**Лабораторна робота №1**

**Тема:** Вимірювання аналогових сигналів

**Мета:** отримати знання про способи вимірювання параметрів аналогових сигналів та первинну обробку результатів вимірювання; отримати навики побудови системи вимірювання аналогових сигналів.

**Теоретичні відомості**

Аналоговий сигнал - сигнал (напруга, струм тощо), неперервний на всьому проміжку часу. Протилежністю аналоговим сигналам є дискретний сигнал, який має обмежені часові рамки (дискрета, імпульс). Аналоговий сигнал є традиційним для використання у радіо-телекомунікаційних системах, системах автоматичного керування тощо. При передачі інформації аналоговим сигналом, його видозміна можлива шляхом зміни частоти чи амплітуди коливань.

Перевагою аналогового сигналу над дискретним є відсутність невизначеності між відліками, яку має дискретний сигнал. Аналоговий сигнал використовує певні властивості середовища для передачі інформації. В електричних сигналах для передачі інформації, може змінюватися напруга, струм, фаза або частота сигналу. Будь-яка інформація може передаватися в формі аналогового сигналу; часто такий сигнал це виміряна реакція на зміну фізичного явища, такого як звук, світло, температура, позиція, або тиск. Фізична величина перетворюється на аналоговий сигнал за допомогою перетворювача. Теоретично аналоговий сигнал має нескінченну роздільну здатність. На практиці аналоговий сигнал піддається електронному шуму і спотворенню, що спричинені каналами передачі і операцій з обробки сигналів, що можуть значно погіршити співвідношення сигнал-шум (SNR). Перетворення аналогового сигналу в цифрову форму породжує постійний шум низького рівня, який називають шумом квантування, але якщо сигнал уже перетворено в цифрову форму в основному він може передаватися або оброблятися без появи додаткового шуму або спотворення. В аналогових системах, важко встановити коли така деградація відбулася. Однак, в цифрових системах, деградацію можна не лише виявити але і виправити.

Основним завданням вимірювань є встановлення числового значення параметрів, що характеризують властивості фізичної величини. У статистичних вимірюваннях вхідний вплив розглядається як реалізація випадкового процесу, імовірнісна характеристика якого вимірюється. При вимірюванні фізичної величини необхідно вказати, до якого моменту часу, відноситься результат вимірювання. В основі формалізованого опису вимірювальної процедури лежить рівняння вимірювань, що встановлює зв'язок результату вимірювання із вхідним впливом і виконуваними перетвореннями. Наявність рівняння вимірювання дозволяє створити необхідне програмне забезпечення, а також провести метрологічний аналіз вимірювальної процедури і результатів вимірювання. При проведенні вимірювань використовуються три види вимірювальних перетворень – аналогові, аналого-цифрові і цифрові. Аналогове перетворення забезпечує необхідний діапазон зміни вимірюваної величини, аналого-цифрове – перехід до числового подання, цифрове – додаткову обробку сигналу.

Основою роботи цифрових вимірювальних приладів є перетворення безперервної (аналогової) величини в цифровий код. Методи аналого-цифрового перетворення у вимірюваннях зводяться до подання миттєвих значень вхідного впливу у фіксовані моменти часу відповідною кодовою комбінацією (числом). Фізичну основу аналого-цифрового перетворення становить стробування і зрівняння з фіксованими опорними рівнями. Найбільшого поширення одержали аналого-цифрові перетворювачі (АЦП) порозрядного кодування, послідовного рахунку, слідкуючого зрівноваження та інші.

**Завдання**

1. Реалізувати вимір 3-х аналогових сигналів в діапазоні від 0 до 5 V з періодом дискретизації 1 секунда.

2. Провести усереднення цих сигналів.

3. Усереднені сигнали необхідно порівняти з аварійною межею 4 V і при виході за аварійну межу виробляти попереджуючу індикацію за допомогою світлодіодів, підключених до виводуу РВ0, РВ1, РВ2.

4. Підготувати для кожного сигналу дані для індикації на 3-розрядних 7-сегментних індикаторах.

Опорна напруга 5,12 V. Тактова частота 8 МГц.

**Код програми**

.include "m16def.inc"

.def temp = R16

.equ CTC\_VALUE = 31250

.def CHANNEL = R17

.def tempH = R18

.def CH0L = R20

.def CH0H = R21

.def CH1L = R22

.def CH1H = R23

.def CH2L = R24

.def CH2H = R25

.def Ans = R15

.def AnsU = R14

.def AnsX = R13

.macro CH\_N ; @0-CHNL | @1-CHNH

add @0,temp

adc @1,tempH ; **Добавить с переносом**

lsr @1

ror @0 ; **Повернуть вправо через перенос**

rjmp CHANGE\_CH

.endm

.macro CMP\_CH\_N ; @0-CHNL | @1-CHNH | @2-TURN\_OFF\_0 | @3-TURN\_ON\_0 | @4-CMPR\_L0 | @5-CMPR\_H0 | @6-END\_CH0 | @7- Port Number (0/1/2)

cpi @1,HIGH(800) ;

brsh @5

rjmp @2

@5:

cpi @1,HIGH(800) ;

breq @4

@3:

sbi PORTB,@7 ; **включить светодиод**

rjmp @6

@4:

cpi @0,LOW(800) ;

brsh @3

@2:

cbi PORTB,@7 ; **выключить светодиод**

@6:

.endm

jmp RESET ; Reset Handler

jmp EXT\_INT0 ; IRQ0 Handler

jmp EXT\_INT1 ; IRQ1 Handler

jmp TIM2\_COMP ; Timer2 Compare Handler

jmp TIM2\_OVF ; Timer2 Overflow Handler

jmp TIM1\_CAPT ; Timer1 Capture Handler

jmp TIM1\_COMPA ; Timer1 CompareA Handler

jmp TIM1\_COMPB ; Timer1 CompareB Handler

jmp TIM1\_OVF ; Timer1 Overflow Handler

jmp TIM0\_OVF ; Timer0 Overflow Handler

jmp SPI\_STC ; SPI Transfer Complete Handler

jmp USART\_RXC ; USART RX Complete Handler

jmp USART\_UDRE ; UDR Empty Handler

jmp USART\_TXC ; USART TX Complete Handler

jmp ADC\_C ; ADC Conversion Complete Handler

jmp EE\_RDY ; EEPROM Ready Handler

jmp ANA\_COMP ; Analog Comparator Handler

jmp TWSI ; Two-wire Serial Interface Handler

jmp EXT\_INT2 ; IRQ2 Handler

jmp TIM0\_COMP ; Timer0 Compare Handler

jmp SPM\_RDY ; Store Program Memory Ready Handler

;

RESET:

ldi temp,high(RAMEND) ; **Старт основной программы**

out SPH,temp ; **Установите указатель стека в верхнюю часть ОЗУ**

ldi temp,low(RAMEND)

out SPL,temp

**;Определение подтягивания и установка высокого уровня выходов**

**;Определение направления для контактов порта**

ldi temp,(0<<PB0)|(0<<PB1)|(0<<PB2)

out PORTB,temp

ldi temp,(1<<DDB0)|(1<<DDB1)|(1<<DDB2)

out DDRB,temp

; Define ADC

ldi temp,(0<<MUX4)|(0<<MUX3)|(0<<MUX2)|(0<<MUX1)|(0<<MUX0)|(0<<ADLAR)|(0<<REFS1)|(0<<REFS0)|(1<<DDB2)

out ADMUX,temp

ldi temp,(1<<ADEN)|(1<<ADSC)|(0<<ADATE)|(1<<ADIE)|(1<<ADPS2)|(1<<ADPS1)|(0<<ADPS0)

out ADCSRA,temp

; Define Timer (sampling time 1s)

ldi temp,high(CTC\_VALUE)

out OCR1AH,temp

ldi temp,low(CTC\_VALUE)

out OCR1AL,temp

ldi temp,(0<<WGM11)|(0<<WGM10)

out TCCR1A,temp

ldi temp,(0<<WGM13)|(1<<WGM12)|(1<<CS12)|(0<<CS11)|(0<<CS10)

out TCCR1B,temp

; start from 0 channel

ldi CHANNEL,0

clr CH0L

clr CH0H

clr CH1L

clr CH1H

clr CH2L

clr CH2H

sei ; **Разрешить прерывания**

LOOP:

rjmp LOOP

ADC\_C: ; **Обработчик завершения преобразования АЦП**

in temp,ADCL; **от АЦП к ядру**

in tempH,ADCH

cpi CHANNEL,0

breq CH0

cpi CHANNEL,1

breq CH1

cpi CHANNEL,2

breq CH2

CHANGE\_CH:

inc CHANNEL

cpi CHANNEL,3

breq END\_ADC

out ADMUX,CHANNEL

ldi temp,(1<<ADEN)|(1<<ADSC)|(0<<ADATE)|(1<<ADIE)|(1<<ADPS2)|(1<<ADPS1)|(0<<ADPS0)

out ADCSRA,temp

END\_ADC:

reti

CH0:

CH\_N CH0L, CH0H

//add CH0L,temp

//adc CH0H,tempH ; **Добавить с переносом**

//lsr CH0H

//ror CH0L ; **Повернуть вправо через перенос**

//rjmp CHANGE\_CH

CH1:

CH\_N CH1L, CH1H

//add CH1L,temp

//adc CH1H,tempH ; **Добавить с переносом**

//lsr CH1H

//ror CH1L ; **Повернуть вправо через перенос**

//rjmp CHANGE\_CH

CH2:

CH\_N CH2L, CH2H

//add CH2L,temp

//adc CH2H,tempH ; **Добавить с переносом**

//lsr CH2H

//ror CH2L ; **Повернуть вправо через перенос**

//rjmp CHANGE\_CH

TIM1\_COMPA: ; **Обработчик сравнения таймера 1**

ldi CHANNEL, 0

out ADMUX,CHANNEL

ldi temp,(1<<ADEN)|(1<<ADSC)|(0<<ADATE)|(1<<ADIE)|(1<<ADPS2)|(1<<ADPS1)|(0<<ADPS0)

out ADCSRA,temp ;start ADC after 3 ch , start next measure cycle

CMP\_CH\_N CH0L, CH0H, TURN\_OFF\_0, TURN\_ON\_0, CMPR\_L0, CMPR\_H0, END\_CH0, 0

CMP\_CH\_N CH1L, CH1H, TURN\_OFF\_1, TURN\_ON\_1, CMPR\_L1, CMPR\_H1, END\_CH1, 1

CMP\_CH\_N CH2L, CH2H, TURN\_OFF\_2, TURN\_ON\_2, CMPR\_L2, CMPR\_H2, END\_CH2, 2

clr Ans; **очистить var**

clr AnsU

clr AnsX

mov temp,CH0L

mov tempH,CH0H

rcall TO\_BCD

sts (CH0\_IND+0),Ans

sts (CH0\_IND+1),AnsU

sts (CH0\_IND+2),AnsX

mov temp,CH1L

mov tempH,CH1H

rcall TO\_BCD

sts (CH1\_IND+0),Ans

sts (CH1\_IND+1),AnsU

sts (CH1\_IND+2),AnsX

mov temp,CH2L

mov tempH,CH2H

rcall TO\_BCD

sts (CH2\_IND+0),Ans

sts (CH2\_IND+1),AnsU

sts (CH2\_IND+2),AnsX

reti

EXT\_INT0: ; IRQ0 Handler

EXT\_INT1: ; IRQ1 Handler

TIM2\_COMP: ; Timer2 Compare Handler

TIM2\_OVF: ; Timer2 Overflow Handler

TIM1\_CAPT: ; Timer1 Capture Handler

TIM1\_COMPB: ; Timer1 CompareB Handler

TIM1\_OVF: ; Timer1 Overflow Handler

TIM0\_OVF: ; Timer0 Overflow Handler

SPI\_STC: ; SPI Transfer Complete Handler

USART\_RXC: ; USART RX Complete Handler

USART\_UDRE: ; UDR Empty Handler

USART\_TXC: ; USART TX Complete Handler

EE\_RDY: ; EEPROM Ready Handler

ANA\_COMP: ; Analog Comparator Handler

TWSI: ; Two-wire Serial Interface Handler

EXT\_INT2: ; IRQ2 Handler

TIM0\_COMP: ; Timer0 Compare Handler

SPM\_RDY:; Store Program Memory Ready Handler

reti

To\_BCD: // **двоичный код в десятичные цифры**

lvl200:

cpi tempH,0

brne Sub200

cpi temp,200

brlo lvl20

Sub200:

subi temp,200

sbci tempH,0

inc Ans

rjmp lvl200

lvl20:

cpi temp,20

brlo lvl2

subi temp,20

inc AnsU

rjmp lvl20

lvl2:

lsr temp; **shift temp ==> /2**

mov AnsX,temp

ret

.dseg

CH0\_IND:

.byte 3

CH1\_IND:

.byte 3

CH2\_IND:

.byte 3

**Висновок:** під час виконання лабораторної роботи ми отримати знання про способи вимірювання параметрів аналогових сигналів та первинну обробку результатів вимірювання, а також навики побудови системи вимірювання аналогових сигналів. Було реалізовано вимір 3-х аналогових сигналів та проведено усереднення цих сигналів. Потім порівняли сигнали з аварійною межею і при виході за неї виконано попереджуючу індикацію за допомогою світлодіодів.