

```
;tabulka instrukci pocitace PETR 51 a jejich ciselnych kodu (OPCODE)
;
;pouzitelne operandy:  data (konstanta) XXX = 0-255
;                      adresa (skoky)   XXX = 0-127
;                      cislo registru   00X = 0-7
```

INSTRUKCE:

```
00 NOP          no operation (increment PC only)
01 HALT         stop running program and return to READY
02 DISPA XXX    display accumulator and wait XXX x 10ms or keypress (if XXX=000)
03 WAIT XXX     wait XXX x 10ms (0 - 2550 ms == 0 - 2,55 sec)
04 LAD XXX      load accumulator directly (load constant)
05 LAR 00X      load Acc from register X (X=0-7)
06 STAR 00X     store Acc to register X (X=0-7)
07 ADD 00X      additon: Acc = Acc + reg X (X=0-7)
08 SUBB 00X     subbstraction: Acc = Acc - reg X (X=0-7)
09 JUMP XXX     direct jump to XXX (XXX = 0-127)
10 CEA 00X      compare if Acc equal with reg X and set Flag
11 CBA 00X      compare if Acc bigger then reg X and set Flag
12 CLA 00X      compare if Acc lower then reg X and set Flag
13 CJMP XXX     conditional jump to XXX (jump if Flag is set)
14 NOT          Acc = NOT Acc
15 OR 00X       Acc = Acc OR reg X
16 AND 00X      Acc = Acc AND reg X
17 XOR 00X      Acc = Acc XOR reg X
18 LAI 00X      Acc = content of address in reg X (address in RAM)
19 STAI 00X     store Acc to address in reg X (address in RAM)
20 JMR 00X      jump to address in reg X (address in RAM)
21 IN           input Port to Acc
22 OUT          Acc to ouput Port
23 INCA         increment Acc
24 DECA         decrement Acc
25 INCR 00X     increment reg X
26 DECR 00X     decrement reg X
27 INDA XXX     input data from keyboard to Acc, XXX = preset data
28 RANDA        random data to Acc
29 RLA         rotate Acc left
30 RRA         rotate Acc right
31 MUL 00X      multiply: Acc = Acc x Reg X
32 DIV 00X      divide: Acc = Acc DIV Reg X, Reg X = Acc MOD Reg X
33 DISPCHAR XXX display 3 characters stored from address XXX
34 STRPC 00X    store PC+2 to Reg X (return addr for subroutines, JMR 00X = ret.)
35 BRK         breakpoint
```

```
;Celkem 36 instrukci
;Vsechny instrukce jsou 2-bytove XX, XXX (OPCODE, OPERAND)
;U nekterych instrukci (NOP, HALT, NOT, IN,...) na hodnote operandu nezalezi.
```