

## **Izvještaj vježbe 4**

Praktikum automatike

## Zadatak:

Potrebno je napisati program u assembleru, koji realizira brojač digitalnog signala na pinu RD0. Brojač treba da broji 135 signala. Brojanje može da ide od 0 do 135 ili od 135 do 0. Smjer brojanja ovisi o signalu koji je doveden na RD1. Ako je ovaj signal „log0“ tada se vrši brojanje naprijed, ukoliko je upravljački signal „log1“ tada se vrši brojanje unazad. Na RD0 se dovodi TTL signal četvrtke sa frekvencijom koja se može mijenjati od 0,01Hz do 1KHz.. Na PORTB-u se nalazi osam LED. Potrebno je napraviti program koji broji impulse četvrtke koju dovodimo na RD0 i rezultat brojanja prikazuje na PORTB. Brojanje može ići naprijed ili nazad u ovisnosti od upravljačkog signala na RD1. Kada brojač izbroji 135 impulsa on se resetuje i starta od početka te se aktivira relej (RC7), ili da zvučni signal (RC3).

## Rješenje:

Prvo port B i port D postavljamo kao digitalni tako što na ANSELB/ANSELD dovodimo '00000000'. Nakon toga portove B i C postavljamo kao izlazne tako što na TRISB/TRISC dovodimo '00000000'. Port D postavimo kao ulazni tako što na TRISD dovedemo '11111111'. Zatim testiramo RD1 da li je 0, ukoliko je 0 preskače instrukciju goto nazad, odnosno broji naprijed. Ukoliko je 0 broji nazad.

Ukoliko brojimo nazad, u port C pomoću LAT registra upisujemo 0, a u brojac i port B upisujemo vrijednost 135. U labeli L1 dovodimo signal četvrtke na RD0, a u labeli L2 vršimo dekrementiranje brojača i porta B, s tim da port B testiramo da li je 0.

Ukoliko na portu B nije 0 vraćamo se u L1, a ukoliko jeste 0 na port C postavljamo '11111111'.

Ukoliko brojimo naprijed, potrebno je u brojac upisati 135 a u port B upisati 0, pa spuštanjem brojača prema 0 inkrementujemo port B i u trenutku kada je brojač 0, znamo da je na portu B vrijednost 135.

## Kod:

```
list p=16f1939
#include <p16f1939.inc>
__CONFIG
__CONFIG1, _FOSC_HS&_WDTE_OFF&_PWRTE_OFF&_MCLRE_ON&_CP_OFF&_CPD_OFF&_BOREN_OFF
&_CLKOUTEN_OFF&_IESO_OFF&_FCMEN_OFF
__CONFIG __CONFIG2, _WRT_OFF&_VCAPEN_OFF&_PLLEN_OFF&_STVREN_OFF&_LVP_OFF

;*****Deklaracija promjenjivih *****
brojac equ 0x87
brojaca equ 0x00
brojacb equ 0x00
brojacc equ 0x00
brojacd equ 0x00

;***** Struktura programske memorije *****
```

```
ORG 0x00 ; Reset vector
goto Glavni
ORG 0x04 ; Interapt vector
goto Glavni ; nema interapt rutine
```

```
Glavni
BANKSEL ANSELB
movlw B'00000000'
movwf ANSELB
```

```
BANKSEL ANSELD
movlw B'00000000'
movwf ANSELD
```

```
BANKSEL TRISD
movlw B'11111111'
movwf TRISD
```

```
movlw B'00000000'
movwf TRISB
```

```
movlw B'00000000'
movwf TRISC
```

```
L
BANKSEL PORTD
btfsc PORTD,1
goto nazad
goto naprijed
```

```
nazad
BANKSEL LATC
movlw B'00000000'
movwf LATC
```

```
movlw d'135'
movwf brojac
BANKSEL LATB
movlw d'135'
movwf LATB
```

```
L1
BANKSEL PORTD
btfsc PORTD,0
goto L1
btfss PORTD,0
goto L1
goto L2
```

```
L2
BANKSEL LATB
decf brojac
decfsz LATB
goto L1
```

```
BANKSEL LATC
movlw B'11111111'
movwf LATC
call sec
goto L
```

```
naprijed
BANKSEL LATC
movlw B'00000000'
movwf LATC
movlw d'135'
movwf brojac
BANKSEL LATB
movlw d'0'
movwf LATB
```

```
L3
BANKSEL PORTD
btfsc PORTD,0
goto L3
btfss PORTD,0
goto L3
goto L4
```

```
L4
BANKSEL LATB
incf LATB
decfsz brojac
goto L3
BANKSEL PORTC
movlw B'11111111'
movwf PORTC
call sec
goto L
sec
movlw 0x01 ; brojac A=5
movwf brojaca ;
L7
movlw 0x05 ; brojac B=14
movwf brojacb ;
L8
movlw 0x05 ; brojac C=46
movwf brojacc ;
L9
movlw 0x05 ; brojac D=193
movwf brojacd ;
L10
decfsz brojacd,1
goto L10
decfsz brojacc,1
goto L9
decfsz brojacb,1
goto L8decfsz brojaca,1
goto L7
return
end
```