+4
1473    07EB 3E39                   mvi a,39h
1474    07ED 3215F0                 sta buffer+5
1475
1476    07F0 212AF0                 lxi h,user_PC
1477    07F3 2238F0                 shld current_register
1478
1479    07F6 2A2AF0                 lhld user_PC
1480    07F9 CDA30A                 call read_register
1481
1482    07FC C9                     ret
1483
1484                    ;---- display carry flag ------------------------------------------
1485
1486    07FD C20508     put_flag:   jnz put_high1
1487    0800 3E3F                   mvi a,3fh
1488    0802 C30708                 jmp skip_put_high1
1489
1490    0805 3E06       put_high1:  mvi a,06h
1491
1492    0807            skip_put_high1:
1493    0807 C9                     ret
1494
1495    0808            register6:
```

```
0808 FE06                   cpi 6
080A C23008                 jnz register7

080D 3E39                   mvi a,39h       ; carry flag
080F 3210F0                 sta buffer
0812 3E6E                   mvi a,6eh
0814 3211F0                 sta buffer+1
0817 3E48                   mvi a,48h
0819 3212F0                 sta buffer+2

081C 2A2CF0                 lhld user_AF
081F 7D                     mov a,l

0820 E601                   ani 1
0822 CDFD07                 call put_flag
0825 3213F0                 sta buffer+3

0828 AF                     xra a
0829 3214F0                 sta buffer+4
082C 3215F0                 sta buffer+5

082F C9                     ret

0830            register7:
0830 FE07                   cpi 7
0832 C25B08                 jnz register8

0835 3E49                   mvi a,49h       ; zero flag
0837 3210F0                 sta buffer
083A 3E79                   mvi a,79h
083C 3211F0                 sta buffer+1
083F 3E50                   mvi a,50h
0841 3212F0                 sta buffer+2
0844 3E5C                   mvi a,5ch
0846 3213F0                 sta buffer+3
0849 3E48                   mvi a,48h
084B 3214F0                 sta buffer+4


084E 2A2CF0                 lhld user_AF
0851 7D                     mov a,l

0852 E640                   ani 40h
0854 CDFD07                 call put_flag
0857 3215F0                 sta  buffer+5

085A C9                     ret


085B            register8:
085B FE08                   cpi 8
085D C28C08                 jnz register9

0860 3E6D                   mvi a,6dh       ; sign flag
0862 3210F0                 sta buffer
0865 3E11                   mvi a,11h
0867 3211F0                 sta buffer+1
086A 3E6F                   mvi a,6fh
086C 3212F0                 sta buffer+2
086F 3E54                   mvi a,54h
0871 3213F0                 sta buffer+3
0874 3E48                   mvi a,48h
0876 3214F0                 sta buffer+4


```

```
0879 2A2CF0                 lhld user_AF
087C 7D                     mov a,l

087D 17                     ral
087E DA8608                 jc  put_high2
0881 3E3F                   mvi a,3fh
0883 C38808                 jmp skip_put_high2

0886 3E06       put_high2:  mvi a,06h
0888            skip_put_high2:
0888 3215F0                 sta  buffer+5

088B C9                     ret


088C            register9:
088C FE09                   cpi 9
088E C2B408                 jnz register10

0891 3E77                   mvi a,77h       ; AC flag
0893 3210F0                 sta buffer
0896 3E39                   mvi a,39h
0898 3211F0                 sta buffer+1
089B 3E48                   mvi a,48h
089D 3212F0                 sta buffer+2

08A0 2A2CF0                 lhld user_AF
08A3 7D                     mov a,l

08A4 E610                   ani 10h
08A6 CDFD07                 call put_flag
08A9 3213F0                 sta  buffer+3
08AC AF                     xra  a
08AD 3214F0                 sta buffer+4
08B0 3215F0                 sta buffer+5

08B3 C9                     ret

08B4            register10:
08B4 FE0A                   cpi 10
08B6 C2DA08                 jnz break

08B9 3E73                   mvi a,73h       ; Parity flag
08BB 3210F0                 sta buffer
08BE 3E48                   mvi a,48h
08C0 3211F0                 sta buffer+1

08C3 2A2CF0                 lhld user_AF
08C6 7D                     mov a,l

08C7 E604                   ani 4
08C9 CDFD07                 call put_flag
08CC 3212F0                 sta  buffer+2
08CF AF                     xra  a
08D0 3213F0                 sta buffer+3
08D3 3214F0                 sta buffer+4
08D6 3215F0                 sta buffer+5
08D9 C9                     ret


                ; ------------------ ALT B SET BREAK POINT -------------------------
08DA FE0B       break:      cpi 11
08DC C21409                 jnz clear_break

08DF 2A2AF0                 lhld user_PC  ; save user PC
```

```
08E2 223EF0                 shld break_address

08E5 7E                     mov a,m       ; get user code
08E6 3240F0                 sta break_opcode ; save it

08E9 E5                     push h
08EA CDCE0A                 call read_memory
08ED E1                     pop h
08EE 3EFF                   mvi a,0FFh    ; RST 7 opcode
08F0 77                     mov m,a       ; replace user code with RST 7

08F1 F5                     push psw

08F2 3A10F0                 lda buffer
08F5 F680                   ori 80h
08F7 3210F0                 sta buffer

08FA 3A11F0                 lda buffer+1
08FD F680                   ori 80h
08FF 3211F0                 sta buffer+1

0902 3A12F0                 lda buffer+2
0905 F680                   ori 80h
0907 3212F0                 sta buffer+2

090A 3A13F0                 lda buffer+3
090D F680                   ori 80h
090F 3213F0                 sta buffer+3
0912 F1                     pop psw

0913 C9                     ret

                ;----------------- ALT C CLEAR BREAK POINT -------------------------

0914 FE0C       clear_break:  cpi 12
0916 C22F09                 jnz insert_byte

0919 2140F0                 lxi h,break_opcode ; restore user code
091C 7E                     mov a,m

091D 2A3EF0                 lhld break_address
0920 77                     mov m,a
0921 222AF0                 shld user_PC
0924 CDCE0A                 call read_memory
0927 AF                     xra a
0928 3226F0                 sta entry_mode
092B CD120B                 call mode_indicator

092E C9                     ret

                ;--------------- ALT E insert byte --------------------------------
                ; insert byte within 512 bytes from current location

092F FE0E       insert_byte: cpi 14      ; test with key E
0931 C25A09                 jnz delete_byte

0934 2A2AF0                 lhld user_PC
0937 E5               push h         ; save PC to stack

0938 110002                 lxi d,512
093B 19               dad d
093C E5               push h
093D C1               pop b          ; copy HL to BC
093E 0B               dcx b

```

```
093F 110002          lxi d,512      ; load counter with 512 bytes

0942            insert_byte1:
0942 0A              ldax b
0943 77              mov m,a
0944 2B              dcx h
0945 0B              dcx b
0946 1B              dcx d
0947 7B              mov a,e
0948 B2              ora d          ; check DE ==0
0949 C24209          jnz insert_byte1

094C E1              pop h          ; restore user PC
094D AF              xra a
094E 77              mov m,a        ; store 00 at insert byte
094F CDCE0A                 call read_memory
0952 AF                     xra a
0953 3226F0                 sta entry_mode
0956 CD120B                 call mode_indicator

0959 C9                      ret


                ;--------------- ALT D delete byte ---------------------------------
                ; delete byte within 512 bytes

095A FE0D       delete_byte: cpi 13
095C C27F09                 jnz beep_chk

095F 2A2AF0                 lhld user_PC
0962 E5                     push h
0963 E5              push h
0964 C1              pop b

0965 03              inx b
0966 110002          lxi d,512

0969            delete_byte1:
0969 0A              ldax b
096A 77              mov m,a
096B 23              inx h
096C 03              inx b
096D 1B              dcx d
096E 7B              mov a,e
096F B2              ora d   ; check if DE ==0
0970 C26909          jnz delete_byte1

0973 E1                     pop h
0974 CDCE0A          call read_memory
0977 AF                     xra a
0978 3226F0                 sta entry_mode
097B CD120B                 call mode_indicator

097E C9                     ret

                ;--------------- ALT F BEEP/NO BEEP  ---------------------------------
097F FE0F       beep_chk:   cpi 15
0981 C28D09                 jnz option1

0984 3A24F0          lda beep_flag
0987 EE01            xri 1
0989 3224F0          sta beep_flag
098C C9              ret


```

```
1756    098D            option1:
1757    098D CD7006                     call display_off ; no service key
1758    0990 C9                         ret
1759
1760
1761
1762    0991 210081     home:           lxi h,home_address
1763    0994 222AF0                     shld user_PC
1764    0997 2110F0                     lxi h,buffer
1765    099A CDCE0A                     call read_memory
1766    099D AF                         xra a
1767    099E 3226F0                     sta entry_mode
1768    09A1 CD120B                     call mode_indicator
1769    09A4 C9                         ret
1770
1771
1772    09A5 3E2A       debug:          mvi a,"*"
1773    09A7 CD3F12                     call cout
1774    09AA C9                         ret
1775
1776                    ; go function, jump from monitor program to user program
1777                    ; save system stack and load user stack
1778                    ; load CPU registers with user registers before jump
1779
1780    09AB            go:
1781    09AB 210000                 lxi h,0
1782
1783    09AE 39                     dad sp          ; save system stack
1784    09AF 2257F0                 shld save_stack
1785
1786    09B2 2A34F0                 lhld user_SP    ; get user stack
1787    09B5 F9                     sphl            ; load user stack
1788
1789    09B6 2A2AF0                 lhld user_PC
1790    09B9 E5                     push h
1791    09BA 2A2CF0                 lhld user_AF
1792    09BD E5                     push h
1793    09BE 2A2EF0                 lhld user_BC
1794    09C1 E5                     push h
1795    09C2 2A30F0                 lhld user_DE
1796    09C5 E5                     push h
1797    09C6 2A32F0                 lhld user_HL
1798    09C9 E5                     push h
1799
1800    09CA E1                     pop h
1801    09CB D1                     pop d
1802    09CC C1                     pop b
1803    09CD F1                     pop psw
1804
1805    09CE C9                     ret             ; jump to user program
1806
1807
1808                    ; single step
1809                    ; load CPU registers with user registers, enable trap signal then jur
1810                    ; program
1811                    ; disassemble line to be executed
1812
1813    09CF            single_step:
1814    09CF 210000                 lxi h,0
1815
1816    09D2 39                     dad sp          ; save system stack
1817    09D3 2257F0                 shld save_stack
1818
1819    09D6 2A34F0                 lhld user_SP
1820    09D9 F9                     sphl            ; load user stack
```

```
1821
1822
1823    09DA 2A2AF0               lhld user_PC   ; get address to be executed
1824    09DD E5                   push h         ; save to stack
1825
1826    09DE 3A25F0               lda uart_found
1827    09E1 FE00                 cpi 0
1828    09E3 CAEF09               jz skip2       ; if no uart, skip disassemble
1829
1830    09E6 CDEA12               call new_line
1831    09E9 CDFB12               call send_tab
1832    09EC CD4A03               call disassemble
1833
1834    09EF           skip2:
1835
1836    09EF E1                   pop h
1837    09F0 222AF0               shld user_PC
1838
1839    09F3 2A2AF0               lhld user_PC
1840    09F6 E5                   push h
1841    09F7 2A2CF0               lhld user_AF
1842    09FA E5                   push h
1843    09FB 2A2EF0               lhld user_BC
1844    09FE E5                   push h
1845    09FF 2A30F0               lhld user_DE
1846    0A02 E5                   push h
1847    0A03 2A32F0               lhld user_HL
1848    0A06 E5                   push h
1849
1850    0A07 E1                   pop h
1851    0A08 D1                   pop d
1852    0A09 C1                   pop b
1853
1854    0A0A 3EBF                 mvi a,0bfh         ; make port_c.6 low to enable trap
1855    0A0C D312                 out system_port_c  ;
1856
1857                   ; now the shift register 74LS164 is running
1858                   ; within 8 ALE, trap will be high, trap will be recorgnized after ins
1859                   ; followed RET was executed
1860
1861    0A0E 00                   nop            ; 1 cycles
1862    0A0F F1                   pop psw        ; 5 cycles
1863    0A10 C9                   ret            ; 3 cycles
1864
1865
1866                   ; set mode to 1
1867
1868    0A11 3E01      address_mode: mvi a,1
1869    0A13 3226F0                  sta entry_mode
1870    0A16 CDCE0A                  call read_memory
1871    0A19 AF                      xra a
1872    0A1A 3228F0                  sta counter2
1873    0A1D C9                      ret
1874
1875    0A1E AF        data_mode:    xra a
1876    0A1F 3226F0                  sta entry_mode
1877    0A22 3227F0                  sta counter1
1878    0A25 CDCE0A                  call read_memory
1879    0A28 C9                      ret
1880
1881    0A29 3E02      function_2nd: mvi a,2
1882    0A2B 3226F0                  sta entry_mode
1883    0A2E 3E77                    mvi a,77h
1884    0A30 3210F0                  sta buffer
1885    0A33 3E38                    mvi a,38h
```

```
0A35 3211F0                 sta buffer+1
0A38 3E78                   mvi a,78h
0A3A 3212F0                 sta buffer+2
0A3D 3E00                   mvi a,0
0A3F 3213F0                 sta buffer+3
0A42 3E00                   mvi a,0
0A44 3214F0                 sta buffer+4
0A47 3E00                   mvi a,0
0A49 3215F0                 sta buffer+5
0A4C C9                     ret


                ; set entry mode to 3
                ; hex data will be used for register modifying

0A4D            modify_register:

0A4D F5                     push psw

0A4E 3E03                   mvi a,3
0A50 3226F0                 sta entry_mode

0A53 3A13F0           lda buffer+3
0A56 F680                   ori 80h
0A58 3213F0                 sta buffer+3

0A5B 3A12F0           lda buffer+2
0A5E F680                   ori 80h
0A60 3212F0                 sta buffer+2

0A63 3A11F0           lda buffer+1
0A66 F680                   ori 80h
0A68 3211F0                 sta buffer+1

0A6B 3A10F0           lda buffer
0A6E F680                   ori 80h
0A70 3210F0                 sta buffer

0A73 F1                     pop psw

0A74 C9                     ret









                ; increment key works with mode0 or mode1 display

0A75 3E00       increment:  mvi a,0
0A77 3226F0                 sta entry_mode  ; switch to data mode
0A7A 3227F0                 sta counter1    ; clear event counter1

0A7D 2A2AF0                 lhld user_PC
0A80 23                     inx  h
0A81 222AF0                 shld user_PC
0A84 CDCE0A                 call read_memory
0A87 C9                     ret

                ; decrement key works with mode0 or mode1 display

0A88 3E00       decrement:  mvi a,0
0A8A 3226F0                sta entry_mode  ; switch to data mode
```

```
0A8D 2A2AF0               lhld user_PC
0A90 2B                   dcx  h
0A91 222AF0               shld user_PC
0A94 CDCE0A               call read_memory
0A97 C9                   ret


                  ; convert nibble 0-F to 8-bit seven segment code
                  ; entry: A
                  ; exit: A

0A98              to_seven_segment:

0A98 E60F                 ani 0fh       ; get only low nibble as the index
0A9A 218E1D               lxi h,convert
0A9D 5F                   mov e,a
0A9E 1600                 mvi d,0
0AA0 19                   dad d
0AA1 7E                   mov a,m        ; get code
0AA2 C9                   ret

                  ; convert [HL] to display buffer 0-3
                  ; for register display
                  ; entry: HL

0AA3              read_register:
0AA3 E5                    push h
0AA4 7C                    mov a,h
0AA5 F5                    push psw
0AA6 0F                    rrc
0AA7 0F                    rrc
0AA8 0F                    rrc
0AA9 0F                    rrc
0AAA CD980A                call to_seven_segment
0AAD 3210F0                sta buffer

0AB0 F1                    pop psw
0AB1 CD980A                call to_seven_segment
0AB4 3211F0                sta buffer+1

0AB7 E1                    pop h

0AB8 E5                    push h

0AB9 7D                    mov a,l
0ABA F5                    push psw
0ABB 0F                    rrc
0ABC 0F                    rrc
0ABD 0F                    rrc
0ABE 0F                    rrc
0ABF CD980A                call to_seven_segment
0AC2 3212F0                sta buffer+2
0AC5 F1                    pop psw
0AC6 CD980A                call to_seven_segment
0AC9 3213F0                sta buffer+3

0ACC E1                    pop h
0ACD C9                    ret


                  ; convert current address and data to display buffer
                  ;

0ACE 2A2AF0       read_memory: lhld user_PC
```

```
0AD1 E5                 push h
0AD2 7C                 mov a,h
0AD3 F5                 push psw
0AD4 0F                 rrc
0AD5 0F                 rrc
0AD6 0F                 rrc
0AD7 0F                 rrc
0AD8 CD980A             call to_seven_segment
0ADB 3210F0             sta buffer

0ADE F1                 pop psw
0ADF CD980A             call to_seven_segment
0AE2 3211F0             sta buffer+1

0AE5 E1                 pop h

0AE6 E5                 push h

0AE7 7D                 mov a,l
0AE8 F5                 push psw
0AE9 0F                 rrc
0AEA 0F                 rrc
0AEB 0F                 rrc
0AEC 0F                 rrc
0AED CD980A             call to_seven_segment
0AF0 3212F0             sta buffer+2
0AF3 F1                 pop psw
0AF4 CD980A             call to_seven_segment
0AF7 3213F0             sta buffer+3

0AFA E1                 pop h
0AFB 7E                 mov a,m   ; read from memory

0AFC F5                 push psw
0AFD 0F                 rrc
0AFE 0F                 rrc
0AFF 0F                 rrc
0B00 0F                 rrc
0B01 CD980A             call to_seven_segment
0B04 3214F0             sta buffer+4
0B07 F1                 pop psw


0B08 CD980A             call to_seven_segment
0B0B 3215F0             sta buffer+5

0B0E CD120B             call mode_indicator

0B11 C9                 ret

0B12            mode_indicator:

0B12 F5                 push psw

0B13 3A26F0             lda entry_mode
0B16 FE00               cpi 0
0B18 C24D0B             jnz mode1

0B1B 3A15F0             lda buffer+5   ; mode 0 indicator
0B1E F680               ori 80h
0B20 3215F0             sta buffer+5

0B23 3A14F0             lda buffer+4
0B26 F680           ori 80h
0B28 3214F0         sta buffer+4
```

```
2081
2082
2083     0B2B 3A13F0                lda buffer+3
2084     0B2E E67F                  ani 7fh
2085     0B30 3213F0                sta buffer+3
2086
2087     0B33 3A12F0                lda buffer+2
2088     0B36 E67F            ani 7fh
2089     0B38 3212F0          sta buffer+2
2090
2091     0B3B 3A11F0          lda buffer+1
2092     0B3E E67F            ani 7fh
2093     0B40 3211F0          sta buffer+1
2094
2095     0B43 3A10F0          lda buffer
2096     0B46 E67F            ani 7fh
2097     0B48 3210F0          sta buffer
2098
2099
2100
2101
2102
2103     0B4B F1                    pop psw
2104     0B4C C9                    ret
2105
2106     0B4D FE01      mode1:      cpi 1
2107     0B4F C2840B                jnz mode2
2108
2109     0B52 3A15F0                lda buffer+5   ; mode 1 indicator
2110     0B55 E67F                  ani 7fh
2111     0B57 3215F0                sta buffer+5
2112
2113     0B5A 3A14F0          lda buffer+4
2114     0B5D E67F            ani 7fh
2115     0B5F 3214F0          sta buffer+4
2116
2117     0B62 3A13F0                lda buffer+3
2118     0B65 F680                  ori 80h
2119     0B67 3213F0                sta buffer+3
2120
2121     0B6A 3A12F0          lda buffer+2
2122     0B6D F680                  ori 80h
2123     0B6F 3212F0                sta buffer+2
2124
2125     0B72 3A11F0          lda buffer+1
2126     0B75 F680                  ori 80h
2127     0B77 3211F0                sta buffer+1
2128
2129     0B7A 3A10F0          lda buffer
2130     0B7D F680                  ori 80h
2131     0B7F 3210F0                sta buffer
2132
2133
2134
2135
2136     0B82 F1                    pop psw
2137     0B83 C9                    ret
2138
2139     0B84 F1        mode2:      pop psw
2140     0B85 C9                    ret
2141
2142     0B86 0E07      cold_boot: mvi c,7
2143     0B88 21A40B                lxi h,title
2144
2145
```

```
0B8B 1650      cold2:     mvi d,50h

0B8D CDBE0C    cold1:     call scan
0B90 15                   dcr d
0B91 C28D0B               jnz cold1

0B94 23                   inx h
0B95 0D                   dcr c
0B96 C28B0B               jnz cold2

0B99 2B                   dcx h

0B9A 0E00                 mvi c,0
0B9C CDBE0C    cold3:     call scan
0B9F 0D                   dcr c
0BA0 C29C0B               jnz cold3

0BA3 C9                   ret

0BA4 0000000000title:   dfb 0,0,0,0,0,0,7fh,3fh,7fh,6dh,0,0


                ; display data read from memory pointed to by HL on LED
                ; entry: HL
                ;

0BB0 1605      demo:      mvi d,5

0BB2 CDBE0C    demo1_2:   call scan
0BB5 15                   dcr d
0BB6 C2B20B               jnz demo1_2
0BB9 23                   inx h
0BBA C9                   ret


                ; convert position key to internal key code 0-F for data entry and 1(
                ; function keys
                ; entry: A = scan code
                ; exit: A  = internal code

0BBB FE02      get_key_code:  cpi 2
0BBD C2C30B                   jnz code1
0BC0 3E00                     mvi a,0
0BC2 C9                       ret

0BC3 FE0A      code1:         cpi 0ah
0BC5 C2CB0B                   jnz code2
0BC8 3E01                     mvi a,1
0BCA C9                       ret

0BCB FE12      code2:         cpi 12h
0BCD C2D30B                   jnz code3
0BD0 3E02                     mvi a,2
0BD2 C9                       ret

0BD3 FE1A      code3:         cpi 1ah
0BD5 C2DB0B                   jnz code4
0BD8 3E03                     mvi a,3
0BDA C9                       ret

0BDB FE03      code4:         cpi 3
0BDD C2E30B                   jnz code5
0BE0 3E04                     mvi a,4
0BE2 C9                       ret
```

```
0BE3 FE0B      code5:         cpi 0bh
0BE5 C2EB0B                   jnz code6
0BE8 3E05                     mvi a,5
0BEA C9                       ret

0BEB FE13      code6:         cpi 13h
0BED C2F30B                   jnz code7
0BF0 3E06                     mvi a,6
0BF2 C9                       ret

0BF3 FE1B      code7:         cpi 1bh
0BF5 C2FB0B                   jnz code8
0BF8 3E07                     mvi a,7
0BFA C9                       ret

0BFB FE04      code8:         cpi 4
0BFD C2030C                   jnz code9
0C00 3E08                     mvi a,8
0C02 C9                       ret

0C03 FE0C      code9:         cpi 0ch
0C05 C20B0C                   jnz code10
0C08 3E09                     mvi a,9
0C0A C9                       ret

0C0B FE14      code10:        cpi 14h
0C0D C2130C                   jnz code11
0C10 3E0A                     mvi a,0ah
0C12 C9                       ret

0C13 FE1C      code11:        cpi 1ch
0C15 C21B0C                   jnz code12
0C18 3E0B                     mvi a,0bh
0C1A C9                       ret

0C1B FE05      code12:        cpi 5
0C1D C2230C                   jnz code13
0C20 3E0C                     mvi a,0ch
0C22 C9                       ret

0C23 FE0D      code13:        cpi 0dh
0C25 C22B0C                   jnz code14
0C28 3E0D                     mvi a,0dh
0C2A C9                       ret

0C2B FE15      code14:        cpi 15h
0C2D C2330C                   jnz code15
0C30 3E0E                     mvi a,0eh
0C32 C9                       ret

0C33 FE1D      code15:        cpi 1dh
0C35 C23B0C                   jnz code16
0C38 3E0F                     mvi a,0fh
0C3A C9                       ret

0C3B FE10      code16:        cpi 10h
0C3D C2430C                   jnz code17
0C40 3E10                     mvi a,10h
0C42 C9                       ret

0C43 FE18      code17:        cpi 18h
0C45 C24B0C                   jnz code18
0C48 3E11                     mvi a,11h
0C4A C9                       ret
```

```
0C4B FE01      code18:        cpi 1
0C4D C2530C                   jnz code19
0C50 3E12                     mvi a,12h
0C52 C9                       ret

0C53 FE00      code19:        cpi 0
0C55 C25B0C                   jnz code20
0C58 3E13                     mvi a,13h
0C5A C9                       ret

0C5B FE08      code20:        cpi 8
0C5D C2630C                   jnz code21
0C60 3E14                     mvi a,14h
0C62 C9                       ret

0C63 FE09      code21:        cpi 9
0C65 C26B0C                   jnz code22
0C68 3E15                     mvi a,15h
0C6A C9                       ret

0C6B FE11      code22:        cpi 11h
0C6D C2730C                   jnz code23
0C70 3E16                     mvi a,16h
0C72 C9                       ret

0C73 FE19      code23:        cpi 19h
0C75 C27B0C                   jnz code24
0C78 3E17                     mvi a,17h
0C7A C9                       ret

0C7B FE2E      code24:        cpi 2eh
0C7D C2830C                   jnz code25
0C80 3E18                     mvi a,18h
0C82 C9                       ret

0C83 FE2F      code25:        cpi 2fh
0C85 C28B0C                   jnz code26
0C88 3E19                     mvi a,19h
0C8A C9                       ret

0C8B 3EFF      code26:        mvi a,0ffh
0C8D C9                       ret

               ; scan display and keyboard unitl key was pressed

0C8E           scan_key:  ; mvi d,50       ; number of loop for timeout if key stil

0C8E           scan_key4:  ;push d         ; save d

0C8E 2110F0                lxi h,buffer
0C91 CDBE0C                call scan
0C94 3A21F0                lda key
0C97 FEFF                  cpi 0ffh
0C99 C28E0C                jnz  scan_key4  ; loop if key still pressed
                           ;pop  d

0C9C F29F0C         jp  scan_key3

0C9F           scan_key2:  ;pop d

                   ; dcr d

                   ; jp scan_key4   ; no repeat function

                                          ; repeat if still pressed when timeout
```

```
2341
2342
2343    0C9F CDB70C    scan_key3:  call debounce  ; debounce after released
2344
2345    0CA2 2110F0                lxi h,buffer
2346    0CA5 CDBE0C    scan_key1:  call scan
2347    0CA8 3A21F0                lda key
2348    0CAB FEFF                  cpi 0ffh
2349    0CAD CAA50C                jz  scan_key1  ; loop until key will be pressed
2350
2351    0CB0 CDB70C                call debounce
2352
2353
2354    0CB3 CDBB0B                call get_key_code
2355
2356                   ;            call out2x
2357    0CB6 C9                    ret
2358
2359
2360    0CB7 0664      debounce:   mvi b,100       ; 20
2361    0CB9 05        debounce1:  dcr b
2362    0CBA C2B90C                jnz debounce1
2363    0CBD C9                    ret
2364
2365
2366                   ; subroutine scan keyboard and display
2367                   ; input: hl pointer to buffer
2368                   ; exit: key = scan code
2369                   ;            -1 no key pressed
2370                   ;
2371
2372    0CBE E5        scan:    push h
2373    0CBF C5                 push b
2374    0CC0 D5                 push d
2375
2376    0CC1 0E06               mvi c,6    ; for 6-digit LED
2377    0CC3 1E00               mvi e,0    ; digit scan code appears at 4-to-10 decoder
2378    0CC5 1600               mvi d,0    ; key position
2379    0CC7 3EFF               mvi a,0ffh ; put -1 to key
2380    0CC9 3221F0             sta key    ; key = -1
2381
2382
2383    0CCC 7B        scan1:   mov a,e
2384    0CCD F6F0               ori 0f0h   ; high nibble must be 1111
2385    0CCF D312               out system_port_c ; active digit first
2386    0CD1 7E                 mov a,m        ; load a with [hl]
2387    0CD2 D311               out system_port_b ; then turn segment on
2388
2389    0CD4 060A               mvi b,10    ; delay for transition process
2390    0CD6 05        wait1:   dcr b
2391    0CD7 C2D60C             jnz wait1
2392
2393    0CDA AF           xra a
2394    0CDB D311         out system_port_b ; turn off segment
2395
2396
2397    0CDD DB10               in  system_port_a  ; read input port
2398
2399    0CDF 0608               mvi b,8            ; check all 8-row
2400    0CE1 1F        shift_key: rar              ; rotate right through carry
2401    0CE2 DAEB0C             jc  next_key       ; if carry = 1 then no key pressed
2402
2403    0CE5 F5                 push psw
2404    0CE6 7A                 mov a,d
2405    0CE7 3221F0             sta key            ; save key position
```

```
0CEA F1                 pop psw

0CEB            next_key:
0CEB 14                 inr d               ; next key position

0CEC 05                 dcr b               ; until 8-bit was shifted
0CED C2E10C             jnz shift_key

                        ; mvi a,0            ; clear a
                        ; out system_port_b   ; turn off led

0CF0 1C                 inr e               ; next digit scan code
0CF1 23                 inx h               ; next location

0CF2 0D                 dcr c               ; next column
0CF3 C2CC0C             jnz scan1

0CF6 CDFD0C             call serial_command

0CF9 D1                 pop d
0CFA C1                 pop b
0CFB E1                 pop h
0CFC C9                 ret


                ;----------- serial commands with 9600 8n1 terminal ----------------
                ; check if serial buffer has command
                ;

0CFD            serial_command:
0CFD 3A25F0             lda uart_found
0D00 FE00               cpi 0
0D02 CA410D             jz  skip_serial

0D05 CD5512             call get_command
0D08 CD600D             call download
0D0B CD420D             call prompting
0D0E CD0113             call hex_dump
0D11 CDFE11             call help
0D14 CDD711             call quick_home
0D17 CDEC11             call io_address
0D1A CDBA11             call new_location
0D1D CD3711             call edit_location
0D20 CDED10             call jump_to_user_pgm
0D23 CDD410             call monitor_function
0D26 CD820D             call ascii_print
0D29 CD9010             call fill_memory
0D2C CD5C0F             call register_display
0D2F CD1C0F             call stack_display
0D32 CD2303             call disassemble1
0D35 CD100F             call single_step_
0D38 CD010F             call print_watch
0D3B CDE60E             call clear_watch
0D3E CDB40D             call set_user_register


0D41            skip_serial:

0D41 C9                 ret

0D42 3A22F0     prompting: lda command
0D45 FE0D                  cpi cr
0D47 C25F0D                jnz exit_prompting
```

```
2471
2472    0D4A            send_prompt:
2473
2474    0D4A CDEA12                 call new_line
2475    0D4D 2A3CF0                 lhld pointer        ; user_PC
2476    0D50 7C                     mov a,h
2477    0D51 CDDB12                 call out2x
2478    0D54 7D                     mov a,l
2479    0D55 CDDB12                 call out2x
2480    0D58 21DD1D                 lxi h,prompt_text
2481    0D5B CD6812                 call put_str
2482    0D5E C9                     ret
2483
2484    0D5F C9         exit_prompting: ret
2485
2486
2487
2488                    ; command execute
2489                    ; get command from serial port
2490
2491    0D60 3A22F0     download: lda command
2492    0D63 FE6C                 cpi "l"
2493    0D65 C2810D               jnz  exit_download
2494
2495    0D68 CDE514               call clear_bcd1     ; reset bcd counter1
2496    0D6B 3E01                 mvi a,1
2497    0D6D 323AF0               sta temp
2498
2499    0D70 AF                   xra a
2500    0D71 3220F0               sta bcs             ; clear byte chekc sum error
2501
2502    0D74 21C61D               lxi h,download_text
2503    0D77 CD6812               call put_str
2504    0D7A CD1314               call get_record
2505    0D7D CD4A0D               call send_prompt
2506    0D80 C9                   ret
2507
2508    0D81 C9         exit_download ret
2509
2510                    ; display printable ASCII code, 20H-7FH
2511
2512    0D82 3A22F0     ascii_print: lda command
2513    0D85 FE61                  cpi "a"
2514    0D87 C2B30D                jnz exit_ascii_print
2515
2516
2517    0D8A 21571E                lxi h, ascii_text
2518    0D8D CD6812                call put_str
2519
2520    0D90 CDEA12                call new_line
2521    0D93 CDEA12                call new_line
2522
2523    0D96 2E20                  mvi l,20h
2524    0D98 0E60                  mvi c,96
2525
2526    0D9A            ascii_print1:
2527
2528    0D9A 7D                    mov a,l
2529    0D9B CD3F12                call cout
2530
2531    0D9E 3E3D                  mvi a,"="
2532    0DA0 CD3F12                call cout
2533    0DA3 7D                    mov a,l
2534    0DA4 CDDB12                call out2x
2535    0DA7 CDF512                call space
```

```
0DAA 2C                 inr l
0DAB 0D                 dcr c
0DAC C29A0D             jnz ascii_print1

0DAF CD4A0D             call send_prompt
0DB2 C9                 ret

0DB3            exit_ascii_print:

0DB3 C9                 ret

                ;------------------- set value to user registers --------------------
                ; set value to user register AF, BC, DE, HL, SP, PC

0DB4            set_user_register:

0DB4 3A22F0             lda command
0DB7 FE73               cpi "s"
0DB9 C2C80E             jnz exit_set_user

0DBC 21051F             lxi h, set_register_text
0DBF CD6812             call put_str

0DC2 CD4B12             call cin
0DC5 FE61               cpi "a"
0DC7 C2F00D             jnz set_user1

0DCA CDEA12             call new_line
0DCD 21A91E             lxi h,af_text
0DD0 CD6812             call put_str
0DD3 2A2CF0             lhld user_AF
0DD6 7C                 mov a,h
0DD7 CDDB12             call out2x
0DDA 7D                 mov a,l
0DDB CDDB12             call out2x
0DDE CDFB12             call send_tab
0DE1 CDAE13             call get_hex2
0DE4 67                 mov h,a
0DE5 CDAE13             call get_hex2
0DE8 6F                 mov l,a
0DE9 222CF0             shld user_AF

0DEC CD4A0D             call send_prompt
0DEF C9                 ret

0DF0            set_user1:
0DF0 FE62               cpi "b"
0DF2 C21B0E             jnz set_user2

0DF5 CDEA12             call new_line
0DF8 21AD1E             lxi h,bc_text
0DFB CD6812             call put_str
0DFE 2A2EF0             lhld user_BC
0E01 7C                 mov a,h
0E02 CDDB12             call out2x
0E05 7D                 mov a,l
0E06 CDDB12             call out2x
0E09 CDFB12             call send_tab
0E0C CDAE13             call get_hex2
0E0F 67                 mov h,a
0E10 CDAE13             call get_hex2
0E13 6F                 mov l,a
0E14 222EF0             shld user_BC

0E17 CD4A0D             call send_prompt
```

```
2601    0E1A C9                 ret
2602
2603    0E1B            set_user2:
2604    0E1B FE64               cpi "d"
2605    0E1D C2460E             jnz set_user3
2606
2607    0E20 CDEA12             call new_line
2608    0E23 21B11E             lxi h,de_text
2609    0E26 CD6812             call put_str
2610    0E29 2A30F0             lhld user_DE
2611    0E2C 7C                 mov a,h
2612    0E2D CDDB12             call out2x
2613    0E30 7D                 mov a,l
2614    0E31 CDDB12             call out2x
2615    0E34 CDFB12             call send_tab
2616    0E37 CDAE13             call get_hex2
2617    0E3A 67                 mov h,a
2618    0E3B CDAE13             call get_hex2
2619    0E3E 6F                 mov l,a
2620    0E3F 2230F0             shld user_DE
2621
2622    0E42 CD4A0D             call send_prompt
2623    0E45 C9                 ret
2624
2625    0E46            set_user3:
2626    0E46 FE68               cpi "h"
2627    0E48 C2710E             jnz set_user4
2628
2629    0E4B CDEA12             call new_line
2630    0E4E 21B51E             lxi h,hl_text
2631    0E51 CD6812             call put_str
2632    0E54 2A32F0             lhld user_HL
2633    0E57 7C                 mov a,h
2634    0E58 CDDB12             call out2x
2635    0E5B 7D                 mov a,l
2636    0E5C CDDB12             call out2x
2637    0E5F CDFB12             call send_tab
2638    0E62 CDAE13             call get_hex2
2639    0E65 67                 mov h,a
2640    0E66 CDAE13             call get_hex2
2641    0E69 6F                 mov l,a
2642    0E6A 2232F0             shld user_HL
2643
2644    0E6D CD4A0D             call send_prompt
2645    0E70 C9                 ret
2646
2647    0E71            set_user4:
2648    0E71 FE73               cpi "s"
2649    0E73 C29C0E             jnz set_user5
2650
2651    0E76 CDEA12             call new_line
2652    0E79 21B91E             lxi h,sp_text
2653    0E7C CD6812             call put_str
2654    0E7F 2A34F0             lhld user_SP
2655    0E82 7C                 mov a,h
2656    0E83 CDDB12             call out2x
2657    0E86 7D                 mov a,l
2658    0E87 CDDB12             call out2x
2659    0E8A CDFB12             call send_tab
2660    0E8D CDAE13             call get_hex2
2661    0E90 67                 mov h,a
2662    0E91 CDAE13             call get_hex2
2663    0E94 6F                 mov l,a
2664    0E95 2234F0             shld user_SP
2665
```

```
0E98 CD4A0D             call send_prompt
0E9B C9                 ret

0E9C            set_user5:
0E9C FE70               cpi "p"
0E9E C2C70E             jnz set_user6

0EA1 CDEA12             call new_line
0EA4 21C21E             lxi h,pc_text
0EA7 CD6812             call put_str
0EAA 2A2AF0             lhld user_PC
0EAD 7C                 mov a,h
0EAE CDDB12             call out2x
0EB1 7D                 mov a,l
0EB2 CDDB12             call out2x
0EB5 CDFB12             call send_tab
0EB8 CDAE13             call get_hex2
0EBB 67                 mov h,a
0EBC CDAE13             call get_hex2
0EBF 6F                 mov l,a
0EC0 222AF0             shld user_PC

0EC3 CD4A0D             call send_prompt
0EC6 C9                 ret

0EC7 C9         set_user6: ret

0EC8            exit_set_user:

0EC8 C9                  ret









0EC9            print_watch_ram:

0EC9 CDEA12             call new_line
0ECC 2100F0             lxi h, watch_ram
0ECF 0E10               mvi c,16
0ED1 7C                 mov a,h
0ED2 CDDB12             call out2x
0ED5 7D                 mov a,l
0ED6 CDDB12             call out2x
0ED9            watch1:
0ED9 CDF512             call space
0EDC 7E                 mov a,m
0EDD CDDB12             call out2x
0EE0 23                 inx h
0EE1 0D                 dcr c
0EE2 C2D90E             jnz watch1
0EE5 C9                 ret



                ;----------- clear watch variables---------------------------------

0EE6 3A22F0     clear_watch: lda command
0EE9 FE63                    cpi "c"
0EEB C2000F                  jnz exit_clear_watch
```

```
0EEE 2100F0               lxi h, watch_ram
0EF1 0E10                 mvi c,16

0EF3 AF        clear1:    xra a
0EF4 77                   mov m,a
0EF5 23                   inx h
0EF6 0D                   dcr c
0EF7 C2F30E               jnz clear1

0EFA CDC90E               call print_watch_ram
0EFD CD4A0D               call send_prompt

0F00 C9        exit_clear_watch: ret









               ;----------- print watch variables ---------------------------------

0F01           print_watch:

0F01 3A22F0                lda command
0F04 FE77                  cpi "w"
0F06 C20F0F                jnz exit_watch

0F09 CDC90E                call print_watch_ram
0F0C CD4A0D                call send_prompt

0F0F C9        exit_watch: ret


               ;---------- single step running with key space ----------------------
0F10           single_step_:

0F10 3A22F0              lda command
0F13 FE20                cpi " "
0F15 C21B0F              jnz exit_step
0F18 CDCF09              call single_step

0F1B           exit_step:
0F1B C9                  ret

               ;------- display stack area from top of stack to initial ------------

0F1C           stack_display:
0F1C 3A22F0              lda command
0F1F FE6B                cpi "k"
0F21 C25B0F              jnz exit_stack

               ;          lxi h,stack_text
               ;          call put_str

0F24 21321E              lxi h,edit_text2
0F27 CD6812              call put_str
0F2A CDEA12              call new_line

0F2D 2A34F0              lhld user_SP

0F30           stack_display1:

0F30 7C                  mov a,h
0F31 CDDB12              call out2x
0F34 7D                  mov a,l
```

```
0F35 CDDB12             call out2x

0F38 CDF512             call space
0F3B CDF512             call space
0F3E 3E5B               mvi a,"["
0F40 CD3F12             call cout

0F43 7E                 mov a,m
0F44 CDDB12             call out2x

0F47 3E5D               mvi a,"]"
0F49 CD3F12             call cout

0F4C CDEA12             call new_line

0F4F 23                 inx h

0F50 119AF0             lxi d, user_stack+32+1  ; load base of user stack

0F53 7D                 mov a,l
0F54 AB                 xra e
0F55 C2300F             jnz stack_display1

0F58 CD4A0D             call send_prompt

0F5B            exit_stack:
0F5B C9                 ret



                ;---------------- registers display ---------------------------------

0F5C            register_display:

0F5C 3A22F0              lda command
0F5F FE72                cpi "r"
0F61 C28110              jnz exit_register

0F64            register_display1:

                ;            lda uart_found
                ;            cpi 0
                ;            jz exit_register ; exit of no uart


0F64 CDEA12              call new_line

0F67            register_display2:
0F67 CDEA12              call new_line

0F6A 21A91E              lxi h,af_text
0F6D CD6812              call put_str
0F70 2A2CF0              lhld user_AF
0F73 7C                  mov a,h
0F74 CDDB12              call out2x
0F77 7D                  mov a,l
0F78 CDDB12              call out2x
0F7B CDF512              call space

0F7E 21AD1E              lxi h,bc_text
0F81 CD6812              call put_str
0F84 2A2EF0              lhld user_BC
0F87 7C                  mov a,h
0F88 CDDB12              call out2x
0F8B 7D                  mov a,l
```

```
2861    0F8C CDDB12             call out2x
2862
2863    0F8F CDF512             call space
2864
2865    0F92 21B11E             lxi h,de_text
2866    0F95 CD6812             call put_str
2867    0F98 2A30F0             lhld user_DE
2868    0F9B 7C                 mov a,h
2869    0F9C CDDB12             call out2x
2870    0F9F 7D                 mov a,l
2871    0FA0 CDDB12             call out2x
2872    0FA3 CDF512             call space
2873
2874    0FA6 21B51E             lxi h,hl_text
2875    0FA9 CD6812             call put_str
2876    0FAC 2A32F0             lhld user_HL
2877    0FAF 7C                 mov a,h
2878    0FB0 CDDB12             call out2x
2879    0FB3 7D                 mov a,l
2880    0FB4 CDDB12             call out2x
2881
2882    0FB7 CDF512             call space
2883
2884    0FBA 21B91E             lxi h,sp_text
2885    0FBD CD6812             call put_str
2886    0FC0 2A34F0             lhld user_SP
2887    0FC3 7C                 mov a,h
2888    0FC4 CDDB12             call out2x
2889    0FC7 7D                 mov a,l
2890    0FC8 CDDB12             call out2x
2891
2892    0FCB CDF512             call space
2893
2894                            ;  lxi h,tos_text
2895                            ;  call put_str
2896                            ;  lhld tos
2897                            ;  mov a,h
2898                            ;  call out2x
2899                            ;  mov a,l
2900                            ;  call out2x
2901                            ;  call space
2902
2903    0FCE 21C21E             lxi h,pc_text
2904    0FD1 CD6812             call put_str
2905    0FD4 2A2AF0             lhld user_PC
2906    0FD7 7C                 mov a,h
2907    0FD8 CDDB12             call out2x
2908    0FDB 7D                 mov a,l
2909    0FDC CDDB12             call out2x
2910
2911    0FDF CDF512             call space
2912
2913    0FE2 21DC1E             lxi h,sign_text
2914    0FE5 CD6812             call put_str
2915    0FE8 2A2CF0             lhld user_AF
2916    0FEB 7D                 mov a,l
2917    0FEC E680               ani 80h
2918    0FEE C2F90F             jnz register_flag1
2919    0FF1 3E30               mvi a,"0"
2920    0FF3 CD3F12             call cout
2921    0FF6 C3FE0F             jmp register_flag2
2922
2923    0FF9            register_flag1:
2924    0FF9 3E31               mvi a,"1"
2925    0FFB CD3F12             call cout
```

```
2926
2927    0FFE            register_flag2:
2928    0FFE CDF512                call space
2929
2930    1001 218210                lxi h,zero_text
2931    1004 CD6812                call put_str
2932    1007 2A2CF0                lhld user_AF
2933    100A 7D                    mov a,l
2934    100B E640                  ani 40h
2935    100D C21810                jnz register_flag3
2936    1010 3E30                  mvi a,"0"
2937    1012 CD3F12                call cout
2938    1015 C31D10                jmp register_flag4
2939
2940    1018            register_flag3:
2941    1018 3E31                  mvi a,"1"
2942    101A CD3F12                call cout
2943
2944    101D            register_flag4:
2945    101D CDF512                call space
2946
2947    1020 218510                lxi h,AC_text
2948    1023 CD6812                call put_str
2949    1026 2A2CF0                lhld user_AF
2950    1029 7D                    mov a,l
2951    102A E610                  ani 10h
2952    102C C23710                jnz register_flag5
2953    102F 3E30                  mvi a,"0"
2954    1031 CD3F12                call cout
2955    1034 C33C10                jmp register_flag6
2956
2957    1037            register_flag5:
2958    1037 3E31                  mvi a,"1"
2959    1039 CD3F12                call cout
2960
2961    103C            register_flag6:
2962    103C CDF512                call space
2963
2964    103F 218910                lxi h,P_text
2965    1042 CD6812                call put_str
2966    1045 2A2CF0                lhld user_AF
2967    1048 7D                    mov a,l
2968    1049 E604                  ani 4
2969    104B C25610                jnz register_flag7
2970    104E 3E30                  mvi a,"0"
2971    1050 CD3F12                call cout
2972    1053 C35B10                jmp register_flag8
2973
2974    1056            register_flag7:
2975    1056 3E31                  mvi a,"1"
2976    1058 CD3F12                call cout
2977
2978    105B            register_flag8:
2979    105B CDF512                call space
2980
2981    105E 218C10                lxi h,CY_text
2982    1061 CD6812                call put_str
2983    1064 2A2CF0                lhld user_AF
2984    1067 7D                    mov a,l
2985    1068 E601                  ani 1
2986    106A C27510                jnz register_flag9
2987    106D 3E30                  mvi a,"0"
2988    106F CD3F12                call cout
2989    1072 C37A10                jmp register_flag10
2990
```

```
2991    1075            register_flag9:
2992    1075 3E31                mvi a,"1"
2993    1077 CD3F12              call cout
2994
2995    107A            register_flag10:
2996    107A CDF512              call space
2997
2998    107D CD4A0D              call send_prompt
2999    1080 C9                  ret
3000
3001    1081            exit_register:
3002    1081 C9                  ret
3003
3004    1082 5A3D00     zero_text   dfb "Z=",0
3005    1085 41433D00   AC_text     dfb "AC=",0
3006    1089 503D00     P_text      dfb "P=",0
3007    108C 43593D00   CY_text     dfb "CY=",0
3008
3009
3010                    ;----------------- fill constant to memory -------------------------
3011
3012    1090            fill_memory:
3013
3014    1090 3A22F0              lda command
3015    1093 FE66                cpi "f"
3016    1095 C2D310              jnz exit_fill
3017    1098 21681E              lxi h,fill_text1
3018    109B CD6812              call put_str
3019
3020    109E CD9513              call get_hex1
3021    10A1 67                  mov h,a
3022    10A2 CD9513              call get_hex1
3023    10A5 6F                  mov l,a
3024    10A6 E5                  push h          ; save begin address to stack
3025
3026    10A7 21791E              lxi h,fill_text2
3027    10AA CD6812              call put_str
3028
3029    10AD CD9513              call get_hex1
3030    10B0 67                  mov h,a
3031    10B1 CD9513              call get_hex1
3032    10B4 6F                  mov l,a
3033    10B5 E5                  push h         ; save end address to stack
3034
3035    10B6 21891E              lxi h,fill_text3
3036    10B9 CD6812              call put_str
3037    10BC CD9513              call get_hex1
3038
3039    10BF 47                  mov b,a        ; byte save to B
3040
3041    10C0 D1                  pop d          ; end address in DE
3042
3043    10C1 E1                  pop h          ; begin address in HL
3044
3045    10C2            fill_memory1:
3046
3047    10C2 78                  mov a,b
3048    10C3 77                  mov m,a
3049    10C4 23                  inx h
3050
3051    10C5 7D                  mov a,l
3052    10C6 BB                  cmp e
3053    10C7 C2C210              jnz fill_memory1
3054
3055    10CA 7C                  mov a,h
```

```
10CB BA                 cmp d
10CC C2C210             jnz fill_memory1


10CF CD4A0D             call send_prompt
10D2 C9                 ret

10D3            exit_fill:

10D3 C9                 ret




                ;---------------- monitor function list ----------------------------

10D4            monitor_function:

10D4 3A22F0             lda command
10D7 FE6D               cpi "m"
10D9 C2EC10             jnz exit_monitor

10DC CDEA12             call new_line
10DF 21D421             lxi h,monitor_text
10E2 CD6812             call put_str
10E5 CDEA12             call new_line
10E8 CD4A0D             call send_prompt
10EB C9                 ret

10EC            exit_monitor:
10EC C9                 ret

                ;----------------------- jump to user program --------------------

10ED 3A22F0     jump_to_user_pgm:   lda command
10F0 FE6A                       cpi "j"
10F2 C23611                     jnz exit_jump

10F5 21401E                     lxi h, jump_text1
10F8 CD6812                     call put_str

10FB 2A2AF0                     lhld user_PC
10FE 7C                         mov a,h
10FF CDDB12                     call out2x
1102 7D                         mov a,l
1103 CDDB12                     call out2x

1106 21521E                     lxi h,jump_text2
1109 CD6812                     call put_str

110C CDAE13                     call get_hex2

110F F5                         push psw

1110 3A23F0                     lda flag1
1113 E601                       ani 1
1115 C22711                     jnz skip_load_PC

1118 F1                         pop psw

1119 67                         mov h,a
111A CDAE13                     call get_hex2
111D 6F                         mov l,a
111E 222AF0                     shld user_PC
1121 CDEA12                     call new_line
```

```
3121    1124 C3AB09                 jmp go
3122
3123    1127           skip_load_PC:
3124    1127 F1                     pop psw
3125    1128 3A23F0                 lda flag1
3126    112B E6FE                   ani 0feh
3127    112D 3223F0                 sta flag1
3128    1130 CDEA12                 call new_line
3129    1133 C3AB09                 jmp go
3130
3131    1136 C9        exit_jump:   ret
3132
3133                   ;------------ edit memory ----------------------------------------
3134
3135
3136    1137 3A22F0    edit_location: lda command
3137    113A FE65                   cpi "e"
3138    113C C2B911                 jnz exit_edit
3139
3140    113F 21EF1D                 lxi h, edit_text
3141    1142 CD6812                 call put_str
3142    1145 CD9513                 call get_hex1
3143    1148 67                     mov h,a
3144    1149 CD9513                 call get_hex1
3145    114C 6F                     mov l,a
3146    114D 223CF0                 shld pointer    ;user_PC
3147
3148    1150 21071E                 lxi h, edit_text1
3149    1153 CD6812                 call put_str
3150
3151    1156 21321E                 lxi h, edit_text2
3152    1159 CD6812                 call put_str
3153
3154    115C CDEA12    edit1:       call new_line
3155
3156    115F 2A3CF0                 lhld pointer    ;user_PC
3157    1162 7C                     mov a,h
3158    1163 CDDB12                 call out2x
3159    1166 7D                     mov a,l
3160    1167 CDDB12                 call out2x
3161    116A CDF512                 call space
3162    116D CDF512                 call space
3163    1170 3E5B                   mvi a,"["
3164    1172 CD3F12                 call cout
3165    1175 7E                     mov a,m
3166    1176 CDDB12                 call out2x
3167    1179 3E5D                   mvi a,"]"
3168    117B CD3F12                 call cout
3169
3170    117E CDF512                 call space
3171
3172    1181 CDAE13                 call get_hex2
3173
3174    1184 F5                     push psw
3175
3176    1185 3A23F0                 lda flag1
3177    1188 E601                   ani 1
3178    118A C2AA11                 jnz exit_edit1    ; Enter key?
3179
3180    118D 3A23F0                 lda flag1
3181    1190 E602                   ani 2
3182    1192 C29A11                 jnz skip_edit1    ; SPACE key?
3183
3184    1195 F1                     pop psw
3185
```

```
1196 77                 mov m,a
1197 C3A311             jmp skip_edit2

119A F1       skip_edit1: pop psw

119B 3A23F0             lda flag1
119E E6FD               ani 0fdh
11A0 3223F0             sta flag1

11A3          skip_edit2:
11A3 23                 inx h
11A4 223CF0             shld pointer    ;user_PC
11A7 C35C11             jmp edit1

11AA F1       exit_edit1: pop psw

11AB 3A23F0             lda flag1
11AE E6FE               ani 0feh
11B0 3223F0             sta flag1

11B3 CDEA12             call new_line
11B6 CD4A0D             call send_prompt

11B9 C9       exit_edit:  ret


11BA 3A22F0   new_location: lda command
11BD FE6E               cpi "n"
11BF C2D611             jnz exit_new_location
11C2 21DF1D             lxi h,new_text
11C5 CD6812             call put_str
11C8 CD9513             call get_hex1
11CB 67                 mov h,a
11CC CD9513             call get_hex1
11CF 6F                 mov l,a
11D0 223CF0             shld pointer       ; user_PC
11D3 CD4A0D             call send_prompt

11D6          exit_new_location:

11D6 C9                 ret




11D7          quick_home:
11D7 3A22F0             lda command
11DA FE71               cpi "q"
11DC C2EB11             jnz exit_quick_home

11DF 210081             lxi h,home_address
11E2 222AF0             shld user_PC
11E5 223CF0             shld pointer
11E8 CD4A0D             call send_prompt

11EB          exit_quick_home:
11EB C9                 ret

                        ; i/o address map

11EC 3A22F0   io_address:  lda command
11EF FE69               cpi "i"
11F1 C2FD11             jnz exit_io

11F4 21EC20             lxi h,io_text
```

```
3251    11F7 CD6812             call put_str
3252    11FA CD4A0D             call send_prompt
3253
3254    11FD C9         exit_io:  ret
3255
3256
3257                    ; help listing
3258
3259    11FE 3A22F0     help:     lda command
3260    1201 FE3F                 cpi "?"
3261    1203 C21512               jnz exit_help
3262
3263    1206 21681D               lxi h,prompt1
3264    1209 CD6812               call put_str
3265    120C 21571F               lxi h,help_text1
3266    120F CD6812               call put_str
3267    1212 CD4A0D               call send_prompt
3268
3269    1215 C9         exit_help:  ret
3270
3271
3272                    ; initialize 16C550 uart to 9600 8n1 with 2MHz clock
3273                    ; 2MHz/13 = 153846Hz
3274
3275    1216            init_uart:
3276
3277    1216 3E83               mvi a,83h
3278    1218 D343               out uart_lcr          ; set DLAB bit to access divider
3279
3280    121A 3E0D               mvi a,13
3281    121C D340               out uart_divisor_lsb
3282    121E 3E00               mvi a,0
3283    1220 D341               out uart_divisor_msb ; 2MHz/13 = 153846 Hz
3284                                                  ; 153846Hz/16 = 9615Hz
3285    1222 3E07               mvi a,7
3286    1224 D342               out uart_fifo         ; init fifo and clear all buffers
3287    1226 3E03               mvi a,03h
3288    1228 D343               out uart_lcr          ; clar DLAB
3289
3290                    ; check uart line status, if the byte is FF then no uart
3291                    ;
3292                    ;
3293    122A AF                 xra a
3294    122B D347               out uart_scr        ; check if there is uart
3295    122D DB47               in uart_scr
3296    122F FE00               cpi 0
3297    1231 CA3912             jz found
3298    1234 AF                 xra a
3299    1235 3225F0             sta uart_found
3300    1238 C9                 ret
3301
3302    1239 3E01       found   mvi a,1
3303    123B 3225F0             sta uart_found
3304    123E C9                 ret
3305
3306    123F 47         cout:   mov b,a               ; save a
3307
3308    1240 DB45       cout1:  in uart_line_status
3309    1242 E620               ani 20h               ; transmitter ready?
3310    1244 CA4012             jz cout1
3311
3312    1247 78                 mov a,b               ; restore a
3313    1248 D340               out uart_buffer
3314    124A C9                 ret
3315
```

```
124B DB45      cin:    in uart_line_status
124D E601              ani 1                   ; data available?
124F CA4B12            jz  cin
1252 DB40              in uart_buffer
1254 C9                ret


1255 DB45      get_command: in uart_line_status
1257 E601                   ani 1
1259 CA6212                 jz no_data
125C DB40                   in uart_buffer
125E 3222F0                 sta command      ; command = ASCII code
1261 C9                     ret

1262 3EFF      no_data:     mvi a,0ffh       ; command == -1
1264 3222F0                 sta command
1267 C9                     ret


                ; print string terminated by 0
                ; input: HL

1268 7E        put_str:  mov a,m    ; get A from [HL]
1269 FE00                cpi 0
126B C26F12              jnz put_str1
126E C9                  ret

126F CD3F12    put_str1: call cout
1272 23                  inx h
1273 F26812              jp put_str

1276 7E        alt_put_str:  mov a,m    ; get A from [HL]
1277 EEAA                xri 0aah
1279 FE00                cpi 0
127B C2B512              jnz put_str2
127E C9                  ret

127F A7A0A0E7FEprompt3:    dfb 0A7h,0A0h,0A0h,0E7h,0FEh,0E1h,087h,092h,09Fh,08Ah,09
128F E3E9F8E5FA            dfb 0E3h,0E9h,0F8h,0E5h,0FAh,0F8h,0E5h,0E9h,0EFh,0F9h,0F
129F EBE3E4E3E4            dfb 0EBh,0E3h,0E4h,0E3h,0E4h,0EDh,08Ah,0E1h,0E3h,0FEh,08
12AF E6FA83A7A0            dfb 0E6h,0FAh,083h,0A7h,0a0h,0aah

12B5 CD3F12    put_str2: call cout
12B8 23                  inx h
12B9 F27612              jp alt_put_str


12BC 21681D    send_prompt1: lxi h,prompt1
12BF CD6812                  call put_str
12C2 C9                      ret

12C3 217F12    send_prompt3: lxi h,prompt3
12C6 CD7612                  call alt_put_str
12C9 C9                      ret


12CA F5        out1x:       push psw
12CB E60F                   ani  0fh
12CD C630                   adi  "0"
12CF FE3A                   cpi  3Ah
12D1 DAD612                 jc  out1x1
12D4 C607                   adi  7

```

```
12D6 CD3F12    out1x1:     call cout
12D9 F1                    pop psw
12DA C9              ret

12DB           out2x:
12DB 0F                    rrc
12DC 0F                    rrc
12DD 0F                    rrc
12DE 0F                    rrc
12DF CDCA12          call out1x
12E2 07                    rlc
12E3 07                    rlc
12E4 07                    rlc
12E5 07                    rlc
12E6 CDCA12          call out1x
12E9 C9              ret

                   ; new_line

12EA 3E0D      new_line:   mvi a,cr
12EC CD3F12         call cout
12EF 3E0A                  mvi a,lf
12F1 CD3F12         call cout
12F4 C9             ret

12F5 3E20      space:      mvi a," "
12F7 CD3F12                call cout
12FA C9                    ret

12FB 3E09      send_tab:   mvi a,9
12FD CD3F12                call cout
1300 C9                    ret



1301 3A22F0    hex_dump:  lda command
1304 FE68                 cpi "h"
1306 C26013               jnz exit_hex_dump

1309 CDEA12               call new_line

130C 0E08                 mvi c,8        ; 8 lines

130E C5        hex_dump2:  push b
130F CDEA12               call new_line
1312 2A3CF0               lhld pointer     ;user_PC
1315 7C                   mov a,h
1316 CDDB12               call out2x
1319 7D                   mov a,l
131A CDDB12               call out2x
131D CDF512               call space

1320 0E10                 mvi c,16

1322 CDF512    hex_dump1:  call space
1325 7E                   mov a,m
1326 CDDB12               call out2x
1329 23                   inx h
132A 0D                   dcr c
132B C22213               jnz hex_dump1

132E CDF512               call space
1331 CDF512               call space
1334 CDF512               call space

```

```
                ; print ASCII representation 20H-7FH
                ; outside such range, print . instead

1337 11F0FF             lxi d,0FFF0h   ; load DE with -16
133A 19                 dad d          ; ADD HL,DE

133B 0E10               mvi c,16

133D 7E        hex_dump5: mov a,m

133E FE20               cpi 20h        ; <20H?
1340 DA4813             jc hex_dump3
1343 FE80               cpi 80h
1345 DA4A13             jc hex_dump4
1348 3E2E      hex_dump3: mvi a,"."
134A CD3F12    hex_dump4: call cout

134D 23                 inx h
134E 0D                 dcr c
134F C23D13             jnz hex_dump5

1352 223CF0             shld pointer   ;user_PC

1355 C1                 pop  b
1356 0D                 dcr c
1357 C20E13             jnz hex_dump2

135A CDEA12             call new_line
135D CD4A0D             call send_prompt

1360 C9        exit_hex_dump:  ret


1361 210081    dump_memory: lxi h,8100h
1364 0E64                   mvi c,100   ; 100 bytes display

1366 CDEA12                 call new_line
1369 7E        dump1:       mov a,m
136A CDDB12          call out2x
136D CDF512          call space
1370 23                     inx h
1371 0D                     dcr c
1372 C26913                 jnz dump1
1375 C9              ret

                ; convert ASCII letter to one nibble 0-F
                ; 0-9 -> al-30
                ; A-F -> al-7
                ; entry: A
                ; exit: A

1376 D630      to_hex:  sui "0"
1378 FE0A               cpi 10
137A DA8113             jc zero_nine
137D E6DF               ani 11011111b
137F D607               sui 7        ; convert to A-F
1381           zero_nine:

1381 C9                 ret

                ; read two ASCII bytes and convert them to one bye 8-bit data
                ; exit: A
                ; used: A, E

1382 CD4B12    get_hex: call cin
```

```
3511    1385 CD7613      call to_hex
3512    1388 0F                 rrc
3513    1389 0F                 rrc
3514    138A 0F                 rrc
3515    138B 0F                 rrc
3516    138C 5F                 mov e,a
3517    138D CD4B12             call cin
3518    1390 CD7613      call to_hex
3519    1393 83                 add e
3520    1394 C9          ret
3521
3522                     ; read two ASCII bytes echo to screen and convert them to one bye 8-b
3523                     ; exit: A
3524
3525    1395 CD4B12      get_hex1: call cin
3526    1398 CD3F12                call cout
3527    139B CD7613      call to_hex
3528    139E 0F                 rrc
3529    139F 0F                 rrc
3530    13A0 0F                 rrc
3531    13A1 0F                 rrc
3532    13A2 5F                 mov e,a
3533    13A3 CD4B12             call cin
3534    13A6 CD3F12             call cout
3535    13A9 CD7613             call to_hex
3536    13AC 83                 add e
3537    13AD C9          ret
3538
3539                     ; read two ASCII bytes echo to screen and convert them to one bye 8-b
3540                     ; exit: A
3541
3542    13AE 3A23F0      get_hex2: lda flag1
3543    13B1 E6FC                  ani 0fch      ; clear flag1.1 and flag1.0
3544    13B3 3223F0                sta flag1
3545
3546    13B6 CD4B12              call cin
3547    13B9 FE0D                cpi cr
3548    13BB CA0114              jz  exit_get_hex2
3549
3550    13BE FE20                cpi " "
3551    13C0 CA0A14              jz exit_get_hex3
3552
3553    13C3 FE30                cpi 30h        ; hex must be 0-9 and A-F
3554    13C5 DAAE13              jc  get_hex2
3555
3556    13C8 FE40                cpi 40h
3557    13CA DAD713              jc  ascii_0_9
3558
3559    13CD FE61                cpi 97         ; < 97?
3560    13CF DAAE13              jc  get_hex2
3561
3562    13D2 FE67                cpi 103        ; >= 103?
3563    13D4 D2AE13              jnc get_hex2
3564
3565    13D7             ascii_0_9:
3566
3567    13D7 CD3F12              call cout
3568    13DA CD7613      call to_hex
3569    13DD 0F                 rrc
3570    13DE 0F                 rrc
3571    13DF 0F                 rrc
3572    13E0 0F                 rrc
3573    13E1 5F                 mov e,a
3574
3575    13E2             get_2nd_hex:
```

```
3576
3577    13E2 CD4B12             call cin
3578
3579    13E5 FE30               cpi 30h         ; hex must be 0-9 and A-F
3580    13E7 DAE213             jc  get_2nd_hex
3581
3582    13EA FE40               cpi 40h
3583    13EC DAF913             jc  ok_0_9
3584
3585    13EF FE61               cpi 97          ; < 97?
3586    13F1 DAE213             jc  get_2nd_hex
3587
3588    13F4 FE67               cpi 103         ; >= 103?
3589    13F6 D2E213             jnc get_2nd_hex
3590
3591    13F9            ok_0_9:
3592    13F9 CD3F12             call cout
3593    13FC CD7613             call to_hex
3594    13FF 83                 add e
3595    1400 C9           ret
3596
3597    1401            exit_get_hex2:
3598
3599    1401 3A23F0             lda flag1
3600    1404 F601               ori 1
3601    1406 3223F0             sta flag1  ; Q key has been pressed
3602    1409 C9                 ret
3603
3604    140A            exit_get_hex3:
3605
3606    140A 3A23F0              lda flag1
3607    140D F602                ori 2
3608    140F 3223F0              sta flag1 ; SPACE key has been pressed
3609    1412 C9                  ret
3610
3611
3612                    ; add check sum
3613
3614
3615                    ; get record, write to SRAM and jump to 8000h
3616                    ; entry: A= byte received, B= byte check sum
3617
3618                    add_bcs:    macro    ; add accumulator with byte check sum stored in
3619                                push psw
3620                                add b
3621                                mov b,a
3622                                pop psw
3623                                endm
3624
3625    001B =          esc     equ 1bh
3626
3627    1413 CD4B12     get_record: call cin
3628    1416 FE1B                   cpi 27
3629    1418 CA6A14                 jz  esc_quit
3630
3631    141B FE3A                   cpi ":"
3632    141D C21314                 jnz get_record      ; wait until begin of record found
3633
3634    1420 0600                   mvi b,0             ; byte check sum
3635
3636    1422 CD8213         call get_hex ; get number of byte
3637    1425 4F                     mov c,a             ; put to c
3638
3639    1426                        add_bcs
3640    1426 F5                     push psw
```

```
3641    1427 80                 add b
3642    1428 47                 mov b,a
3643    1429 F1                 pop psw
3644    142A                    endm
3645
3646    142A CD8213         call get_hex ; get destination address, put to bx register
3647    142D 67                 mov  h,a            ; save high byte
3648
3649    142E                    add_bcs
3650    142E F5                 push psw
3651    142F 80                 add b
3652    1430 47                 mov b,a
3653    1431 F1                 pop psw
3654    1432                    endm
3655
3656    1432 CD8213             call get_hex
3657    1435 6F                 mov  l,a            ; and low byte
3658
3659    1436                    add_bcs
3660    1436 F5                 push psw
3661    1437 80                 add b
3662    1438 47                 mov b,a
3663    1439 F1                 pop psw
3664    143A                    endm
3665
3666    143A CD8213         call get_hex
3667
3668    143D                    add_bcs
3669    143D F5                 push psw
3670    143E 80                 add b
3671    143F 47                 mov b,a
3672    1440 F1                 pop psw
3673    1441                    endm
3674
3675    1441 FE01               cpi  1              ; end of record type is 01 ?
3676    1443 C26B14             jnz  data_record    ; jump if not 01
3677
3678    1446 CD4B12    wait_cr:    call cin
3679    1449 FE0D                  cpi  cr
3680    144B C24614                jnz wait_cr     ; until end of record sending! with cr de
3681
3682    144E 3EFF                  mvi a, 0ffh     ; turn speaker off
3683    1450 D312                  out system_port_c
3684    1452 AF                    xra a
3685    1453 D300                  out 0           ; turn off GPIO
3686
3687    1455 CDC914                call print_bcd1
3688    1458 CDF512                call space
3689    145B 3A20F0                lda bcs
3690    145E CDD301                call pint8u
3691    1461 21921E                lxi h,error_text
3692    1464 CD6812                call put_str
3693    1467 CDEA12                call new_line
3694
3695    146A C9        esc_quit:   ret
3696
3697
3698    146B CD8213    data_record: call get_hex       ; get data byte
3699    146E 77                      mov m,a            ; save to SRAM at [HL]
3700
3701    146F                         add_bcs
3702    146F F5                     push psw
3703    1470 80                     add b
3704    1471 47                     mov b,a
3705    1472 F1                     pop psw
```

```
3706    1473                        endm
3707
3708    1473 CD9914                  call inc_bcd1
3709
3710                            ;    ori 7fh
3711                            ;    out system_port_c   ; make buzzer sound
3712
3713
3714    1476 23                      inx h               ; next location
3715
3716    1477 0D                      dcr c
3717    1478 C26B14                  jnz data_record     ; until c = 0
3718
3719    147B 78                      mov a,b
3720    147C 2F                      cma
3721    147D 47                      mov b,a
3722    147E 04                      inr b               ; compute two's complement
3723
3724    147F CD8213                  call get_hex        ; get check sum
3725
3726    1482 B8                      cmp b
3727    1483 CA8D14                  jz  skip_error
3728
3729    1486 3A20F0                  lda bcs
3730    1489 3C                      inr a
3731    148A 3220F0                  sta bcs
3732
3733    148D            skip_error:
3734    148D 3A3AF0                  lda temp          ; then shift into temp8
3735    1490 07                      rlc
3736    1491 323AF0                  sta temp
3737    1494 D300                    out 0              ; send to GPIO
3738
3739    1496 C31314                 jmp get_record ; back to next record
3740
3741
3742                    ;---------------- increment BCD counter1 -------------------------
3743
3744    1499 E5         inc_bcd1:    push h
3745
3746    149A 2141F0                  lxi h,bcd_counter1
3747    149D AF                      xra a
3748
3749    149E 7E                      mov a,m
3750    149F C601                    adi 1
3751    14A1 27                      daa
3752    14A2 77                      mov m,a
3753    14A3 23                      inx h
3754
3755    14A4 7E                      mov a,m
3756    14A5 CE00                    aci 0
3757    14A7 27                      daa
3758    14A8 77                      mov m,a
3759    14A9 23                      inx h
3760    14AA 7E                      mov a,m
3761    14AB CE00                    aci 0
3762    14AD 27                      daa
3763    14AE 77                      mov m,a
3764
3765    14AF E1                      pop h
3766
3767    14B0 C9                      ret
3768
3769    14B1 E5         inc_bcd2:    push h
3770
```

```
3771    14B2 2144F0               lxi h,bcd_counter2
3772    14B5 AF                   xra a
3773
3774    14B6 7E                   mov a,m
3775    14B7 C601                 adi 1
3776    14B9 27                   daa
3777    14BA 77                   mov m,a
3778    14BB 23                   inx h
3779
3780    14BC 7E                   mov a,m
3781    14BD CE00                 aci 0
3782    14BF 27                   daa
3783    14C0 77                   mov m,a
3784    14C1 23                   inx h
3785    14C2 7E                   mov a,m
3786    14C3 CE00                 aci 0
3787    14C5 27                   daa
3788    14C6 77                   mov m,a
3789
3790    14C7 E1                   pop h
3791
3792    14C8 C9                   ret
3793
3794
3795    14C9           print_bcd1:
3796    14C9 2143F0               lxi h,bcd_counter1+2
3797    14CC 7E                   mov a,m
3798    14CD CDDB12               call out2x
3799    14D0 2142F0               lxi h,bcd_counter1+1
3800    14D3 7E                   mov a,m
3801    14D4 CDDB12               call out2x
3802    14D7 2141F0               lxi h,bcd_counter1
3803    14DA 7E                   mov a,m
3804    14DB CDDB12               call out2x
3805    14DE 21DF1E               lxi h,byte_text
3806    14E1 CD6812               call put_str
3807    14E4 C9                   ret
3808
3809    14E5 210000    clear_bcd1:  lxi h,0
3810    14E8 2241F0               shld bcd_counter1
3811    14EB 2242F0               shld bcd_counter1+1
3812    14EE 2243F0               shld bcd_counter1+2
3813    14F1 C9                   ret
3814
3815    14F2           print_bcd2:
3816    14F2 2146F0               lxi h,bcd_counter2+2
3817    14F5 7E                   mov a,m
3818    14F6 CDDB12               call out2x
3819    14F9 2145F0               lxi h,bcd_counter2+1
3820    14FC 7E                   mov a,m
3821    14FD CDDB12               call out2x
3822    1500 2144F0               lxi h,bcd_counter2
3823    1503 7E                   mov a,m
3824    1504 CDDB12               call out2x
3825    1507 21DF1E               lxi h,byte_text
3826    150A CD6812               call put_str
3827    150D C9                   ret
3828
3829    150E 210000    clear_bcd2:  lxi h,0
3830    1511 2244F0               shld bcd_counter2
3831    1514 2245F0               shld bcd_counter2+1
3832    1517 2246F0               shld bcd_counter2+2
3833    151A C9                   ret
3834
3835                            ; constants
```

```
3836
3837    151B            INS_TABLE:
3838
3839    151B 1B17               DWL C0 ;     "NOP",TAB,RS           ; 00
3840    151D 2017               DWL C1 ;     "LXI",TAB,"B,",RS      ; 01
3841    151F 2717               DWL C2 ;     "STAX",TAB,"B",RS      ; 02
3842    1521 2E17               DWL C3 ;     "INX",TAB,"B",RS       ; 03
3843    1523 3417               DWL C4 ;     "INR",TAB,"B",RS       ; 04
3844    1525 3A17               DWL C5 ;     "DCR",TAB,"B",RS       ; 05
3845    1527 4017               DWL C6 ;     "MVI",TAB,"B,",RS      ; 06
3846    1529 4717               DWL C7 ;     "RLC",TAB,RS           ; 07
3847    152B 4C17               DWL C8 ;     "DFB",TAB,RS           ; 08
3848    152D 5117               DWL C9 ;     "DAD",TAB,"B",RS       ; 09
3849    152F 5717               DWL CA ;     "LDAX",TAB,"B",RS      ; 0A
3850    1531 5E17               DWL CB ;     "DCX",TAB,"B",RS       ; 0B
3851    1533 6417               DWL CC ;     "INR",TAB,"C",RS       ; 0C
3852    1535 6A17               DWL CD ;     "DCR",TAB,"C",RS       ; 0D
3853    1537 7017               DWL CE ;     "MVI",TAB,"C,",RS      ; 0E
3854    1539 7717               DWL CF ;     "RRC",TAB,RS           ; 0F
3855    153B 7C17               DWL C10 ;    "DFB",TAB,RS           ; 10
3856    153D 8117               DWL C11 ;    "LXI",TAB,"D,",RS      ; 11
3857    153F 8817               DWL C12 ;    "STAX",TAB,"D",RS      ; 12
3858    1541 8F17               DWL C13 ;    "INX",TAB,"D",RS       ; 13
3859    1543 9517               DWL C14 ;    "INR",TAB,"D",RS       ; 14
3860    1545 9B17               DWL C15 ;    "DCR",TAB,"D",RS       ; 15
3861    1547 A117               DWL C16 ;    "MVI",TAB,"D,",RS      ; 16
3862    1549 A817               DWL C17 ;    "RAL",TAB,RS           ; 17
3863    154B AD17               DWL C18 ;    "DFB",TAB,RS            ; 18
3864    154D B217               DWL C19 ;    "DAD",TAB,"D",RS       ; 19
3865    154F B817               DWL C1A ;    "LDAX",TAB,"D",RS      ; 1A
3866    1551 BF17               DWL C1B ;    "DCX",TAB,"D",RS       ; 1B
3867    1553 C517               DWL C1C ;    "INR",TAB,"E",RS       ; 1C
3868    1555 CB17               DWL C1D ;    "DCR",TAB,"E",RS       ; 1D
3869    1557 D117               DWL C1E ;    "MVI",TAB,"E,",RS      ; 1E
3870    1559 D817               DWL C1F ;    "RAR",TAB,RS           ; 1F
3871    155B DD17               DWL C20 ;    "RIM",TAB,RS           ; 20
3872    155D E217               DWL C21 ;    "LXI",TAB,"H,",RS      ; 21
3873    155F E917               DWL C22 ;    "SHLD",TAB,RS          ; 22
3874    1561 EF17               DWL C23 ;    "INX",TAB,"H",RS       ; 23
3875    1563 F517               DWL C24 ;    "INR",TAB,"H",RS       ; 24
3876    1565 FB17               DWL C25 ;    "DCR",TAB,"H",RS       ; 25
3877    1567 0118               DWL C26 ;    "MVI",TAB,"H,",RS      ; 26
3878    1569 0818               DWL C27 ;    "DAA",TAB,RS           ; 27
3879    156B 0D18               DWL C28 ;    "DFB",TAB,RS            ; 28
3880    156D 1218               DWL C29 ;    "DAD",TAB,"H",RS       ; 29
3881    156F 1818               DWL C2A ;    "LHLD",TAB,RS          ; 2A
3882    1571 1E18               DWL C2B ;    "DCX",TAB,"H",RS       ; 2B
3883    1573 2418               DWL C2C ;    "INR",TAB,"L",RS       ; 2C
3884    1575 2A18               DWL C2D ;    "DCR",TAB,"L",RS       ; 2D
3885    1577 3018               DWL C2E ;    "MVI",TAB,"L,",RS      ; 2E
3886    1579 3718               DWL C2F ;    "CMA",TAB,RS           ; 2F
3887    157B 3C18               DWL C30 ;    "SIM",TAB,RS           ; 30
3888    157D 4118               DWL C31 ;    "LXI",TAB,"SP,",RS     ; 31
3889    157F 4918               DWL C32 ;    "STA",TAB,RS           ; 32
3890    1581 4E18               DWL C33 ;    "INX",TAB,"SP",RS      ; 33
3891    1583 5518               DWL C34 ;    "INR",TAB,"M",RS       ; 34
3892    1585 5B18               DWL C35 ;    "DCR",TAB,"M",RS       ; 35
3893    1587 6118               DWL C36 ;    "MVI",TAB,"M,",RS      ; 36
3894    1589 6818               DWL C37 ;    "STC",TAB,RS           ; 37
3895    158B 6D18               DWL C38 ;    "DFB",TAB,RS            ; 38
3896    158D 7218               DWL C39 ;    "DAD",TAB,"SP",RS      ; 39
3897    158F 7918               DWL C3A ;    "LDA",TAB,RS           ; 3A
3898    1591 7E18               DWL C3B ;    "DCX",TAB,"SP",RS      ; 3B
3899    1593 8518               DWL C3C ;    "INR",TAB,"A",RS       ; 3C
3900    1595 8B18               DWL C3D ;    "DCR",TAB,"A",RS       ; 3D
```

```
3901    1597 9118           DWL C3E ;    "MVI",TAB,"A,",RS       ; 3E
3902    1599 9818           DWL C3F ;    "CMC",TAB,RS            ; 3F
3903    159B 9D18           DWL C40 ;    "MOV",TAB,"B,B",RS      ; 40
3904    159D A518           DWL C41 ;    "MOV",TAB,"B,C",RS      ; 41
3905    159F AD18           DWL C42 ;    "MOV",TAB,"B,D",RS      ; 42
3906    15A1 B518           DWL C43 ;    "MOV",TAB,"B,E",RS      ; 43
3907    15A3 BD18           DWL C44 ;    "MOV",TAB,"B,H",RS      ; 44
3908    15A5 C518           DWL C45 ;    "MOV",TAB,"B,L",RS      ; 45
3909    15A7 CD18           DWL C46 ;    "MOV",TAB,"B,M",RS      ; 46
3910    15A9 D518           DWL C47 ;    "MOV",TAB,"B,A",RS      ; 47
3911    15AB DD18           DWL C48 ;    "MOV",TAB,"C,B",RS      ; 48
3912    15AD E518           DWL C49 ;    "MOV",TAB,"C,C",RS      ; 49
3913    15AF ED18           DWL C4A ;    "MOV",TAB,"C,D",RS      ; 4A
3914    15B1 F518           DWL C4B ;    "MOV",TAB,"C,E",RS      ; 4B
3915    15B3 FD18           DWL C4C ;    "MOV",TAB,"C,H",RS      ; 4C
3916    15B5 0519           DWL C4D ;    "MOV",TAB,"C,L",RS      ; 4D
3917    15B7 0D19           DWL C4E ;    "MOV",TAB,"C,M",RS      ; 4E
3918    15B9 1519           DWL C4F ;    "MOV",TAB,"C,A",RS      ; 4F
3919    15BB 1D19           DWL C50 ;    "MOV",TAB,"D,B",RS      ; 50
3920    15BD 2519           DWL C51 ;    "MOV",TAB,"D,C",RS      ; 51
3921    15BF 2D19           DWL C52 ;    "MOV",TAB,"D,D",RS      ; 52
3922    15C1 3519           DWL C53 ;    "MOV",TAB,"D,E",RS      ; 53
3923    15C3 3D19           DWL C54 ;    "MOV",TAB,"D,H",RS      ; 54
3924    15C5 4519           DWL C55 ;    "MOV",TAB,"D,L",RS      ; 55
3925    15C7 4D19           DWL C56 ;    "MOV",TAB,"D,M",RS      ; 56
3926    15C9 5519           DWL C57 ;    "MOV",TAB,"D,A",RS      ; 57
3927    15CB 5D19           DWL C58 ;    "MOV",TAB,"E,B",RS      ; 58
3928    15CD 6519           DWL C59 ;    "MOV",TAB,"E,C",RS      ; 59
3929    15CF 6D19           DWL C5A ;    "MOV",TAB,"E,D",RS      ; 5A
3930    15D1 7519           DWL C5B ;    "MOV",TAB,"E,E",RS      ; 5B
3931    15D3 7D19           DWL C5C ;    "MOV",TAB,"E,H",RS      ; 5C
3932    15D5 8519           DWL C5D ;    "MOV",TAB,"E,L",RS      ; 5D
3933    15D7 8D19           DWL C5E ;    "MOV",TAB,"E,M",RS      ; 5E
3934    15D9 9519           DWL C5F ;    "MOV",TAB,"E,A",RS      ; 5F
3935    15DB 9D19           DWL C60 ;    "MOV",TAB,"H,B",RS      ; 60
3936    15DD A519           DWL C61 ;    "MOV",TAB,"H,C",RS      ; 61
3937    15DF AD19           DWL C62 ;    "MOV",TAB,"H,D",RS      ; 62
3938    15E1 B519           DWL C63 ;    "MOV",TAB,"H,E",RS      ; 63
3939    15E3 BD19           DWL C64 ;    "MOV",TAB,"H,H",RS      ; 64
3940    15E5 C519           DWL C65 ;    "MOV",TAB,"H,L",RS      ; 65
3941    15E7 CD19           DWL C66 ;    "MOV",TAB,"H,M",RS      ; 66
3942    15E9 D519           DWL C67 ;    "MOV",TAB,"H,A",RS      ; 67
3943    15EB DD19           DWL C68 ;    "MOV",TAB,"L,B",RS      ; 68
3944    15ED E519           DWL C69 ;    "MOV",TAB,"L,C",RS      ; 69
3945    15EF ED19           DWL C6A ;    "MOV",TAB,"L,D",RS      ; 6A
3946    15F1 F519           DWL C6B ;    "MOV",TAB,"L,E",RS      ; 6B
3947    15F3 FD19           DWL C6C ;    "MOV",TAB,"L,H",RS      ; 6C
3948    15F5 051A           DWL C6D ;    "MOV",TAB,"L,L",RS      ; 6D
3949    15F7 0D1A           DWL C6E ;    "MOV",TAB,"L,M",RS      ; 6E
3950    15F9 151A           DWL C6F ;    "MOV",TAB,"L,A",RS      ; 6F
3951    15FB 1D1A           DWL C70 ;    "MOV",TAB,"M,B",RS      ; 70
3952    15FD 251A           DWL C71 ;    "MOV",TAB,"M,C",RS      ; 71
3953    15FF 2D1A           DWL C72 ;    "MOV",TAB,"M,D",RS      ; 72
3954    1601 351A           DWL C73 ;    "MOV",TAB,"M,E",RS      ; 73
3955    1603 3D1A           DWL C74 ;    "MOV",TAB,"M,H",RS      ; 74
3956    1605 451A           DWL C75 ;    "MOV",TAB,"M,L",RS      ; 75
3957    1607 4D1A           DWL C76 ;    "HLT",TAB,RS            ; 76
3958    1609 521A           DWL C77 ;    "MOV",TAB,"M,A",RS      ; 77
3959    160B 5A1A           DWL C78 ;    "MOV",TAB,"A,B",RS      ; 78
3960    160D 621A           DWL C79 ;    "MOV",TAB,"A,C",RS      ; 79
3961    160F 6A1A           DWL C7A ;    "MOV",TAB,"A,D",RS      ; 7A
3962    1611 721A           DWL C7B ;    "MOV",TAB,"A,E",RS      ; 7B
3963    1613 7A1A           DWL C7C ;    "MOV",TAB,"A,H",RS      ; 7C
3964    1615 821A           DWL C7D ;    "MOV",TAB,"A,L",RS      ; 7D
3965    1617 8A1A           DWL C7E ;    "MOV",TAB,"A,M",RS      ; 7E
```

```
3966    1619 921A            DWL C7F ;    "MOV",TAB,"A,A",RS      ; 7F
3967    161B 9A1A            DWL C80 ;    "ADD",TAB,"B",RS        ; 80
3968    161D A01A            DWL C81 ;    "ADD",TAB,"C",RS        ; 81
3969    161F A61A            DWL C82 ;    "ADD",TAB,"D",RS        ; 82
3970    1621 AC1A            DWL C83 ;    "ADD",TAB,"E",RS        ; 83
3971    1623 B21A            DWL C84 ;    "ADD",TAB,"H",RS        ; 84
3972    1625 B81A            DWL C85 ;    "ADD",TAB,"L",RS        ; 85
3973    1627 BE1A            DWL C86 ;    "ADD",TAB,"M",RS        ; 86
3974    1629 C41A            DWL C87 ;    "ADD",TAB,"A",RS        ; 87
3975    162B CA1A            DWL C88 ;    "ADC",TAB,"B",RS        ; 88
3976    162D D01A            DWL C89 ;    "ADC",TAB,"C",RS        ; 89
3977    162F D61A            DWL C8A ;    "ADC",TAB,"D",RS        ; 8A
3978    1631 DC1A            DWL C8B ;    "ADC",TAB,"E",RS        ; 8B
3979    1633 E21A            DWL C8C ;    "ADC",TAB,"H",RS        ; 8C
3980    1635 E81A            DWL C8D ;    "ADC",TAB,"L",RS        ; 8D
3981    1637 EE1A            DWL C8E ;    "ADC",TAB,"M",RS        ; 8E
3982    1639 F41A            DWL C8F ;    "ADC",TAB,"A",RS        ; 8F
3983    163B FA1A            DWL C90 ;    "SUB",TAB,"B",RS        ; 90
3984    163D 001B            DWL C91 ;    "SUB",TAB,"C",RS        ; 91
3985    163F 061B            DWL C92 ;    "SUB",TAB,"D",RS        ; 92
3986    1641 0C1B            DWL C93 ;    "SUB",TAB,"E",RS        ; 93
3987    1643 121B            DWL C94 ;    "SUB",TAB,"H",RS        ; 94
3988    1645 181B            DWL C95 ;    "SUB",TAB,"L",RS        ; 95
3989    1647 1E1B            DWL C96 ;    "SUB",TAB,"M",RS        ; 96
3990    1649 241B            DWL C97 ;    "SUB",TAB,"A",RS        ; 97
3991    164B 2A1B            DWL C98 ;    "SBB",TAB,"B",RS        ; 98
3992    164D 301B            DWL C99 ;    "SBB",TAB,"C",RS        ; 99
3993    164F 361B            DWL C9A ;    "SBB",TAB,"D",RS        ; 9A
3994    1651 3C1B            DWL C9B ;    "SBB",TAB,"E",RS        ; 9B
3995    1653 421B            DWL C9C ;    "SBB",TAB,"H",RS        ; 9C
3996    1655 481B            DWL C9D ;    "SBB",TAB,"L",RS        ; 9D
3997    1657 4E1B            DWL C9E ;    "SBB",TAB,"M",RS        ; 9E
3998    1659 541B            DWL C9F ;    "SBB",TAB,"A",RS        ; 9F
3999    165B 5A1B            DWL CA0 ;    "ANA",TAB,"B",RS        ; A0
4000    165D 601B            DWL CA1 ;    "ANA",TAB,"C",RS        ; A1
4001    165F 661B            DWL CA2 ;    "ANA",TAB,"D",RS        ; A2
4002    1661 6C1B            DWL CA3 ;    "ANA",TAB,"E",RS        ; A3
4003    1663 721B            DWL CA4 ;    "ANA",TAB,"H",RS        ; A4
4004    1665 781B            DWL CA5 ;    "ANA",TAB,"L",RS        ; A5
4005    1667 7E1B            DWL CA6 ;    "ANA",TAB,"M",RS        ; A6
4006    1669 841B            DWL CA7 ;    "ANA",TAB,"A",RS        ; A7
4007    166B 8A1B            DWL CA8 ;    "XRA",TAB,"B",RS        ; A8
4008    166D 901B            DWL CA9 ;    "XRA",TAB,"C",RS        ; A9
4009    166F 961B            DWL CAA ;    "XRA",TAB,"D",RS        ; AA
4010    1671 9C1B            DWL CAB ;    "XRA",TAB,"E",RS        ; AB
4011    1673 A21B            DWL CAC ;    "XRA",TAB,"H",RS        ; AC
4012    1675 A81B            DWL CAD ;    "XRA",TAB,"L",RS        ; AD
4013    1677 AE1B            DWL CAE ;    "XRA",TAB,"M",RS        ; AE
4014    1679 B41B            DWL CAF ;    "XRA",TAB,"A",RS        ; AF
4015    167B BA1B            DWL CB0 ;    "ORA",TAB,"B",RS        ; B0
4016    167D C01B            DWL CB1 ;    "ORA",TAB,"C",RS        ; B1
4017    167F C61B            DWL CB2 ;    "ORA",TAB,"D",RS        ; B2
4018    1681 CC1B            DWL CB3 ;    "ORA",TAB,"E",RS        ; B3
4019    1683 D21B            DWL CB4 ;    "ORA",TAB,"H",RS        ; B4
4020    1685 D81B            DWL CB5 ;    "ORA",TAB,"L",RS        ; B5
4021    1687 DE1B            DWL CB6 ;    "ORA",TAB,"M",RS        ; B6
4022    1689 E41B            DWL CB7 ;    "ORA",TAB,"A",RS        ; B7
4023    168B EA1B            DWL CB8 ;    "CMP",TAB,"B",RS        ; B8
4024    168D F01B            DWL CB9 ;    "CMP",TAB,"C",RS        ; B9
4025    168F F61B            DWL CBA ;    "CMP",TAB,"D",RS        ; BA
4026    1691 FC1B            DWL CBB ;    "CMP",TAB,"E",RS        ; BB
4027    1693 021C            DWL CBC ;    "CMP",TAB,"H",RS        ; BC
4028    1695 081C            DWL CBD ;    "CMP",TAB,"L",RS        ; BD
4029    1697 0E1C            DWL CBE ;    "CMP",TAB,"M",RS        ; BE
4030    1699 141C            DWL CBF ;    "CMP",TAB,"A",RS        ; BF
```

```
4031     169B 1A1C              DWL CC0 ;    "RNZ",TAB,RS             ; C0
4032     169D 1F1C              DWL CC1 ;    "POP",TAB,"B",RS         ; C1
4033     169F 251C              DWL CC2 ;    "JNZ",TAB,RS             ; C2
4034     16A1 2A1C              DWL CC3 ;    "JMP",TAB,RS             ; C3
4035     16A3 2F1C              DWL CC4 ;    "CNZ",TAB,RS             ; C4
4036     16A5 341C              DWL CC5 ;    "PUSH",TAB,"B",RS        ; C5
4037     16A7 3B1C              DWL CC6 ;    "ADI",TAB,RS             ; C6
4038     16A9 401C              DWL CC7 ;    "RST",TAB,"0",RS         ; C7
4039     16AB 461C              DWL CC8 ;    "RZ",TAB,RS              ; C8
4040     16AD 4A1C              DWL CC9 ;    "RET",TAB,RS             ; C9
4041     16AF 4F1C              DWL CCA ;    "JZ",TAB,RS              ; CA
4042     16B1 531C              DWL CCB ;    "DFB",TAB,RS              ; CB
4043     16B3 581C              DWL CCC ;    "CZ",TAB,RS              ; CC
4044     16B5 5C1C              DWL CCD ;    "CALL",TAB,RS            ; CD
4045     16B7 621C              DWL CCE ;    "ACI",TAB,RS             ; CE
4046     16B9 671C              DWL CCF ;    "RST",TAB,"1",RS         ; CF
4047     16BB 6D1C              DWL CD0 ;    "RNC",TAB,RS             ; D0
4048     16BD 721C              DWL CD1 ;    "POP",TAB,"D",RS         ; D1
4049     16BF 781C              DWL CD2 ;    "JNC",TAB,RS             ; D2
4050     16C1 7D1C              DWL CD3 ;    "OUT",TAB,RS             ; D3
4051     16C3 821C              DWL CD4 ;    "CNC",TAB,RS             ; D4
4052     16C5 871C              DWL CD5 ;    "PUSH",TAB,"D",RS        ; D5
4053     16C7 8E1C              DWL CD6 ;    "SUI",TAB,RS             ; D6
4054     16C9 931C              DWL CD7 ;    "RST",TAB,"2",RS         ; D7
4055     16CB 991C              DWL CD8 ;    "RC",TAB,RS              ; D8
4056     16CD 9D1C              DWL CD9 ;    "DFB",TAB,RS              ; D9
4057     16CF A21C              DWL CDA ;    "JC",TAB,RS              ; DA
4058     16D1 A61C              DWL CDB ;    "IN",TAB,RS              ; DFB
4059     16D3 AA1C              DWL CDC ;    "CC",TAB,RS              ; DC
4060     16D5 AE1C              DWL CDD ;    "DFB",TAB,RS              ; DD
4061     16D7 B31C              DWL CDE ;    "SBI",TAB,RS             ; DE
4062     16D9 B81C              DWL CDF ;    "RST",TAB,"3",RS         ; DF
4063     16DB BE1C              DWL CE0 ;    "RPO",TAB,RS             ; E0
4064     16DD C31C              DWL CE1 ;    "POP",TAB,"H",RS         ; E1
4065     16DF C91C              DWL CE2 ;    "JPO",TAB,RS             ; E2
4066     16E1 CE1C              DWL CE3 ;    "XTHL",TAB,RS            ; E3
4067     16E3 D41C              DWL CE4 ;    "CPO",TAB,RS             ; E4
4068     16E5 D91C              DWL CE5 ;    "PUSH",TAB,"H",RS        ; E5
4069     16E7 E01C              DWL CE6 ;    "ANI",TAB,RS             ; E6
4070     16E9 E51C              DWL CE7 ;    "RST",TAB,"4",RS         ; E7
4071     16EB EB1C              DWL CE8 ;    "RPE",TAB,RS             ; E8
4072     16ED F01C              DWL CE9 ;    "PCHL",TAB,RS            ; E9
4073     16EF F61C              DWL CEA ;    "JPE",TAB,RS             ; EA
4074     16F1 FB1C              DWL CEB ;    "XCHG",TAB,RS            ; EB
4075     16F3 011D              DWL CEC ;    "CPE",TAB,RS             ; EC
4076     16F5 061D              DWL CED ;    "DFB",TAB,RS              ; ED
4077     16F7 0B1D              DWL CEE ;    "XRI",TAB,RS             ; EE
4078     16F9 101D              DWL CEF ;    "RST",TAB,"5",RS         ; EF
4079     16FB 161D              DWL CF0 ;    "RP",TAB,RS              ; F0
4080     16FD 1A1D              DWL CF1 ;    "POP",TAB,"PSW",RS       ; F1
4081     16FF 221D              DWL CF2 ;    "JP",TAB,RS              ; F2
4082     1701 261D              DWL CF3 ;    "DI",TAB,RS              ; F3
4083     1703 2A1D              DWL CF4 ;    "CP",TAB,RS              ; F4
4084     1705 2E1D              DWL CF5 ;    "PUSH",TAB,"PSW",RS      ; F5
4085     1707 371D              DWL CF6 ;    "ORI",TAB,RS             ; F6
4086     1709 3C1D              DWL CF7 ;    "RST",TAB,"6",RS         ; F7
4087     170B 421D              DWL CF8 ;    "RM",TAB,RS              ; F8
4088     170D 461D              DWL CF9 ;    "SPHL",TAB,RS            ; F9
4089     170F 4C1D              DWL CFA ;    "JM",TAB,RS              ; FA
4090     1711 501D              DWL CFB ;    "EI",TAB,RS              ; FB
4091     1713 541D              DWL CFC ;    "CM",TAB,RS              ; FC
4092     1715 581D              DWL CFD ;    "DFB",TAB,RS              ; FD
4093     1717 5D1D              DWL CFE ;    "CPI",TAB,RS             ; FE
4094     1719 621D              DWL CFF ;    "RST",TAB,"7",RS,      ; FF
4095
```

```
                ; --------- mnemonic table ------------------------------------


    171B           MNEM
    171B 4E4F500900C0        DFB      "NOP",TAB,RS            ; 00
    1720 4C58490942C1        DFB      "LXI",TAB,"B,",RS       ; 01
    1727 5354415809C2        DFB      "STAX",TAB,"B",RS       ; 02
    172E 494E580942C3        DFB      "INX",TAB,"B",RS        ; 03
    1734 494E520942C4        DFB      "INR",TAB,"B",RS        ; 04
    173A 4443520942C5        DFB      "DCR",TAB,"B",RS        ; 05
    1740 4D56490942C6        DFB      "MVI",TAB,"B,",RS       ; 06
    1747 524C430900C7        DFB      "RLC",TAB,RS            ; 07
    174C 4446420900C8        DFB      "DFB",TAB,RS            ; 08
    1751 4441440942C9        DFB      "DAD",TAB,"B",RS        ; 09
    1757 4C44415809CA        DFB      "LDAX",TAB,"B",RS       ; 0A
    175E 4443580942CB        DFB      "DCX",TAB,"B",RS        ; 0B
    1764 494E520943CC        DFB      "INR",TAB,"C",RS        ; 0C
    176A 4443520943CD        DFB      "DCR",TAB,"C",RS        ; 0D
    1770 4D56490943CE        DFB      "MVI",TAB,"C,",RS       ; 0E
    1777 5252430900CF        DFB      "RRC",TAB,RS            ; 0F
    177C 4446420900C10        DFB      "DFB",TAB,RS            ; 10
    1781 4C58490944C11        DFB      "LXI",TAB,"D,",RS       ; 11
    1788 5354415809C12        DFB      "STAX",TAB,"D",RS       ; 12
    178F 494E580944C13        DFB      "INX",TAB,"D",RS        ; 13
    1795 494E520944C14        DFB      "INR",TAB,"D",RS        ; 14
    179B 4443520944C15        DFB      "DCR",TAB,"D",RS        ; 15
    17A1 4D56490944C16        DFB      "MVI",TAB,"D,",RS       ; 16
    17A8 52414C0900C17        DFB      "RAL",TAB,RS            ; 17
    17AD 4446420900C18        DFB      "DFB",TAB,RS            ; 18
    17B2 4441440944C19        DFB      "DAD",TAB,"D",RS        ; 19
    17B8 4C44415809C1A        DFB      "LDAX",TAB,"D",RS       ; 1A
    17BF 4443580944C1B        DFB      "DCX",TAB,"D",RS        ; 1B
    17C5 494E520945C1C        DFB      "INR",TAB,"E",RS        ; 1C
    17CB 4443520945C1D        DFB      "DCR",TAB,"E",RS        ; 1D
    17D1 4D56490945C1E        DFB      "MVI",TAB,"E,",RS       ; 1E
    17D8 5241520900C1F        DFB      "RAR",TAB,RS            ; 1F
    17DD 52494D0900C20        DFB      "RIM",TAB,RS            ; 20
    17E2 4C58490948C21        DFB      "LXI",TAB,"H,",RS       ; 21
    17E9 53484C4409C22        DFB      "SHLD",TAB,RS           ; 22
    17EF 494E580948C23        DFB      "INX",TAB,"H",RS        ; 23
    17F5 494E520948C24        DFB      "INR",TAB,"H",RS        ; 24
    17FB 4443520948C25        DFB      "DCR",TAB,"H",RS        ; 25
    1801 4D56490948C26        DFB      "MVI",TAB,"H,",RS       ; 26
    1808 4441410900C27        DFB      "DAA",TAB,RS            ; 27
    180D 4446420900C28        DFB      "DFB",TAB,RS            ; 28
    1812 4441440948C29        DFB      "DAD",TAB,"H",RS        ; 29
    1818 4C484C4409C2A        DFB      "LHLD",TAB,RS           ; 2A
    181E 4443580948C2B        DFB      "DCX",TAB,"H",RS        ; 2B
    1824 494E52094CC2C        DFB      "INR",TAB,"L",RS        ; 2C
    182A 444352094CC2D        DFB      "DCR",TAB,"L",RS        ; 2D
    1830 4D5649094CC2E        DFB      "MVI",TAB,"L,",RS       ; 2E
    1837 434D410900C2F        DFB      "CMA",TAB,RS            ; 2F
    183C 53494D0900C30        DFB      "SIM",TAB,RS            ; 30
    1841 4C58490953C31        DFB      "LXI",TAB,"SP,",RS      ; 31
    1849 5354410900C32        DFB      "STA",TAB,RS            ; 32
    184E 494E580953C33        DFB      "INX",TAB,"SP",RS       ; 33
    1855 494E52094DC34        DFB      "INR",TAB,"M",RS        ; 34
    185B 444352094DC35        DFB      "DCR",TAB,"M",RS        ; 35
    1861 4D5649094DC36        DFB      "MVI",TAB,"M,",RS       ; 36
    1868 5354430900C37        DFB      "STC",TAB,RS            ; 37
    186D 4446420900C38        DFB      "DFB",TAB,RS            ; 38
    1872 4441440953C39        DFB      "DAD",TAB,"SP",RS       ; 39
    1879 4C44410900C3A        DFB      "LDA",TAB,RS            ; 3A
    187E 4443580953C3B        DFB      "DCX",TAB,"SP",RS       ; 3B
    1885 494E520941C3C        DFB      "INR",TAB,"A",RS        ; 3C
```

```
4161    188B 4443520941C3D        DFB     "DCR",TAB,"A",RS        ; 3D
4162    1891 4D56490941C3E        DFB     "MVI",TAB,"A,",RS       ; 3E
4163    1898 434D430900C3F        DFB     "CMC",TAB,RS            ; 3F
4164    189D 4D4F560942C40        DFB     "MOV",TAB,"B,B",RS      ; 40
4165    18A5 4D4F560942C41        DFB     "MOV",TAB,"B,C",RS      ; 41
4166    18AD 4D4F560942C42        DFB     "MOV",TAB,"B,D",RS      ; 42
4167    18B5 4D4F560942C43        DFB     "MOV",TAB,"B,E",RS      ; 43
4168    18BD 4D4F560942C44        DFB     "MOV",TAB,"B,H",RS      ; 44
4169    18C5 4D4F560942C45        DFB     "MOV",TAB,"B,L",RS      ; 45
4170    18CD 4D4F560942C46        DFB     "MOV",TAB,"B,M",RS      ; 46
4171    18D5 4D4F560942C47        DFB     "MOV",TAB,"B,A",RS      ; 47
4172    18DD 4D4F560943C48        DFB     "MOV",TAB,"C,B",RS      ; 48
4173    18E5 4D4F560943C49        DFB     "MOV",TAB,"C,C",RS      ; 49
4174    18ED 4D4F560943C4A        DFB     "MOV",TAB,"C,D",RS      ; 4A
4175    18F5 4D4F560943C4B        DFB     "MOV",TAB,"C,E",RS      ; 4B
4176    18FD 4D4F560943C4C        DFB     "MOV",TAB,"C,H",RS      ; 4C
4177    1905 4D4F560943C4D        DFB     "MOV",TAB,"C,L",RS      ; 4D
4178    190D 4D4F560943C4E        DFB     "MOV",TAB,"C,M",RS      ; 4E
4179    1915 4D4F560943C4F        DFB     "MOV",TAB,"C,A",RS      ; 4F
4180    191D 4D4F560944C50        DFB     "MOV",TAB,"D,B",RS      ; 50
4181    1925 4D4F560944C51        DFB     "MOV",TAB,"D,C",RS      ; 51
4182    192D 4D4F560944C52        DFB     "MOV",TAB,"D,D",RS      ; 52
4183    1935 4D4F560944C53        DFB     "MOV",TAB,"D,E",RS      ; 53
4184    193D 4D4F560944C54        DFB     "MOV",TAB,"D,H",RS      ; 54
4185    1945 4D4F560944C55        DFB     "MOV",TAB,"D,L",RS      ; 55
4186    194D 4D4F560944C56        DFB     "MOV",TAB,"D,M",RS      ; 56
4187    1955 4D4F560944C57        DFB     "MOV",TAB,"D,A",RS      ; 57
4188    195D 4D4F560945C58        DFB     "MOV",TAB,"E,B",RS      ; 58
4189    1965 4D4F560945C59        DFB     "MOV",TAB,"E,C",RS      ; 59
4190    196D 4D4F560945C5A        DFB     "MOV",TAB,"E,D",RS      ; 5A
4191    1975 4D4F560945C5B        DFB     "MOV",TAB,"E,E",RS      ; 5B
4192    197D 4D4F560945C5C        DFB     "MOV",TAB,"E,H",RS      ; 5C
4193    1985 4D4F560945C5D        DFB     "MOV",TAB,"E,L",RS      ; 5D
4194    198D 4D4F560945C5E        DFB     "MOV",TAB,"E,M",RS      ; 5E
4195    1995 4D4F560945C5F        DFB     "MOV",TAB,"E,A",RS      ; 5F
4196    199D 4D4F560948C60        DFB     "MOV",TAB,"H,B",RS      ; 60
4197    19A5 4D4F560948C61        DFB     "MOV",TAB,"H,C",RS      ; 61
4198    19AD 4D4F560948C62        DFB     "MOV",TAB,"H,D",RS      ; 62
4199    19B5 4D4F560948C63        DFB     "MOV",TAB,"H,E",RS      ; 63
4200    19BD 4D4F560948C64        DFB     "MOV",TAB,"H,H",RS      ; 64
4201    19C5 4D4F560948C65        DFB     "MOV",TAB,"H,L",RS      ; 65
4202    19CD 4D4F560948C66        DFB     "MOV",TAB,"H,M",RS      ; 66
4203    19D5 4D4F560948C67        DFB     "MOV",TAB,"H,A",RS      ; 67
4204    19DD 4D4F56094CC68        DFB     "MOV",TAB,"L,B",RS      ; 68
4205    19E5 4D4F56094CC69        DFB     "MOV",TAB,"L,C",RS      ; 69
4206    19ED 4D4F56094CC6A        DFB     "MOV",TAB,"L,D",RS      ; 6A
4207    19F5 4D4F56094CC6B        DFB     "MOV",TAB,"L,E",RS      ; 6B
4208    19FD 4D4F56094CC6C        DFB     "MOV",TAB,"L,H",RS      ; 6C
4209    1A05 4D4F56094CC6D        DFB     "MOV",TAB,"L,L",RS      ; 6D
4210    1A0D 4D4F56094CC6E        DFB     "MOV",TAB,"L,M",RS      ; 6E
4211    1A15 4D4F56094CC6F        DFB     "MOV",TAB,"L,A",RS      ; 6F
4212    1A1D 4D4F56094DC70        DFB     "MOV",TAB,"M,B",RS      ; 70
4213    1A25 4D4F56094DC71        DFB     "MOV",TAB,"M,C",RS      ; 71
4214    1A2D 4D4F56094DC72        DFB     "MOV",TAB,"M,D",RS      ; 72
4215    1A35 4D4F56094DC73        DFB     "MOV",TAB,"M,E",RS      ; 73
4216    1A3D 4D4F56094DC74        DFB     "MOV",TAB,"M,H",RS      ; 74
4217    1A45 4D4F56094DC75        DFB     "MOV",TAB,"M,L",RS      ; 75
4218    1A4D 484C540900C76        DFB     "HLT",TAB,RS            ; 76
4219    1A52 4D4F56094DC77        DFB     "MOV",TAB,"M,A",RS      ; 77
4220    1A5A 4D4F560941C78        DFB     "MOV",TAB,"A,B",RS      ; 78
4221    1A62 4D4F560941C79        DFB     "MOV",TAB,"A,C",RS      ; 79
4222    1A6A 4D4F560941C7A        DFB     "MOV",TAB,"A,D",RS      ; 7A
4223    1A72 4D4F560941C7B        DFB     "MOV",TAB,"A,E",RS      ; 7B
4224    1A7A 4D4F560941C7C        DFB     "MOV",TAB,"A,H",RS      ; 7C
4225    1A82 4D4F560941C7D        DFB     "MOV",TAB,"A,L",RS      ; 7D
```

```
4226    1A8A 4D4F560941C7E        DFB      "MOV",TAB,"A,M",RS      ; 7E
4227    1A92 4D4F560941C7F        DFB      "MOV",TAB,"A,A",RS      ; 7F
4228    1A9A 4144440942C80        DFB      "ADD",TAB,"B",RS        ; 80
4229    1AA0 4144440943C81        DFB      "ADD",TAB,"C",RS        ; 81
4230    1AA6 4144440944C82        DFB      "ADD",TAB,"D",RS        ; 82
4231    1AAC 4144440945C83        DFB      "ADD",TAB,"E",RS        ; 83
4232    1AB2 4144440948C84        DFB      "ADD",TAB,"H",RS        ; 84
4233    1AB8 414444094CC85        DFB      "ADD",TAB,"L",RS        ; 85
4234    1ABE 414444094DC86        DFB      "ADD",TAB,"M",RS        ; 86
4235    1AC4 4144440941C87        DFB      "ADD",TAB,"A",RS        ; 87
4236    1ACA 4144430942C88        DFB      "ADC",TAB,"B",RS        ; 88
4237    1AD0 4144430943C89        DFB      "ADC",TAB,"C",RS        ; 89
4238    1AD6 4144430944C8A        DFB      "ADC",TAB,"D",RS        ; 8A
4239    1ADC 4144430945C8B        DFB      "ADC",TAB,"E",RS        ; 8B
4240    1AE2 4144430948C8C        DFB      "ADC",TAB,"H",RS        ; 8C
4241    1AE8 414443094CC8D        DFB      "ADC",TAB,"L",RS        ; 8D
4242    1AEE 414443094DC8E        DFB      "ADC",TAB,"M",RS        ; 8E
4243    1AF4 4144430941C8F        DFB      "ADC",TAB,"A",RS        ; 8F
4244    1AFA 5355420942C90        DFB      "SUB",TAB,"B",RS        ; 90
4245    1B00 5355420943C91        DFB      "SUB",TAB,"C",RS        ; 91
4246    1B06 5355420944C92        DFB      "SUB",TAB,"D",RS        ; 92
4247    1B0C 5355420945C93        DFB      "SUB",TAB,"E",RS        ; 93
4248    1B12 5355420948C94        DFB      "SUB",TAB,"H",RS        ; 94
4249    1B18 535542094CC95        DFB      "SUB",TAB,"L",RS        ; 95
4250    1B1E 535542094DC96        DFB      "SUB",TAB,"M",RS        ; 96
4251    1B24 5355420941C97        DFB      "SUB",TAB,"A",RS        ; 97
4252    1B2A 5342420942C98        DFB      "SBB",TAB,"B",RS        ; 98
4253    1B30 5342420943C99        DFB      "SBB",TAB,"C",RS        ; 99
4254    1B36 5342420944C9A        DFB      "SBB",TAB,"D",RS        ; 9A
4255    1B3C 5342420945C9B        DFB      "SBB",TAB,"E",RS        ; 9B
4256    1B42 5342420948C9C        DFB      "SBB",TAB,"H",RS        ; 9C
4257    1B48 534242094CC9D        DFB      "SBB",TAB,"L",RS        ; 9D
4258    1B4E 534242094DC9E        DFB      "SBB",TAB,"M",RS        ; 9E
4259    1B54 5342420941C9F        DFB      "SBB",TAB,"A",RS        ; 9F
4260    1B5A 414E410942CA0        DFB      "ANA",TAB,"B",RS        ; A0
4261    1B60 414E410943CA1        DFB      "ANA",TAB,"C",RS        ; A1
4262    1B66 414E410944CA2        DFB      "ANA",TAB,"D",RS        ; A2
4263    1B6C 414E410945CA3        DFB      "ANA",TAB,"E",RS        ; A3
4264    1B72 414E410948CA4        DFB      "ANA",TAB,"H",RS        ; A4
4265    1B78 414E41094CCA5        DFB      "ANA",TAB,"L",RS        ; A5
4266    1B7E 414E41094DCA6        DFB      "ANA",TAB,"M",RS        ; A6
4267    1B84 414E410941CA7        DFB      "ANA",TAB,"A",RS        ; A7
4268    1B8A 5852410942CA8        DFB      "XRA",TAB,"B",RS        ; A8
4269    1B90 5852410943CA9        DFB      "XRA",TAB,"C",RS        ; A9
4270    1B96 5852410944CAA        DFB      "XRA",TAB,"D",RS        ; AA
4271    1B9C 5852410945CAB        DFB      "XRA",TAB,"E",RS        ; AB
4272    1BA2 5852410948CAC        DFB      "XRA",TAB,"H",RS        ; AC
4273    1BA8 585241094CCAD        DFB      "XRA",TAB,"L",RS        ; AD
4274    1BAE 585241094DCAE        DFB      "XRA",TAB,"M",RS        ; AE
4275    1BB4 5852410941CAF        DFB      "XRA",TAB,"A",RS        ; AF
4276    1BBA 4F52410942CB0        DFB      "ORA",TAB,"B",RS        ; B0
4277    1BC0 4F52410943CB1        DFB      "ORA",TAB,"C",RS        ; B1
4278    1BC6 4F52410944CB2        DFB      "ORA",TAB,"D",RS        ; B2
4279    1BCC 4F52410945CB3        DFB      "ORA",TAB,"E",RS        ; B3
4280    1BD2 4F52410948CB4        DFB      "ORA",TAB,"H",RS        ; B4
4281    1BD8 4F5241094CCB5        DFB      "ORA",TAB,"L",RS        ; B5
4282    1BDE 4F5241094DCB6        DFB      "ORA",TAB,"M",RS        ; B6
4283    1BE4 4F52410941CB7        DFB      "ORA",TAB,"A",RS        ; B7
4284    1BEA 434D500942CB8        DFB      "CMP",TAB,"B",RS        ; B8
4285    1BF0 434D500943CB9        DFB      "CMP",TAB,"C",RS        ; B9
4286    1BF6 434D500944CBA        DFB      "CMP",TAB,"D",RS        ; BA
4287    1BFC 434D500945CBB        DFB      "CMP",TAB,"E",RS        ; BB
4288    1C02 434D500948CBC        DFB      "CMP",TAB,"H",RS        ; BC
4289    1C08 434D50094CCBD        DFB      "CMP",TAB,"L",RS        ; BD
4290    1C0E 434D50094DCBE        DFB      "CMP",TAB,"M",RS        ; BE
```

```
4291    1C14 434D500941CBF       DFB     "CMP",TAB,"A",RS        ; BF
4292    1C1A 524E5A0900CC0       DFB     "RNZ",TAB,RS            ; C0
4293    1C1F 504F500942CC1       DFB     "POP",TAB,"B",RS        ; C1
4294    1C25 4A4E5A0900CC2       DFB     "JNZ",TAB,RS            ; C2
4295    1C2A 4A4D500900CC3       DFB     "JMP",TAB,RS            ; C3
4296    1C2F 434E5A0900CC4       DFB     "CNZ",TAB,RS            ; C4
4297    1C34 5055534809CC5       DFB     "PUSH",TAB,"B",RS       ; C5
4298    1C3B 4144490900CC6       DFB     "ADI",TAB,RS            ; C6
4299    1C40 5253540930CC7       DFB     "RST",TAB,"0",RS        ; C7
4300    1C46 525A0900  CC8       DFB     "RZ",TAB,RS             ; C8
4301    1C4A 5245540900CC9       DFB     "RET",TAB,RS            ; C9
4302    1C4F 4A5A0900  CCA       DFB     "JZ",TAB,RS             ; CA
4303    1C53 4446420900CCB       DFB     "DFB",TAB,RS             ; CB
4304    1C58 435A0900  CCC       DFB     "CZ",TAB,RS             ; CC
4305    1C5C 43414C4C09CCD       DFB     "CALL",TAB,RS           ; CD
4306    1C62 4143490900CCE       DFB     "ACI",TAB,RS            ; CE
4307    1C67 5253540931CCF       DFB     "RST",TAB,"1",RS        ; CF
4308    1C6D 524E430900CD0       DFB     "RNC",TAB,RS            ; D0
4309    1C72 504F500944CD1       DFB     "POP",TAB,"D",RS        ; D1
4310    1C78 4A4E430900CD2       DFB     "JNC",TAB,RS            ; D2
4311    1C7D 4F55540900CD3       DFB     "OUT",TAB,RS            ; D3
4312    1C82 434E430900CD4       DFB     "CNC",TAB,RS            ; D4
4313    1C87 5055534809CD5       DFB     "PUSH",TAB,"D",RS       ; D5
4314    1C8E 5355490900CD6       DFB     "SUI",TAB,RS            ; D6
4315    1C93 5253540932CD7       DFB     "RST",TAB,"2",RS        ; D7
4316    1C99 52430900  CD8       DFB     "RC",TAB,RS             ; D8
4317    1C9D 4446420900CD9       DFB     "DFB",TAB,RS             ; D9
4318    1CA2 4A430900  CDA       DFB     "JC",TAB,RS             ; DA
4319    1CA6 494E0900  CDB       DFB     "IN",TAB,RS             ; DFB
4320    1CAA 43430900  CDC       DFB     "CC",TAB,RS             ; DC
4321    1CAE 4446420900CDD       DFB     "DFB",TAB,RS             ; DD
4322    1CB3 5342490900CDE       DFB     "SBI",TAB,RS            ; DE
4323    1CB8 5253540933CDF       DFB     "RST",TAB,"3",RS        ; DF
4324    1CBE 52504F0900CE0       DFB     "RPO",TAB,RS            ; E0
4325    1CC3 504F500948CE1       DFB     "POP",TAB,"H",RS        ; E1
4326    1CC9 4A504F0900CE2       DFB     "JPO",TAB,RS            ; E2
4327    1CCE 5854484C09CE3       DFB     "XTHL",TAB,RS           ; E3
4328    1CD4 43504F0900CE4       DFB     "CPO",TAB,RS            ; E4
4329    1CD9 5055534809CE5       DFB     "PUSH",TAB,"H",RS       ; E5
4330    1CE0 414E490900CE6       DFB     "ANI",TAB,RS            ; E6
4331    1CE5 5253540934CE7       DFB     "RST",TAB,"4",RS        ; E7
4332    1CEB 5250450900CE8       DFB     "RPE",TAB,RS            ; E8
4333    1CF0 5043484C09CE9       DFB     "PCHL",TAB,RS           ; E9
4334    1CF6 4A50450900CEA       DFB     "JPE",TAB,RS            ; EA
4335    1CFB 5843484709CEB       DFB     "XCHG",TAB,RS           ; EB
4336    1D01 4350450900CEC       DFB     "CPE",TAB,RS            ; EC
4337    1D06 4446420900CED       DFB     "DFB",TAB,RS             ; ED
4338    1D0B 5852490900CEE       DFB     "XRI",TAB,RS            ; EE
4339    1D10 5253540935CEF       DFB     "RST",TAB,"5",RS        ; EF
4340    1D16 52500900  CF0       DFB     "RP",TAB,RS             ; F0
4341    1D1A 504F500950CF1       DFB     "POP",TAB,"PSW",RS      ; F1
4342    1D22 4A500900  CF2       DFB     "JP",TAB,RS             ; F2
4343    1D26 44490900  CF3       DFB     "DI",TAB,RS             ; F3
4344    1D2A 43500900  CF4       DFB     "CP",TAB,RS             ; F4
4345    1D2E 5055534809CF5       DFB     "PUSH",TAB,"PSW",RS     ; F5
4346    1D37 4F52490900CF6       DFB     "ORI",TAB,RS            ; F6
4347    1D3C 5253540936CF7       DFB     "RST",TAB,"6",RS        ; F7
4348    1D42 524D0900  CF8       DFB     "RM",TAB,RS             ; F8
4349    1D46 5350484C09CF9       DFB     "SPHL",TAB,RS           ; F9
4350    1D4C 4A4D0900  CFA       DFB     "JM",TAB,RS             ; FA
4351    1D50 45490900  CFB       DFB     "EI",TAB,RS             ; FB
4352    1D54 434D0900  CFC       DFB     "CM",TAB,RS             ; FC
4353    1D58 4446420900CFD       DFB     "DFB",TAB,RS             ; FD
4354    1D5D 4350490900CFE       DFB     "CPI",TAB,RS            ; FE
4355    1D62 5253540937CFF       DFB     "RST",TAB,"7",RS,     ; FF
```

```
4356
4357
4358
4359    1D68 0D0A0A3830prompt1:    dfb cr,lf,lf,"8085 MICROPROCESSOR KIT (? HELP)",CR,LF,0
4360
4361    1D8E 3F065B4F66convert  dfb 3fh,06h,5bh,4fh,66h,6dh,7dh,07h,7fh,6fh,77h,7ch,39h,5eh,
4362
4363    1D9E 0000000000off_display: dfb 0,0,0,0,0,0
4364
4365                   ; lcd message     |--- 16x2-------|
4366    1DA4 3830383520prompt2:     dfb "8085 KIT 32k RAM",0
4367    1DB5 3936303055text3:       dfb "9600UART 8254T/C",0
4368
4369
4370    1DC6 6C6F616420download_text: dfb "load Intel hex file...",0
4371    1DDD 3E00      prompt_text:  dfb ">",0
4372    1DDF 6E6577206Cnew_text:     dfb "new location = ",0
4373    1DEF 6564697420edit_text:    dfb "edit memory location = ",0
4374    1E07 0D0A456E74edit_text1:   dfb cr,lf,"Enter to quit, SPACE key to view content",0
4375    1E32 0D0A0A4144edit_text2:   dfb cr,lf,lf,"ADDR  DATA",0
4376
4377    1E40 6A756D7020jump_text1:   dfb "jump to address [",0
4378    1E52 5D203D2000jump_text2:   dfb "] = ",0
4379    1E57 7072696E74ascii_text:   dfb "print ASCII code",0
4380
4381    1E68 426567696Efill_text1:   dfb "Begin address = ",0
4382    1E79 20456E6420fill_text2:   dfb " End address = ",0
4383    1E89 2044617461fill_text3:   dfb " Data = ",0
4384
4385    1E92 206572726Ferror_text:   dfb " errors",0
4386
4387    1E9A 6469736173disassemble_text: dfb "disassemble...",0
4388
4389    0000 =         eos           equ 0
4390
4391    1EA9 41463D00  af_text:  dfb "AF=",eos
4392    1EAD 42433D00  bc_text:  dfb "BC=",eos
4393    1EB1 44453D00  de_text:  dfb "DE=",eos
4394    1EB5 484C3D00  hl_text:  dfb "HL=",eos
4395    1EB9 53503D00  sp_text:  dfb "SP=",eos
4396    1EBD 544F533D00tos_text: dfb "TOS=",eos
4397
4398    1EC2 50433D00  pc_text:  dfb "PC=",eos
4399    1EC6 5B53205A20flag_text: dfb "[S Z - AC  - P - CY]=",eos
4400
4401    1EDC 533D00    sign_text: dfb "S=",0
4402
4403    1EDF 2062797465byte_text: dfb " bytes loaded",0
4404
4405    1EED 535441434Bstack_text: dfb "STACK Memory Contents..",0
4406
4407    1F05 7365742076set_register_text: dfb "set value to user register (enter A for AF) 1
4408
4409
4410    1F34 0D0A0A3830help_text:  dfb cr,lf,lf,"8085 MICROPROCESSOR KIT (? HELP)"
4411    1F57 0D0A41202Dhelp_text1: dfb cr,lf,   "A - ASCII code"
4412    1F67 0D0A43202D            dfb cr,lf,   "C - clear watch variables"
4413    1F82 0D0A44202D            dfb cr,lf,   "D - disassemble"
4414    1F93 0D0A45202D            dfb cr,lf,   "E - edit memory"
4415    1FA4 0D0A46202D            dfb cr,lf,   "F - fill constant"
4416    1FB7 0D0A48202D            dfb cr,lf,   "H - hex dump"
4417    1FC5 0D0A49202D            dfb cr,lf,   "I - i/o address map"
4418    1FDA 0D0A4A202D            dfb cr,lf,   "J - jump to user program"
4419    1FF4 0D0A4B202D            dfb cr,lf,   "K - display user STACK"
4420    200C 0D0A4C202D            dfb cr,lf,   "L - load Intel hex file"
```

```
4421     2025 0D0A4D202D            dfb cr,lf,   "M - monitor call number"
4422     203E 0D0A4E202D            dfb cr,lf,   "N - new location pointer"
4423     2058 0D0A51202D            dfb cr,lf,   "Q - quick home location"
4424     2071 0D0A52202D            dfb cr,lf,   "R - user register display"
4425     208C 0D0A53202D            dfb cr,lf,   "S - set value to user register"
4426     20AC 0D0A57202D            dfb cr,lf,   "W - watch variables"
4427     20C1 0D0A535041            dfb cr,lf,   "SPACE BAR - single step"
4428     20DA 0D0A3F202D            dfb cr,lf,   "? - help menu",cr,lf,0
4429
4430
4431     20EC 0D0A303048io_text     dfb cr,lf,"00H  GPIO LED"
4432                                ;dfb cr,lf,"        D4-D7=input port"
4433     20FB 0D0A                  dfb cr,lf
4434     20FD 0D0A313048            dfb cr,lf,"10H-13H 8255 system PPI, 10H=PORTA, 11H=PORTB,
4435     2144 0D0A                  dfb cr,lf
4436     2146 0D0A323048            dfb cr,lf,"20H-23H 8254 programmable counter, 20H=counter
4437     2186 0D0A202020            dfb cr,lf,"        22H=counter2, 23H control register"
4438     21B2 0D0A                  dfb cr,lf
4439                                ;dfb cr,lf,"30H-33H 8255 user PPI, 30H=PORTA, 31H=PORTB,
4440                                ;dfb cr,lf
4441     21B4 0D0A343048            dfb cr,lf,"40H-47H C16550 UART registers",0
4442
4443
4444     21D4           monitor_text:
4445
4446     21D4 7365652069            dfb  "see input parameters in user manual",cr,lf
4447     21F9 0D0A31456E            dfb  cr,lf,"1Enn MVI E,function_number"
4448     2215 0D0A434620            dfb  cr,lf,"CF   RST 1"
4449     2221 0D0A                  dfb  cr,lf
4450     2223 0D0A303020            dfb  cr,lf,"00 - demo"  ; #0 running LED with HL pointer
4451     222E 0D0A303120            dfb  cr,lf,"01 - delay" ; #1 simple delay routine
4452     223A 0D0A303220            dfb  cr,lf,"02 - cold_boot" ; #2 show 8085 running
4453     224A 0D0A303320            dfb  cr,lf,"03 - scan"    ; #3 scan display one cycle
4454     2255 0D0A303420            dfb  cr,lf,"04 - cin"     ; #4 get byte from console
4455     225F 0D0A303520            dfb  cr,lf,"05 - cout"    ; #5 print byte to console
4456     226A 0D0A303620            dfb  cr,lf,"06 - put_str"  ; #6 print string with 0 term
4457     2278 0D0A303720            dfb  cr,lf,"07 - init_lcd" ; #7 initialize lcd
4458     2287 0D0A303820            dfb  cr,lf,"08 - lcd_ready" ; #8 wait until lcd is ready
4459     2297 0D0A303920            dfb  cr,lf,"09 - clear_lcd" ; #9 clear lcd display
4460     22A7 0D0A304120            dfb  cr,lf,"0A - goto_xy"   ; #10 set lcd cursor positio
4461     22B5 0D0A304220            dfb  cr,lf,"0B - put_str_lcd" ; #11 print ASCII string o
4462     22C7 0D0A304320            dfb  cr,lf,"0C - put_ch_lcd"  ; #12 print ASCII letter o
4463     22D8 0D0A304420            dfb  cr,lf,"0D - demo2",0  ; #13 run GPIO LED
4464     22E5 0D0A304520            dfb  cr,lf,"0E - pint16u",0 ;#14 print 16-bit unsigned
4465
4466
4467
4468                          ;  data segment
4469
4470     F000                   org system_ram
4471
4472     F000           watch_ram dfs 16       ; watch variable F000-F00F
4473     F010           buffer   dfs 16        ; buffer display
4474     F020           bcs      dfs 1         ; byte checksum
4475     F021           key      dfs 1         ; key position
4476     F022           command  dfs 1         ; serial command
4477     F023           flag1    dfs 1         ; user flag
4478                                           ; flag1.0 Space key was pressed
4479                                           ; flag1.1 Enter key was pressed
4480     F024           beep_flag dfs 1        ; beep/no beep
4481
4482     F025           uart_found dfs 1       ; 0 = no uart, 1 uart found
4483
4484     F026           entry_mode dfs 1       ; 0 for data mode
4485                                           ; 1 for address mode
```

```
                                        ; 2 register display
F027            counter1   dfs 1        ; event counter1 for data entry
F028            counter2   dfs 1        ; event counter2 for address entry
F029            warm_code  dfs 1        ; warm boot code

F02A            user_PC  dfs 2          ; user PC
F02C            user_AF  dfs 2          ; user AF
F02E            user_BC  dfs 2          ; user BC
F030            user_DE  dfs 2          ; user DE
F032            user_HL  dfs 2          ; user HL
F034            user_SP  dfs 2          ; user SP
F036            tos      dfs 2          ; Top of STACK
F038            current_register dfs 2 ; stores current displayed register

F03A            temp     dfs 2          ; temporary 16-bit storage

F03C            pointer  dfs 2          ; for hex dump


F03E            break_address dfs 2     ; break address
F040            break_opcode  dfs 1     ; opcode that saved

F041            bcd_counter1  dfs 3     ; 6-digit BCD counter1
F044            bcd_counter2  dfs 3     ; 6-digit BCD counter2

F047            line_buffer   dfs 16    ; reserved for terminal printing


F057            save_stack    dfs 2     ; for saving system stack
F059            system_stack  dfs 32
F079            user_stack    dfs 32

0000                    END
```

```
4519
4519    1085  AC_TEXT           0A11  ADDRESS_MODE      1EA9  AF_TEXT
4520    1276  ALT_PUT_STR       13D7  ASCII_0_9         0D82  ASCII_PRINT
4521    0D9A  ASCII_PRINT1      1E57  ASCII_TEXT        F041  BCD_COUNTER1
4522    F044  BCD_COUNTER2      F020  BCS               1EAD  BC_TEXT
4523    097F  BEEP_CHK          F024  BEEP_FLAG         0650  BEEP_ON
4524    01A5  BIN1              01B0  BIN2              01A3  BIN2ASCII
4525    01BA  BIN3              01C5  BIN4              08DA  BREAK
4526    F03E  BREAK_ADDRESS     F040  BREAK_OPCODE      F010  BUFFER
4527    0080  BUSY              0641  BUZZER            0652  BUZZER1
4528    1EDF  BYTE_TEXT         171B  C0                1720  C1
4529    177C  C10               1781  C11               1788  C12
4530    178F  C13               1795  C14               179B  C15
4531    17A1  C16               17A8  C17               17AD  C18
4532    17B2  C19               17B8  C1A               17BF  C1B
4533    17C5  C1C               17CB  C1D               17D1  C1E
4534    17D8  C1F               1727  C2                17DD  C20
4535    17E2  C21               17E9  C22               17EF  C23
4536    17F5  C24               17FB  C25               1801  C26
4537    1808  C27               180D  C28               1812  C29
4538    1818  C2A               181E  C2B               1824  C2C
4539    182A  C2D               1830  C2E               1837  C2F
4540    172E  C3                183C  C30               1841  C31
4541    1849  C32               184E  C33               1855  C34
4542    185B  C35               1861  C36               1868  C37
4543    186D  C38               1872  C39               1879  C3A
4544    187E  C3B               1885  C3C               188B  C3D
4545    1891  C3E               1898  C3F               1734  C4
4546    189D  C40               18A5  C41               18AD  C42
4547    18B5  C43               18BD  C44               18C5  C45
4548    18CD  C46               18D5  C47               18DD  C48
4549    18E5  C49               18ED  C4A               18F5  C4B
4550    18FD  C4C               1905  C4D               190D  C4E
4551    1915  C4F               173A  C5                191D  C50
4552    1925  C51               192D  C52               1935  C53
4553    193D  C54               1945  C55               194D  C56
4554    1955  C57               195D  C58               1965  C59
4555    196D  C5A               1975  C5B               197D  C5C
4556    1985  C5D               198D  C5E               1995  C5F
4557    1740  C6                199D  C60               19A5  C61
4558    19AD  C62               19B5  C63               19BD  C64
4559    19C5  C65               19CD  C66               19D5  C67
4560    19DD  C68               19E5  C69               19ED  C6A
4561    19F5  C6B               19FD  C6C               1A05  C6D
4562    1A0D  C6E               1A15  C6F               1747  C7
4563    1A1D  C70               1A25  C71               1A2D  C72
4564    1A35  C73               1A3D  C74               1A45  C75
4565    1A4D  C76               1A52  C77               1A5A  C78
4566    1A62  C79               1A6A  C7A               1A72  C7B
4567    1A7A  C7C               1A82  C7D               1A8A  C7E
4568    1A92  C7F               174C  C8                1A9A  C80
4569    1AA0  C81               1AA6  C82               1AAC  C83
4570    1AB2  C84               1AB8  C85               1ABE  C86
4571    1AC4  C87               1ACA  C88               1AD0  C89
4572    1AD6  C8A               1ADC  C8B               1AE2  C8C
4573    1AE8  C8D               1AEE  C8E               1AF4  C8F
4574    1751  C9                1AFA  C90               1B00  C91
4575    1B06  C92               1B0C  C93               1B12  C94
4576    1B18  C95               1B1E  C96               1B24  C97
4577    1B2A  C98               1B30  C99               1B36  C9A
4578    1B3C  C9B               1B42  C9C               1B48  C9D
4579    1B4E  C9E               1B54  C9F               1757  CA
4580    1B5A  CA0               1B60  CA1               1B66  CA2
4581    1B6C  CA3               1B72  CA4               1B78  CA5
4582    1B7E  CA6               1B84  CA7               1B8A  CA8
```

```
4583   1B90  CA9              1B96  CAA              1B9C  CAB
4584   1BA2  CAC              1BA8  CAD              1BAE  CAE
4585   1BB4  CAF              175E  CB               1BBA  CB0
4586   1BC0  CB1              1BC6  CB2              1BCC  CB3
4587   1BD2  CB4              1BD8  CB5              1BDE  CB6
4588   1BE4  CB7              1BEA  CB8              1BF0  CB9
4589   1BF6  CBA              1BFC  CBB              1C02  CBC
4590   1C08  CBD              1C0E  CBE              1C14  CBF
4591   1764  CC               1C1A  CC0              1C1F  CC1
4592   1C25  CC2              1C2A  CC3              1C2F  CC4
4593   1C34  CC5              1C3B  CC6              1C40  CC7
4594   1C46  CC8              1C4A  CC9              1C4F  CCA
4595   1C53  CCB              1C58  CCC              1C5C  CCD
4596   1C62  CCE              1C67  CCF              176A  CD
4597   1C6D  CD0              1C72  CD1              1C78  CD2
4598   1C7D  CD3              1C82  CD4              1C87  CD5
4599   1C8E  CD6              1C93  CD7              1C99  CD8
4600   1C9D  CD9              1CA2  CDA              1CA6  CDB
4601   1CAA  CDC              1CAE  CDD              1CB3  CDE
4602   1CB8  CDF              1770  CE               1CBE  CE0
4603   1CC3  CE1              1CC9  CE2              1CCE  CE3
4604   1CD4  CE4              1CD9  CE5              1CE0  CE6
4605   1CE5  CE7              1CEB  CE8              1CF0  CE9
4606   1CF6  CEA              1CFB  CEB              1D01  CEC
4607   1D06  CED              1D0B  CEE              1D10  CEF
4608   1777  CF               1D16  CF0              1D1A  CF1
4609   1D22  CF2              1D26  CF3              1D2A  CF4
4610   1D2E  CF5              1D37  CF6              1D3C  CF7
4611   1D42  CF8              1D46  CF9              1D4C  CFA
4612   1D50  CFB              1D54  CFC              1D58  CFD
4613   1D5D  CFE              1D62  CFF              124B  CIN
4614   0EF3  CLEAR1           14E5  CLEAR_BCD1       150E  CLEAR_BCD2
4615   0914  CLEAR_BREAK      020A  CLEAR_LCD        0EE6  CLEAR_WATCH
4616   0BC3  CODE1            0C0B  CODE10           0C13  CODE11
4617   0C1B  CODE12           0C23  CODE13           0C2B  CODE14
4618   0C33  CODE15           0C3B  CODE16           0C43  CODE17
4619   0C4B  CODE18           0C53  CODE19           0BCB  CODE2
4620   0C5B  CODE20           0C63  CODE21           0C6B  CODE22
4621   0C73  CODE23           0C7B  CODE24           0C83  CODE25
4622   0C8B  CODE26           0BD3  CODE3            0BDB  CODE4
4623   0BE3  CODE5            0BEB  CODE6            0BF3  CODE7
4624   0BFB  CODE8            0C03  CODE9            0B8D  COLD1
4625   0B8B  COLD2            0B9C  COLD3            0B86  COLD_BOOT
4626   F022  COMMAND          0052  COMMAND_READ     0050  COMMAND_WRITE
4627   0023  CONTROL_8254     0034  CONTROL_WORD_8254 1D8E  CONVERT
4628   0020  COUNTER0_8254    F027  COUNTER1         0021  COUNTER1_8254
4629   F028  COUNTER2         0022  COUNTER2_8254    123F  COUT
4630   1240  COUT1            000D  CR               F038  CURRENT_REGISTER
4631   108C  CY_TEXT          0619  DATA_KEY         0624  DATA_KEY1
4632   062C  DATA_KEY2        0634  DATA_KEY3        063C  DATA_KEY4
4633   0A1E  DATA_MODE        0053  DATA_READ        146B  DATA_RECORD
4634   0051  DATA_WRITE       0CB7  DEBOUNCE        0CB9  DEBOUNCE1
4635   09A5  DEBUG            0A88  DECREMENT        0610  DELAY
4636   064B  DELAY_NOBEEP     0669  DELAY_US         066B  DELAY_US1
4637   095A  DELETE_BYTE      0969  DELETE_BYTE1     0BB0  DEMO
4638   0BB2  DEMO1_2          1EB1  DE_TEXT          0336  DIS2
4639   03A0  DISASSEM1        03AF  DISASSEM2        035E  DISASSEM3
4640   034A  DISASSEMBLE      0323  DISASSEMBLE1     1E9A  DISASSEMBLE_TEXT
4641   0670  DISPLAY_OFF      0D60  DOWNLOAD         1DC6  DOWNLOAD_TEXT
4642   1369  DUMP1            1361  DUMP_MEMORY      0416  D_DISASSEM1
4643   0425  D_DISASSEM2      03D4  D_DISASSEM3      03C0  D_DISASSEMBLE
4644   03EC  D_ONE_TAB        115C  EDIT1            1137  EDIT_LOCATION
4645   1DEF  EDIT_TEXT        1E07  EDIT_TEXT1       1E32  EDIT_TEXT2
4646   0719  ENTER_ADDRESS    06DF  ENTER_DATA       0704  ENTER_DATA1
4647   067F  ENTER_REGISTER   F026  ENTRY_MODE       0000  EOS
```

```
4648   1E92  ERROR_TEXT          001B  ESC                 146A  ESC_QUIT
4649   0DB3  EXIT_ASCII_PRINT    0211  EXIT_CLEAR          0F00  EXIT_CLEAR_WATCH
4650   0349  EXIT_DISASSEMBLE    0D81  EXIT_DOWNLOAD       11B9  EXIT_EDIT
4651   11AA  EXIT_EDIT1          10D3  EXIT_FILL           1401  EXIT_GET_HEX2
4652   140A  EXIT_GET_HEX3       1215  EXIT_HELP           1360  EXIT_HEX_DUMP
4653   11FD  EXIT_IO             1136  EXIT_JUMP           10EC  EXIT_MONITOR
4654   11D6  EXIT_NEW_LOCATION   0D5F  EXIT_PROMPTING      11EB  EXIT_QUICK_HOME
4655   1081  EXIT_REGISTER       0EC8  EXIT_SET_USER       0F5B  EXIT_STACK
4656   0F1B  EXIT_STEP           0F0F  EXIT_WATCH          1090  FILL_MEMORY
4657   10C2  FILL_MEMORY1        1E68  FILL_TEXT1          1E79  FILL_TEXT2
4658   1E89  FILL_TEXT3          F023  FLAG1               1EC6  FLAG_TEXT
4659   1239  FOUND               05A8  FUNCTION1           05B3  FUNCTION2
4660   05BE  FUNCTION3           05C9  FUNCTION4           05D5  FUNCTION5
4661   05E0  FUNCTION6           05EB  FUNCTION7           05F6  FUNCTION8
4662   0601  FUNCTION9           0A29  FUNCTION_2ND        059D  FUNCTION_KEY
4663   13E2  GET_2ND_HEX        1255  GET_COMMAND         1382  GET_HEX
4664   1395  GET_HEX1            13AE  GET_HEX2            0BBB  GET_KEY_CODE
4665   0436  GET_NUMBER_OF_BYTE  1413  GET_RECORD          09AB  GO
4666   0234  GOTO_XY             0243  GOTO_XY1            024E  GOTO_XY2
4667   0259  GOTO_XY3            0264  GOTO_XY4            0000  GPIO
4668   11FE  HELP               1F34  HELP_TEXT           1F57  HELP_TEXT1
4669   1301  HEX_DUMP            1322  HEX_DUMP1           130E  HEX_DUMP2
4670   1348  HEX_DUMP3           134A  HEX_DUMP4           133D  HEX_DUMP5
4671   1EB5  HL_TEXT             0991  HOME                8100  HOME_ADDRESS
4672   0A75  INCREMENT           1499  INC_BCD1            14B1  INC_BCD2
4673   0199  INIT_8254           0212  INIT_LCD            1216  INIT_UART
4674   092F  INSERT_BYTE         0942  INSERT_BYTE1        151B  INS_TABLE
4675   11EC  IO_ADDRESS          20EC  IO_TEXT             0715  IT_IS_RAM
4676   1E40  JUMP_TEXT1          1E52  JUMP_TEXT2          10ED  JUMP_TO_USER_PGM
4677   F021  KEY                0591  KEY_EXECUTE         0200  LCD_READY
4678   0201  LCD_READY1          000A  LF                  F047  LINE_BUFFER
4679   0190  MAIN                171B  MNEM                0B4D  MODE1
4680   0B84  MODE2               0B12  MODE_INDICATOR      0A4D  MODIFY_REGISTER
4681   026B  MONITOR_CALL        10D4  MONITOR_FUNCTION    21D4  MONITOR_TEXT
4682   8000  MY_ROM              12EA  NEW_LINE            11BA  NEW_LOCATION
4683   1DDF  NEW_TEXT           0CEB  NEXT_KEY            1262  NO_DATA
4684   043E  NUMBER1             0486  NUMBER10            048E  NUMBER11
4685   0496  NUMBER12            049E  NUMBER13            04A6  NUMBER14
4686   04AE  NUMBER15            04B6  NUMBER16            04BE  NUMBER17
4687   04C6  NUMBER18            04CE  NUMBER19            0446  NUMBER2
4688   04D6  NUMBER20            04DE  NUMBER21            04E6  NUMBER22
4689   04EE  NUMBER23            04F6  NUMBER24            04FE  NUMBER25
4690   0506  NUMBER26            050E  NUMBER27            0516  NUMBER28
4691   051E  NUMBER29            044E  NUMBER3             0526  NUMBER30
4692   052E  NUMBER31            0536  NUMBER32            053E  NUMBER33
4693   0546  NUMBER34            054E  NUMBER35            0556  NUMBER36
4694   055E  NUMBER37            0566  NUMBER38            056E  NUMBER39
4695   0456  NUMBER4             0576  NUMBER40            057E  NUMBER41
4696   0586  NUMBER42            058E  NUMBER43            045E  NUMBER5
4697   0466  NUMBER6             046E  NUMBER7             0476  NUMBER8
4698   047E  NUMBER9             1D9E  OFF_DISPLAY         0673  OFF_DISPLAY1
4699   13F9  OK_0_9              0376  ONE_TAB             098D  OPTION1
4700   12CA  OUT1X               12D6  OUT1X1              12DB  OUT2X
4701   1EC2  PC_TEXT             01EE  PINT1               01F9  PINT2
4702   01D3  PINT8U              F03C  POINTER             14C9  PRINT_BCD1
4703   14F2  PRINT_BCD2          0F01  PRINT_WATCH         0EC9  PRINT_WATCH_RAM
4704   1D68  PROMPT1             1DA4  PROMPT2             127F  PROMPT3
4705   0D42  PROMPTING           1DDD  PROMPT_TEXT         0265  PUT_CH_LCD
4706   07FD  PUT_FLAG            0805  PUT_HIGH1           0886  PUT_HIGH2
4707   1268  PUT_STR             126F  PUT_STR1            12B5  PUT_STR2
4708   0224  PUT_STR_LCD         022B  PUT_STR_LCD1        1089  P_TEXT
4709   11D7  QUICK_HOME          0ACE  READ_MEMORY         0AA3  READ_REGISTER
4710   0771  REGISTER1           08B4  REGISTER10          078D  REGISTER2
4711   07A9  REGISTER3           07C5  REGISTER4           07E1  REGISTER5
4712   0808  REGISTER6           0830  REGISTER7           085B  REGISTER8
```

```
4713   088C  REGISTER9           0F5C  REGISTER_DISPLAY   0F64  REGISTER_DISPLAY1
4714   0F67  REGISTER_DISPLAY2   0FF9  REGISTER_FLAG1     107A  REGISTER_FLAG10
4715   0FFE  REGISTER_FLAG2      1018  REGISTER_FLAG3     101D  REGISTER_FLAG4
4716   1037  REGISTER_FLAG5      103C  REGISTER_FLAG6     1056  REGISTER_FLAG7
4717   105B  REGISTER_FLAG8      1075  REGISTER_FLAG9     0000  ROM
4718   0000  RS                  F057  SAVE_STACK         0CBE  SCAN
4719   0CCC  SCAN1               0C8E  SCAN_KEY           0CA5  SCAN_KEY1
4720   0C9F  SCAN_KEY2           0C9F  SCAN_KEY3          0C8E  SCAN_KEY4
4721   0753  SELECT_REGISTER     0D4A  SEND_PROMPT        12BC  SEND_PROMPT1
4722   12C3  SEND_PROMPT3        12FB  SEND_TAB           0CFD  SERIAL_COMMAND
4723   02C5  SERVICE_RST2        029D  SERVICE_RST7       02ED  SERVICE_TRAP
4724   1F05  SET_REGISTER_TEXT   0DF0  SET_USER1          0E1B  SET_USER2
4725   0E46  SET_USER3           0E71  SET_USER4          0E9C  SET_USER5
4726   0EC7  SET_USER6           0DB4  SET_USER_REGISTER  072F  SHIFT_ADDRESS
4727   06F2  SHIFT_DATA          0CE1  SHIFT_KEY          068A  SHIFT_REGISTER
4728   1EDC  SIGN_TEXT           09CF  SINGLE_STEP        0F10  SINGLE_STEP_
4729   031E  SKIP1               09EF  SKIP2              014B  SKIP_BOOT
4730   016E  SKIP_COLD_BOOT      119A  SKIP_EDIT1         11A3  SKIP_EDIT2
4731   148D  SKIP_ERROR         0131  SKIP_LCD           1127  SKIP_LOAD_PC
4732   0807  SKIP_PUT_HIGH1     0888  SKIP_PUT_HIGH2     0185  SKIP_SEND_PROMPT
4733   0D41  SKIP_SERIAL        0020  SP                 12F5  SPACE
4734   1EB9  SP_TEXT            0F1C  STACK_DISPLAY      0F30  STACK_DISPLAY1
4735   1EED  STACK_TEXT         0100  START              0010  SYSTEM_PORT_A
4736   0011  SYSTEM_PORT_B      0012  SYSTEM_PORT_C      0013  SYSTEM_PORT_CONTROL
4737   F000  SYSTEM_RAM         F059  SYSTEM_STACK       0009  TAB
4738   F03A  TEMP              063D  TEST_BUZZER        0602  TEST_LED
4739   0604  TEST_LED1          1DB5  TEXT3              0BA4  TITLE
4740   F036  TOS               1EBD  TOS_TEXT           1376  TO_HEX
4741   0A98  TO_SEVEN_SEGMENT   0040  UART_BUFFER        0040  UART_DIVISOR_LSB
4742   0041  UART_DIVISOR_MSB   0042  UART_FIFO          F025  UART_FOUND
4743   0043  UART_LCR           0045  UART_LINE_STATUS   0047  UART_SCR
4744   F02C  USER_AF            F02E  USER_BC            F030  USER_DE
4745   F032  USER_HL            F02A  USER_PC            0030  USER_PORT_A
4746   0031  USER_PORT_B        0032  USER_PORT_C        0033  USER_PORT_CONTROL
4747   F034  USER_SP            F079  USER_STACK         027F  VECTOR_TABLE
4748   0CD6  WAIT1              1446  WAIT_CR            F029  WARM_CODE
4749   0ED9  WATCH1             F000  WATCH_RAM          1381  ZERO_NINE
4750   1082  ZERO_TEXT
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
4751
```