Міністерство освіти і науки України

Українська академія друкарства

**ЗВІТ**

з лабораторної роботи № 5

на тему «Семисегментний індикатор. Статична індикація»

Виконала:

студ. групи КН-41

Гончарук Т. О.

Перевірив:

к. ф-м. н., доц. Пушак А. С.

Львів — 2021

**Тема роботи:** Семисегментний індикатор. Статична індикація.

**Мета роботи:** Ознайомлення з семисегментним індикатором. Вивчення схем підключення семисегментного індикатора до мікроконтролера. Вивчення алгоритму виведення чисел на індикатор.

**Хід роботи**

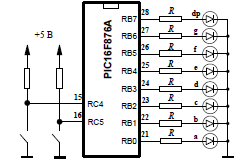
1. У середовищі Proteus складіть схему приведену на рис. 5.1.

2. У середовищі MPLAB створити проєкт з кодом для виводу на індикатор чисел у зростаючому та спадному порядку починаючи з нуля за кожним натисканням певної з кнопок.

3. Перевірити його роботу у середовищі Proteus.

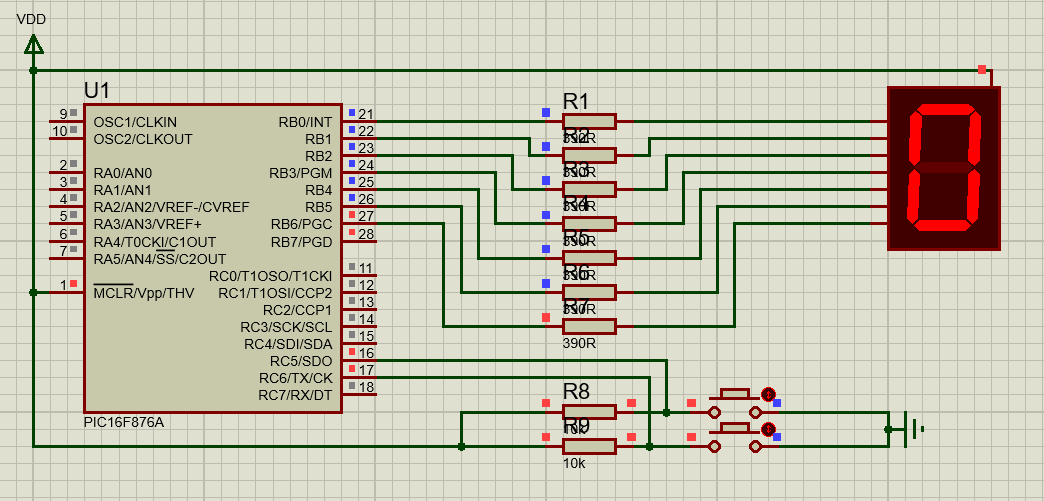
**Результати виконання**

1. У середовищі Proteus складіть схему приведену на рис. 5.1.



**Рис. 5.1. Схема експериментальної макетної плати для вивчення статичної індикації**

У середовищі Proteus дана схема виглядає наступним чином:

****

**Рис. 5.2. Схема в симуляторі Proteus**

2. У середовищі MPLAB створити проєкт з кодом для виводу на індикатор чисел у зростаючому та спадному порядку починаючи з нуля за кожним натисканням певної з кнопок.

#pragma config FOSC = HS

#pragma config WDTE = OFF

#pragma config PWRTE = OFF

#pragma config BOREN = OFF

#pragma config LVP = OFF

#pragma config CPD = OFF

#pragma config WRT = OFF

#pragma config CP = OFF

#include <xc.h>

#define \_XTAL\_FREQ 20000000

void select(int j) {

switch(j)

{

case 0:

PORTB = ~0b00111111;

break;

case 1:

PORTB = ~0b00000110;

break;

case 2:

PORTB = ~0b01011011;

break;

case 3:

PORTB = ~0b01001111;

break;

case 4:

PORTB = ~0b01100110;

break;

case 5:

PORTB = ~0b01101101;

break;

case 6:

PORTB = ~0b01111101;

break;

case 7:

PORTB = ~0b00000111;

break;

case 8:

PORTB = ~0b11111111;

break;

case 9:

PORTB = ~0b01101111;

break;

}

}

void main(void)

{

unsigned int j=0;

TRISB = 0b00000000;

TRISC = 0b11111111;

PORTB = 0b00000000;

while(1)

{

select(j);

if(RC5==0)

{

j++;

if(j==10)

{

j=0;

}

select(j);

while(RC5==0)

{

\_\_delay\_ms (10);

}

}

if(RC6==0)

{

j--;

if(j==-1)

{

j=9;

}

select(j);

while(RC6==0)

{

\_\_delay\_ms (10);

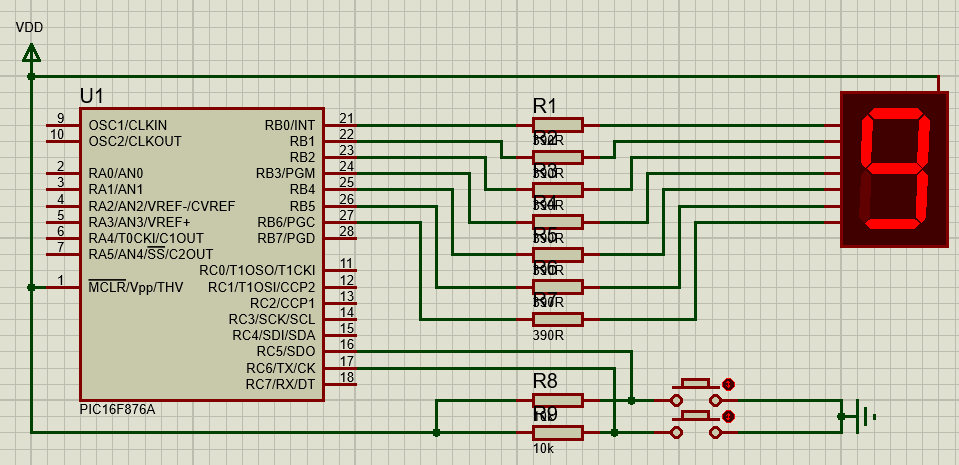
}

}

}

}

3. Перевірити роботу коду на симуляторі Proteus.



**Рис. 5.3. Симуляція коду в програмі Proteus**

**Контрольні запитання**

1. Які бувають типи семисегментних індикаторів?

Існує два типи індикаторів: з спільним катодом або спільним анодом.

2. Опишіть схему під’єднання індикаторів до мікроконтролера?

Сегменти індикатора позначають буквами a, b, c, d, e, f, g (a – верхній елемент, далі букви присвоюють сегментам за годинниковою стрілкою; g – центральний сегмент; dp – крапка).

8 незалежних елементів, кожний з яких може знаходитися в одному з двох станів – “горить” або не “горить”, дають можливість відобразити десяткові числа і деякі символи.

Кожен сегмент підключається до певного виводу певного порту. Посилаючи в певний порт відповідне число, на його виводах будуть виставлені відповідні логічні рівні для засвічування певних сегментів індикатора.

3. Які символи можна вивести з допомогою семисегментних індикаторів?

Назву семисегментний індикатор отримав у зв’язку з тим, що зображення символу формується з допомогою семи окремо керуючих елементів – сегментів. Ці елементи дозволяють відображати будь-яку цифру 0..9, а також деякі символи, наприклад: “-“, “A”, “b”, “C”, “d”, “E”, “F” та інші.

4. В якому режимі повинні працювати виводи порту для під’єднання семисегментного індикатора із спільним анодом?

Сегмент засвічується поданням логічної 1.

**Висновки**

Під час виконання лабораторного практикуму я написала алгоритм для виводу на індикатор чисел у зростаючому та спадному порядку починаючи з нуля за кожним натисканням певної з кнопок. Даний алгоритм був протестований у симуляторі Proteus на відповідній схемі.