Mikroprocesorové praktikum **FJFI ČVUT v Praze**

Úloha 04 – A/D převodník

Jméno: **Kateřina Hrnečková** Měřeno: **4.11.2022** Klasifikace:

1 Pracovní úkoly

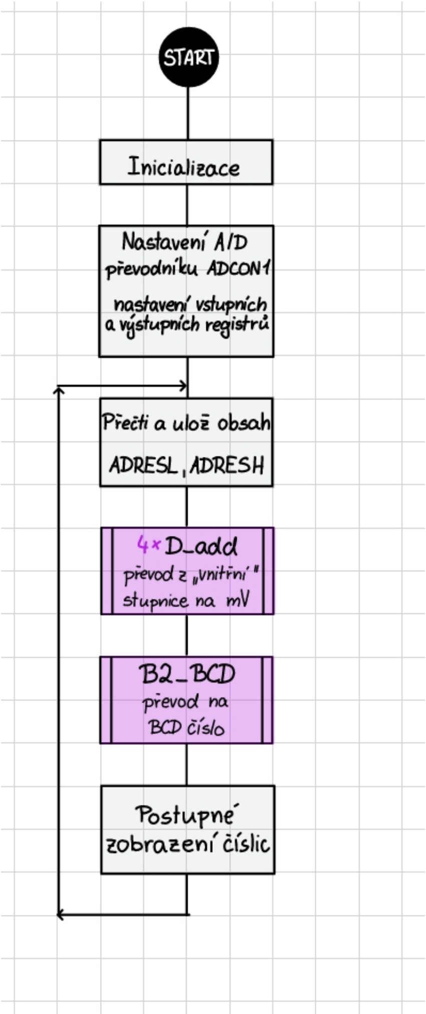
1. Seznamte se s modulem A/D převodníku.
2. Použijte modul A/D převodníku k měření analogových veličin.

2 Vypracování

Řešená úloha: Voltmetr

```
        list    p=PIC16F877A
        __config 0x2F09
        #include "p16f877a.inc"
;---------------------------------
count        equ        0x20
temp        equ        0x21

H_byte        equ        0x22
L_byte        equ        0x23
R0        equ        0x24
R1        equ        0x25
R2        equ        0x26
cnt1        equ        0x27
cnt2        equ        0x28
l        equ        0x29
h        equ        0x30

        org        0
        goto        Start

        org        0x04

Start:
        banksel        adcon1
        bsf        trisa, 2
        movlw        B'11000000'
        movwf        adcon1
        banksel        adcon0
        movlw        B'10010001'
        movwf        adcon0

        banksel trisd
        clrf        trisd

        movlw        B'11101000'
        movwf        trisb

        banksel portb
        movlw        0xff
        movwf        portb

Main:

        bsf        adcon0, go
        btfsc        adcon0, go
        goto $-1

        banksel        adresl
        movf        adresl, w
        banksel        L_byte
        movwf        L_byte


        banksel adresh
        movf        adresh, w
        banksel H_byte
        movwf        H_byte

        movf        H_byte, w
        banksel h
        movwf        h

        banksel        L_byte
        movf        L_byte, w
        banksel l
        movwf        l

;---- VYSOCE EFEKTIVNI NASOBENÍ 5 ----
        call        D_add
        call        D_add
        call        D_add
        call        D_add


        call B2_BCD
        ; L_byte -> R2
        ; H_byte -> R1

;---------- PRVNI CISLO ---------------
        movlw        0x0f
        andwf        R2, w
        call        Table

        movwf        portd
        bcf        portb,0

        call        Wait

        bsf        portb,0
```

```
;---------- DRUHE CISLO ---------------
        movlw        0x0f
        swapf        R2, f
        andwf        R2, w
        call        Table

        movwf        portd
        bcf        portb,1

        call        Wait

        bsf        portb,1

;---------- TRETI CISLO ---------------
        movlw        0x0f
        andwf        R1, w
        call        Table

        movwf        portd
        bcf        portb,2

        call        Wait

        bsf        portb,2

;---------- CTVRTE CISLO ---------------
        movlw        0x0f
        swapf        R1, f
        andwf        R1, w
        call        Table

        movwf        portd
        bcf        portb,4

        call        Wait

        bsf        portb,4
;----------------------------------------

        goto Main


; --------------------------------------------------------

Table:  addwf   pcl,f           ;Display segments table
        retlw   B’11000000’     ;0
        retlw   B’11111001’     ;1
        retlw   B’10100100’     ;2
        retlw   B’10110000’     ;3
        retlw   B’10011001’     ;4
        retlw   B’10010010’     ;5
        retlw   B’10000010’     ;6
        retlw   B’11111000’     ;7
        retlw   B’10000000’     ;8
        retlw   B’10010000’     ;9
        retlw   B’11111111’     ;display off
;
Wait:
        movlw        0x01
        movwf        cnt2
Wait_A:
        movlw   0xFF            ;this subroutine wait 770 cycles
        movwf   cnt1
Wait_B:
        decfsz  cnt1,f          ;decrement cnt1
        goto    Wait_B

        decfsz  cnt2,f          ;decrement cnt2
        goto    Wait_A
        return                  ;if cnt1=0 then return

;---------------------------------------------------------------
B2_BCD:
        bcf        STATUS,0                ; clear the carry bit
        movlw        0x10
        movwf   count
        clrf    R0
        clrf    R1
        clrf    R2
loop16:
        rlf     L_byte, F
        rlf     H_byte, F
        rlf     R2, F
        rlf     R1, F
        rlf     R0, F
 ;
        decfsz  count, F
        goto    adjDEC
        RETLW   0
 ;
adjDEC:
        movlw   R2
        movwf   FSR
        call    adjBCD
 ;
        movlw   R1
        movwf   FSR
        call    adjBCD
 ;
        movlw   R0
        movwf   FSR
        call    adjBCD
 ;
        goto    loop16
 ;
adjBCD:
        movlw   3
        addwf   0,W
        movwf   temp
        btfsc   temp,3          ; test if result > 7
        movwf   0
        movlw   0x30
        addwf   0,W
        movwf   temp
        btfsc   temp,7          ; test if result > 7
        movwf   0               ; save as MSD
        RETLW   0

        ;------------------------------------------------------

D_add:
        banksel l
        movf        l,W
        banksel L_byte
        addwf        L_byte, F ; add lsb
        btfsc        STATUS,C ; add in carry
        incf        H_byte, F
        banksel h
        movf        h,W
        banksel        H_byte
        addwf        H_byte, F ; add msb
        retlw        0


        end
```


START
Inicializace
Nastavení A/D převodníku ADCON1
nastavení vstupních a výstupních registrů
Přečti a ulož obsah
ADRESL, ADRESH
4× D_add
převod z „vnitřní" stupnice na mV
B2_BCD
převod na BCD číslo
Postupné zobrazení číslic

Obr. 1: Diagram vypracování úlohy – Voltmetr.