

Лабораторна робота №4

ВИВІД ІНФОРМАЦІЇ НА LED-ІНДИКАТОРИ

Мета роботи: ознайомлення з методами виводу інформації на LED-індикатори; засвоєння методу статичної індикації; засвоєння методу динамічної індикації; розробка тестового проекту; програмування мікроконтролера лабораторного стенду.

4.1 Теоретичні відомості

Класифікація індикаторів

У залежності від типу фізичних процесів, що використовуються, розрізняють наступні види індикаторів - лампи накаливання; вакуумні, люмінісцентні, світлодіодні (LED), рідкокристалічні (LCD), плазмові панелі, активні та пасивні матриці.

У мікроконтролерних системах керування та обробки інформації найбільш розповсюдженими є LED та LCD індикатори.

Індикатори LED - типу

Такі індикатори використовують при побудові дисплеїв, що працюють у широкому діапазоні температур (-50 - +100°C).

Індикатори характеризуються високою надійністю, яскравістю (у окремих зразків фірми Luxeon Star до 200 Кд), малою вартістю. Фірми Kingbright та Paralight виробляють індикатори з сімома основними кольорами випромінювання.

У LED-індикаторів звичайно є лише елементи індикації, а інтерфейсні вузли відсутні.

Виробляються індикатори із такими конструкціями:

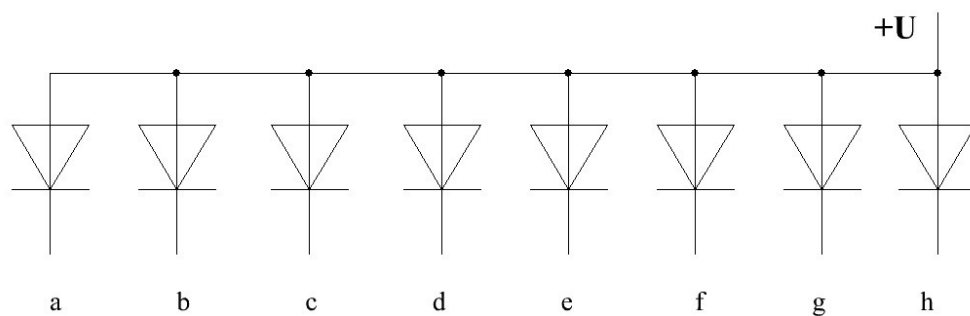
- одиночні індикатори циліндричної форми із діаметрами 1, 3, 5, 7, 10мм та кластери;
- мнемонічні індикатори (прямокутної, трикутної, символної форми);

- світлові табло прямокутної та круглої форми; графічні лінійки;
- одиночні семисегментні індикатори та їх набори (2-10 десяткових разрядів);
- мозаїчні індикатори.

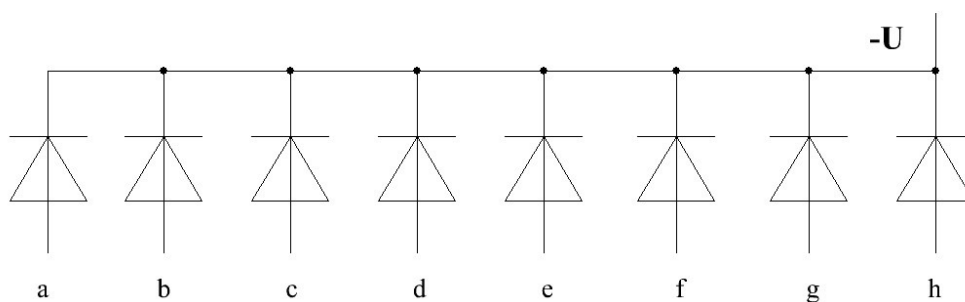
Випромінювання LED-індикатора викликається протіканням прямого струму. Величина падіння напруги на прямо зміщеному світлодіоді залежить від величини струму та використаної технології, що визначає колір випромінювання.

Одиночні індикатори звичайно мають один світлодіод. В індикаторах з великою площею поверхні (табло, семисегментні індикатори великого розміру) можуть використовуватися декілька послідовно ввімкнених світлодіодів.

У дисплеях для відтворення цифр використовують семисегментні індикатори. У залежності від електричної схеми розрізняють два типи таких індикаторів – із спільним анодом (рис.4.1,а) та із спільним катодом (рис. 4.1,б).



а)



б)

Рис.4.1

На вибір типу індикатора впливають особливості принципової схеми пристрою індикації та допустимі рівні вихідного струму (високого та низького рівня) буферних підсилювачів. Сегменти індикатору позначаються літерами “a – h”. Звичайно використовується варіант розміщення сегментів, що приведений на рис. 4.2.

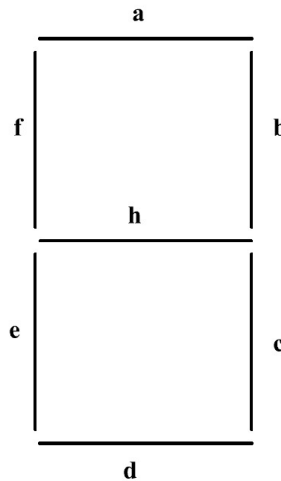


Рис.4.2

Для побудови багаторозрядних дисплеїв можуть застосовуватися одиночні семисегментні індикатори, або їх набори. У залежності від електричної схеми розрізняють набори призначені для статичної та динамічної індикації.

Набори семисегментних індикаторів статичної індикації мають для кожного розряду числа повний комплект ліній керування (рис. 4.1).

У наборах, що використовуються у схемах динамічної індикації (рис.4.3), лінії керування сегментами “a - h” усіх розрядів об’єднані. Розділяються лише лінії живлення розрядних груп (анодних або катодних).

Мозаїчні індикатори використовують у разі необхідності відображення графічних об’єктів, наприклад символів алфавіту. Такі індикатори побудовані на базі матриці індикаторів. Розрізняють два типи мозаїчних індикаторів. Перший тип використовується для побудови дисплеїв із великою кількістю пікселів.

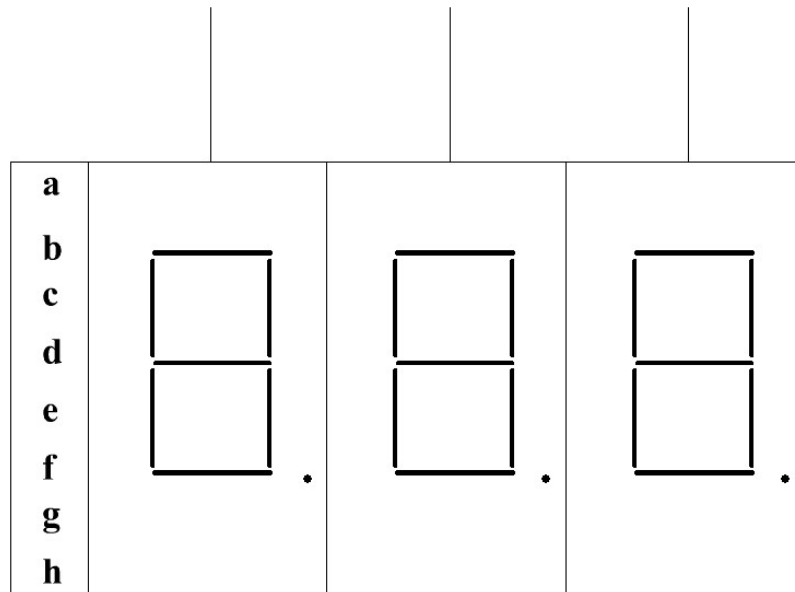


Рис.4.3

Розмір таких індикаторів 4x4, 8x8, 10x10 світлодіодів. Електричну схему мозаїчного індикатора розміром 2x2 приведено на рис.4.4.

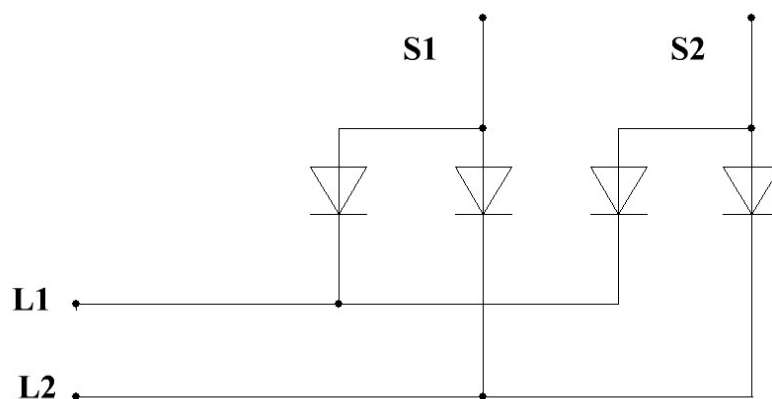


Рис.4.4

Мозаїчні індикатори можуть використовуватися лише у дисплеях із динамічною індикацією.

Метод статичної індикації

У дисплеях, що використовують метод статичної індикації, через світлодіоди індикатора протікає безперервний робочий струм. Для забезпечення такого метода кожен одиночний семисегментний індикатор необхідно підключати через власний буферний підсилювач. В якості таких підсилювачів досить часто використовують паралельні регістри. Для

передачі інформації на регістри використовується шина P8 мікроконтролера (є деяких AVR-процесорів), або ця шина формується програмний шляхом.

На рис.4.5 приведено принципову схему дворозрядного індикатора із програмним формуванням шини зв'язку з буферними регістрами. У схемі використано два буферних регістра D2, D3, підключених до загальної паралельної шини, яка формується з використанням порту PC.

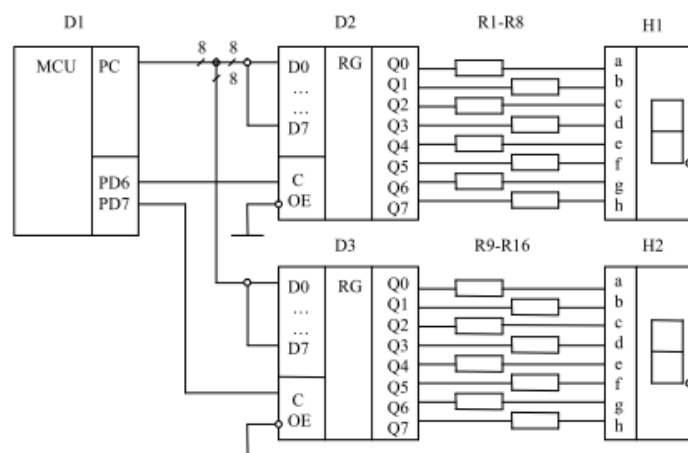


Рис.4.5

Для активізації виходів регістрів на лінії керування OE заведені активні логічні рівні -«0». Запис інформації у регістри одиниць та десятків відбувається за рахунок формування відповідних сигналів керування на лініях портів PD6, PD7. Резистори R1 – R16 використовуються для обмеження струму, що протікає через світлодіоди індикаторів H1, H2.

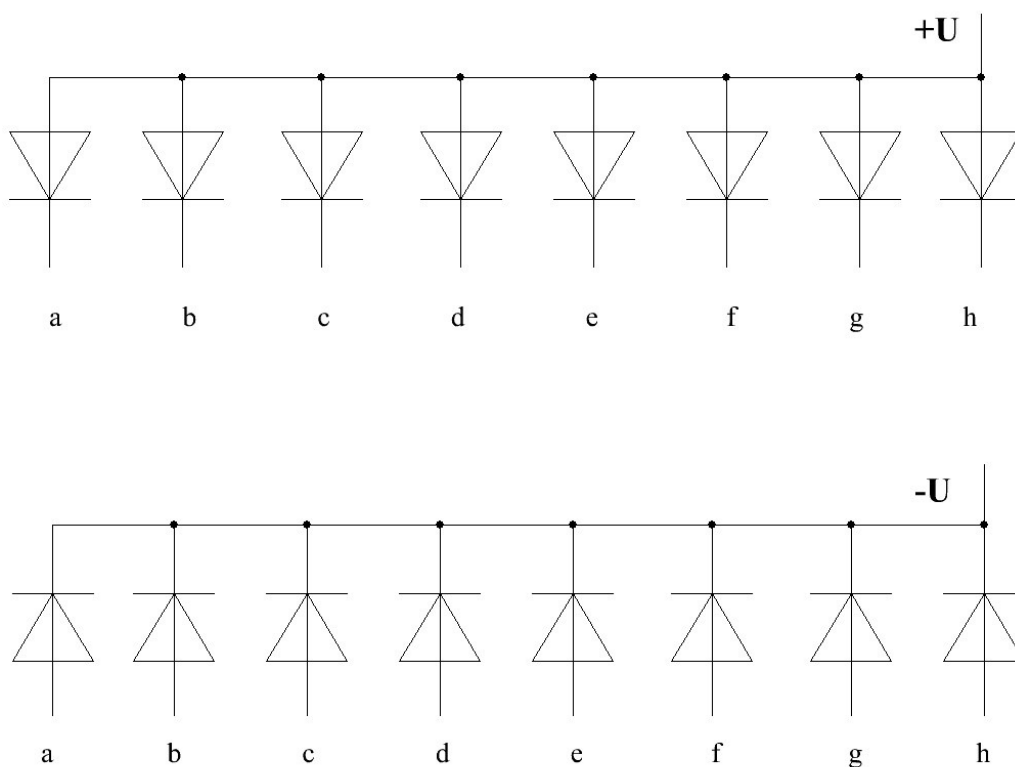
Використання метода статичної індикації спрощує програму-драйвер індикатора, але при цьому за рахунок великої кількості ліній керування та додаткових регістрів суттєво ускладнюється печатна плата.

Метод динамічної індикації

У дисплеях, що використовують динамічний метод індикації, через світлодіоди протікає імпульсний струм. Використовується спільний буферний регістр-підсилювач для керування сегментами “a - h” індикатора. Напруга живлення по черзі поступає на розряди індикатора.

Синхронно із напругою живлення розрядів визначається і стан сегментів цих розрядів. Таким чином інформація по черзі виводиться у розряди сотень, десятків та одиниць. Для вилучення оптичних ефектів частота комутації кожного розряду повинна перевищувати 25 Гц. Світлодіоди індикаторів доцільно використовувати на частотах комутації до 0,5-4 кГц. Для забезпечення нормальної яскравості випромінювання сегментів необхідно підвищувати імпульсний струм світлодіодів так, щоб середнє значення струму відповідало номінальному статичному струму. Так у трирозрядному індикаторі необхідно підвищувати імпульсний струм у три рази, у порівнянні з відповідним індикатором статичного типу.

Динамічний метод індикації реалізується схемотехнічно простіше, у



буферний регістр - підсилювач, який підключено до паралельної шини мікроконтролера (порт РС). Для активізації виходів регістра на лінію OE подано нулевий логічний рівень напруги. Активізація відповідних розрядів індикатора реалізується за допомогою ліній процесора PD6, PD7.

Для запису інформації з паралельної шини у регістр використовується сигнал керування, що формується на лінії PD3 мікроконтролера.

Резистори R1 – R8 використовуються для обмеження струму світлодіодів індикатора. Для нормалізації напруги, що прикладається до світлодіодів індикатора використано підсилювачі напруги D3.1, D3.2. Транзистори VT1, VT2 використовуються для комутації розрядів десятків та одиниць індикатора.

Підготовка даних для виводу на семисегментний індикатор

Для виводу цифр у семисегментний індикатор необхідно провести їх перекодування. Визначення кодів залежить від упаковки сегментів в індикаторі, упаковки паралельної шини даних та від того, який вихідний рівень порту паралельної шини є активним. У табл.4.1 приведено приклад визначення кодів «0» та «2» для схеми розташування сегментів, що наведено на рис. 4.2, визначеного у таблиці порядку пакування бітів порту, та індикатора з спільним анодом (LOW-активних рівней керування сегментами).

Таблиця 4.2

Сегменти індикатора	h	g	f	e	d	c	b	a
Упаковка лінії шини	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
Код «0»	1	1	0	0	0	0	0	0
Код «2»	1	0	1	0	0	1	0	0

Символічні описи кодів цифр розміщують в описовій частині програми.

Приклад 1

```

;-----
.equ CODE_0 = 0b11000000 ;код «0»
.equ CODE_2 = 0b10100100 ;код «2»
;-----

```

Сформовані таким чином коди індикації цифр розміщують у таблиці в пам'яті програм мікроконтролера після всіх підпрограм.

Приклад 2

```

;-----

```

```
.ORG $07F0
```

```
;-----  
;Таблиця кодів дешифрації індикатора  
LED_CODE:  
.DB  
CODE_0,CODE_1,CODE_2,CODE_3,CODE_4,CODE_5,CODE_6,CODE_7,CODE_  
8,  
CODE_9  
;-----
```

Для визначення кодів індикації по значенням цифр використовують програмний фрагмент який приведено у третьому прикладі. У регістрі R_ONE розташовується цифра, що виводиться у розряд індикатора.

Приклад 3

```
;-----  
ldi ZH,HIGH(2*LED_CODE) ;завантаження початку таблиці  
ldi ZL,LOW(2*LED_CODE) ;кодів  
add ZL,R_ONE ;визначення зміщення по таблиці  
lpm ;читання коду індикації в R0  
;-----
```

Звичайно необхідно передавати на дисплей числа, у яких значення десяткових розрядів невідомо, У таких випадках використовують двійково-десятькове перетворення з подальшим перекодуванням символів розрядів.

Нижче приведено підпрограму BCD–перетворення однобайтних чисел. Передача параметрів в підпрограму реалізується за допомогою файлового регістру TMP_BCD, результати перетворення розташовані у файлових регістрах R_HUND, R_TEN, R_ONE.

Визначення десяткових розрядів реалізується за рахунок виділення значень сотень десятків та одиниць у вихідному числі.

Приклад 4

```
;-----  
;Підпрограма BCD-перетворення  
;-----  
BCD:  
clr R_TEN ;очистка регістрів результату  
clr R_HUND  
;-----  
BCD_ONE:  
subi TMP_BCD,10 ;визначення кількості десятків
```

```

brcs BCD_TEN
inc R_TEN
rjmp BCD_ONE
;-----
BCD_TEN:
subi TMP_BCD,-10
mov R_ONE,TMP_BCD ;визначення кількості одиниць
BCD_TEN1:
subi R_TEN,10 ;визначення кількості сотень
brcs BCD_HUND
inc R_HUND
rjmp BCD_TEN1
BCD_HUND:
subi R_TEN,-10 ;визначення даних регістру десятків
;-----
ret
;-----

```

Драйвер дворозрядного індикатора з статичною індикацією

Складемо програму керування дворозрядним індикатором із статичною індикацією схему якого приведено на рис. 4.5.

У схемі використано регістри D2, D3 із статичним керуванням. Запис інформації відбувається при подачі на вхід “С” високого рівня напруги. Регістр D2 з’єднано з індикатором десятків Н1, а D3 з індикатором одиниць – Н2.

Коди цифр визначались для семисегментного індикатора з загальним анодом (L-активні рівні напруги), для схеми розташування сегментів, що приведено на рис.4.2. У таблиці 2 приведено коди двох цифр що виводяться (“2” та “0”).

Приклад 5

```

;-----
;Процесор ATmega16-16PI, частота 10 МГц
;-----
#include "m16def.inc" ;Файл опису процесора
;-----
.equ ini_DDRC = 0xff ;лінії налаштовані на передачу

.equ ini_PORTC = 0xff ;індикатори погашені
.equ ini_DDRD = 0b11000000 ;передачу/прийм
.equ ini_PORTD = 0 ;початковий стан – «0»
.equ CS_TEN = PD6 ;керування регістром десятків
.equ CS_ONE = PD7 ;керування регістром одиниць
;-----
.def TMP = R16 ;символічне ім'я регістру R16

```

```

;-----
.equ CODE_0 = 0b11000000 ;код «0»
.equ CODE_2 = 0b10100100 ;код «2»
;-----
;Програма
;-----
.CSEG
.ORG 0
    ;----- Ініціалізація

init:
ldi TMP,ini_DDRC ; ініціалізація порту C
out DDRC,TMP
ldi TMP,ini_PORTC
out PORTC,TMP
ldi TMP,ini_DDRD ; ініціалізація порту D
out DDRD,TMP
ldi TMP,ini_PORTD
out PORTD,TMP
ldi TMP,high(RAMEND) ;завантаження адреси початку стеку
out SPH,TMP
ldi TMP,low (RAMEND)
out SPL,TMP
    ;-----

;----- Основна програма
MAIN:
rcall DISPLAY ;підпрограма роботи з дисплеєм
rjmp MAIN
;-----
;Підпрограма роботи із дисплеєм
DISPLAY:
DIS_TEN:
ldi TMP, CODE_2 ;передача у порт коду десятків
out PORTC,TMP
sbi PORTD,CS_TEN ;строб регістра десятків
cbi PORTD,CS_TEN
DIS_ONE:
ldi TMP, CODE_0 ;передача у порт коду одиниць
out PORTC,TMP
sbi PORTD,CS_ONE ;строб регістра одиниць
    cbi PORTD,CS_ONE

DIS_END:
ret
;-----
.exit
;-----

```

Драйвер дворозрядного індикатора з динамічною індикацією та паралельною шиною

Складемо програму керування дворозрядним індикатором із динамічною індикацією схему якого приведено на рис. 4.6.

У програмі забезпечено вивід на індикатор довільного числа, значення якого розташовано у регістрі TMP (файловий регістр).

Для забезпечення стабільної частоти індикації розрядів використаємо таймер/рахівник та систему переривань. Частота генератора мікроконтролера складає 10МГц. Оберемо період комутації розрядів індикатора 204мкс. Такий період забезпечується у разі вибору попереднього подільника таймера $f/8$ та розміщення у регістр TCNT0 початкового нулевого значення. Такий вибір виключає процедуру завантаження регістру TCNT0 під час ініціалізації та у підпрограмі переривань. Індикація розрядів забезпечується програмними фрагментами, що розташовані у програмі переривань.

Виділення значень десяткових розрядів забезпечується підпрограмою BCD-перетворення, яка описана у четвертому прикладі. Перекодування чисел забезпечується у додатковій підпрограмі CODE.

Приклад 6

```
;-----  
;Процесор ATmega16-16PI, частота 10 МГц  
;-----  
.include "m16def.inc" ;Файл опису процесора  
;-----  
.equ ini_DDRC = 0xff ;усі лінії налаштовані на передачу  
    .equ ini_PORTC = 0xff ;початковий стан - індикатори погашені  
  
.equ ini_DDRD = 0b11101000 ;лінії налаштовані на передачу/прийм  
.equ ini_PORTD = 0b11100000 ;початковий стан ліній керування  
.equ CS_ONE = PD7 ;лінія керування розрядом одиниць  
.equ CS_TEN = PD6 ;лінія керування розрядом десятків  
.equ LED_CS = PD3 ;лінія запису у буферний регістр  
;-----  
.def TMP = R16 ;символічне ім'я регістру R16  
.def TMP_BCD = R17 ;символічне ім'я регістру R17  
.def R_ONE = R18 ;символічне ім'я регістру R18  
.def R_TEN = R19 ;символічне ім'я регістру R19  
.def R_HUND = R20 ;символічне ім'я регістру R20  
.def RC_ONE = R21 ;символічне ім'я регістру R21  
.def RC_TEN = R22 ;символічне ім'я регістру R22
```

```

.def RC_HUND = R23 ;символічне ім`я регістру R23
;-----
.equ ini_TIMSK = (1<<TOIE0)
.equ TC0_ON_8=0b00000010 ;вмикання TC0/8
;-----
.equ CODE_0 = 0xC0 ;код "0"
.equ CODE_1 = 0xF9 ;код "1"
.equ CODE_2 = 0xA4 ;код "2"
.equ CODE_3 = 0xB0 ;код "3"
.equ CODE_4 = 0x99 ;код "4"
.equ CODE_5 = 0x92 ;код "5"
.equ CODE_6 = 0x82 ;код "6"
.equ CODE_7 = 0xF8 ;код "7"
.equ CODE_8 = 0x80 ;код "8"
.equ CODE_9 = 0x90 ;код "9"
;-----
;ПРОГРАМА
;-----
.CSEG
.ORG $0000
rjmp init ;Вектор розгалуження при рестарті
.ORG OVFOaddr
rjmp INT_TC0_OVF ;Вектор переривань переповнення TC0
;-----
.ORG $0030
init:
;-----Ініціалізація портів
ldi TMP,ini_DDRC ;ініціалізація порту C
out DDRC,TMP
ldi TMP,ini_PORTC
out PORTC,TMP
ldi TMP,ini_DDRD ;ініціалізація порту D
out DDRD,TMP
ldi TMP,ini_PORTD
out PORTD,TMP

;-----Ініціалізація стеку
ldi TMP,high(RAMEND);завантаження адреси початку стеку
out SPH,TMP
ldi TMP,low (RAMEND)
out SPL,TMP
;-----Ініціалізація таймера TC0
ldi TMP, ini_TIMSK ;дозвіл переривань переповнення TC0
out TIMSK,TMP
ldi TMP, TC0_ON_8 ;вмикання TC0
out TCCR0,TMP
sei ;загальний дозвіл переривань
;-----
;-----
;Основна програма
;-----
ldi TMP_BCD,180 ;завантаження числа, що виводиться
call BCD ;BCD-перетворення числа
call CODE ;визначення кодів цифр

```

```

MAIN:
rjmp MAIN
;-----
;-----
;Підпрограма BCD-перетворення
;отримує байт даних з TMP_BCD,
;повертає цифри розрядів у регістри R_HUND, R_TEN, R_ONE
;-----
BCD:
;-----
;Підпрограма описана у четвертому прикладі
;-----
ret
;-----
;-----
;Підпрограма перекодування
;отримує цифри розрядів з регістрів R_HUND, R_TEN, R_ONE
;повертає коди цифр розрядів у регістри RC_HUND, RC_TEN, RC_ONE
;-----
CODE:
ldi ZH,HIGH(2*LED_CODE) ;завантаження початку таблиці кодів
CODE_ONE:
ldi ZL,LOW(2*LED_CODE)
add ZL,R_ONE ;визначення зміщення по таблиці
lpm ;читання коду індикації в R0
mov RC_ONE,R0
CODE_TEN:
ldi ZL,LOW(2*LED_CODE)
add ZL,R_TEN ;визначення зміщення по таблиці
lpm ;читання коду індикації в R0

mov RC_TEN,R0
CODE_HUND:
ldi ZL,LOW(2*LED_CODE)
add ZL,R_HUND ;визначення зміщення по таблиці
lpm ;читання коду індикації в R0
mov RC_HUND,R0
ret
;-----
;-----
;Підпрограма переривань переповнення TC0
;отримує коди цифр розрядів із регістрів RC_HUND, RC_TEN, RC_ONE
;-----
INT_TC0_OVF:
sbis PORTD, CS_TEN ;розгалуження індикації одиниць чи десятків
rjmp INT_ONE
sbis PORTD, CS_ONE
rjmp INT_TEN
INT_ONE:
sbi PORTD,CS_TEN
out PORTC,RC_ONE ;вивід у порт коду одиниць
sbi PORTD,LED_CS ;строб запису у буферний регістр
nop
cbi PORTD,LED_CS

```

```

cbi PORTD,CS_ONE ;активізація розряду одиниць
rjmp INT_TC0_END
INT_TEN:
sbi PORTD,CS_ONE
out PORTC,RC_TEN ;вивід у порт коду десятків
sbi PORTD,LED_CS ;строб запису у буферний регістр
nop
cbi PORTD,LED_CS
cbi PORTD,CS_TEN ;активізація розряду десятків
INT_TC0_END:
reti
;-----
.ORG $0100
;-----
;Таблиця кодів дешифрації індикатора
LED_CODE:
.DB
CODE_0,CODE_1,CODE_2,CODE_3,CODE_4,CODE_5,CODE_6,CODE_7,CODE_
8,
CODE_9
;-----
.exit
;-----

```

4.2 Завдання до лабораторної роботи

З використанням мови асемблеру необхідно розробити програму мікроконтролера, яка забезпечує вивід символів у задані розряди багаторозрядного семисегментного індикатора. Варіанти вихідних даних розташовані в табл. 4.1. Створити макети згідно рис.4.7 та рис.4.8

Таблиця 4.1

№ варіанту	Значення для 1 макету	Значення для 2 макету
1	101	1011
2	102	1012
3	103	1013
4	104	1014
5	105	1015
6	106	1016
7	107	1017
8	108	1018
9	109	1019
10	110	1020
11	111	1021
12	112	1022
13	113	1023
14	114	1024
15	115	1025
16	116	1026
17	117	1027
18	118	1028
19	119	1029
20	120	1030
21	121	1031

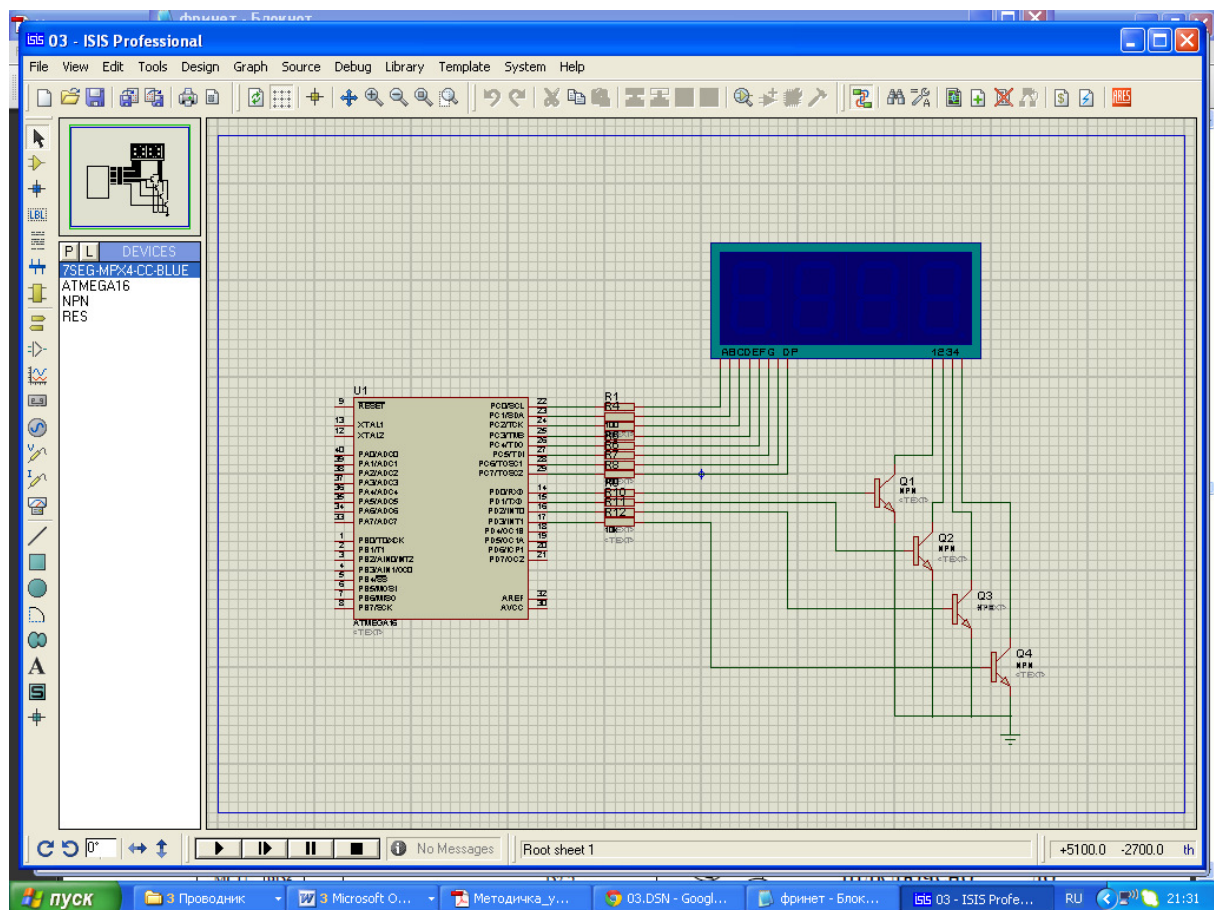


Рис.4.8

У стенді використовуються наступні елементи керування LED- дисплеєм:

4.3 Зміст звіту

1. Тема та мета роботи.
2. Завдання до лабораторної роботи.
3. Текст програми з поясненнями.
4. Таблиця вмісту регістрів.
5. Результат роботи програми.
6. Висновки

4.4 Контрольні запитання

1. Визначіть типи індикаторів.
2. Визначіть типи LED-індикаторів.
3. У чому полягає метод статичної індикації, та як реалізується схемотехнічно?
4. У чому полягає метод динамічної індикації, та як він реалізується схемотехнічно?
5. Яким чином готуються дані для виводу на багаторозрядні семисегментні індикатори?
6. Які існують особливості програм драйверів багаторозрядних семисегментних індикаторів для метода статичної індикації?
7. Які існують особливості програм драйверів багаторозрядних семисегментних індикаторів для метода динамічної індикації?

