

Міністерство освіти і науки України  
Українська академія друкарства

**ЗВІТ**  
з лабораторної роботи № 6  
на тему «Семисегментні індикатори. Динамічна індикація»

Виконала:

студ. групи КН-41

Гончарук Т. О.

Перевірив:

к. ф-м. н., доц. Пушак А. С.

**Тема роботи:** Семисегментні індикатори. Динамічна індикація.

**Мета роботи:** Вивчення схеми підключення семисегментних індикаторів для виводу багаторозрядних чисел. Вивчення алгоритму для динамічної індикації багаторозрядних чисел.

### Хід роботи

1. У середовищі Proteus складіть схему приведену на рис. 6.1.
2. У середовищі MPLAB створити проєкт з кодом для виводу на індикатор чисел у зростаючому порядку починаючи з нуля.
3. Перевірити його роботу у середовищі Proteus.

### Результати виконання

1. У середовищі Proteus складіть схему приведену на рис. 6.1.

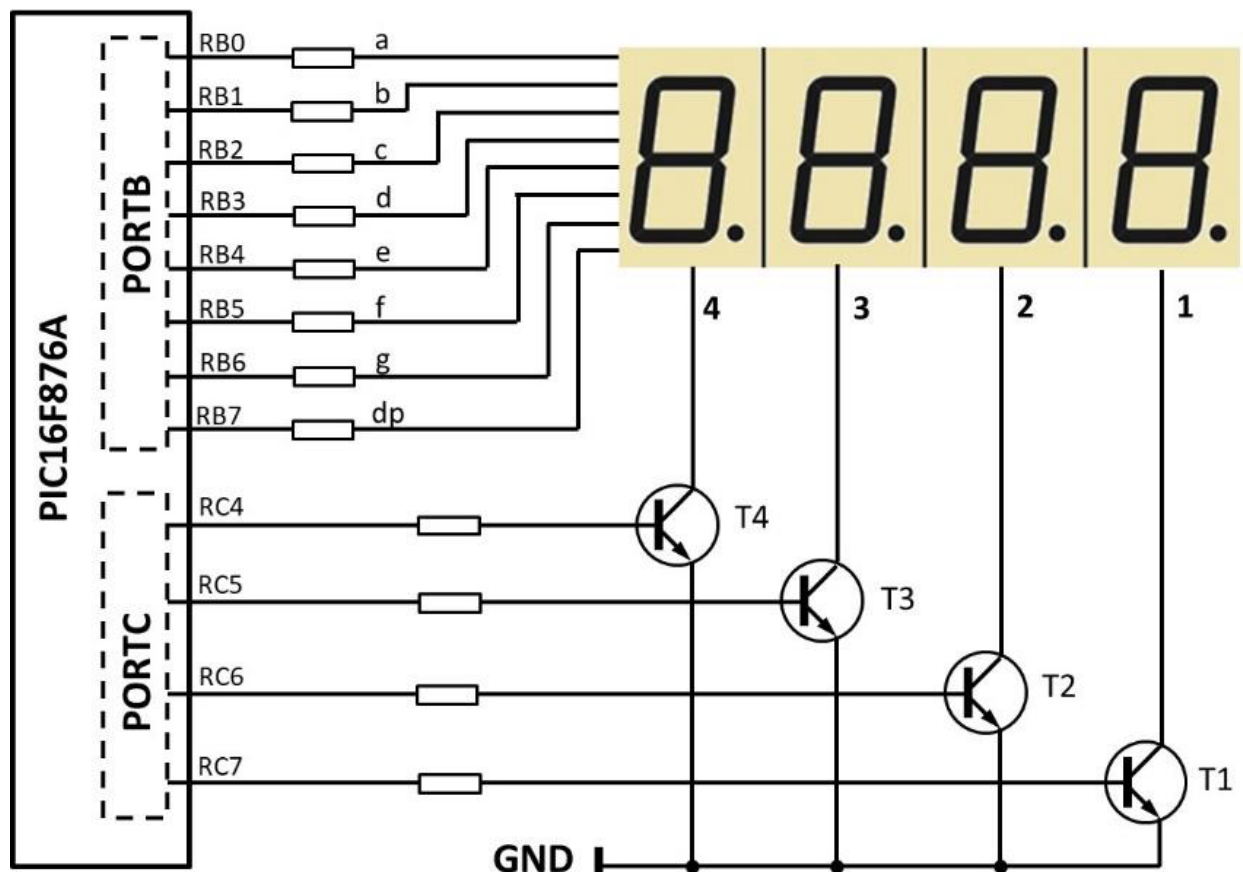
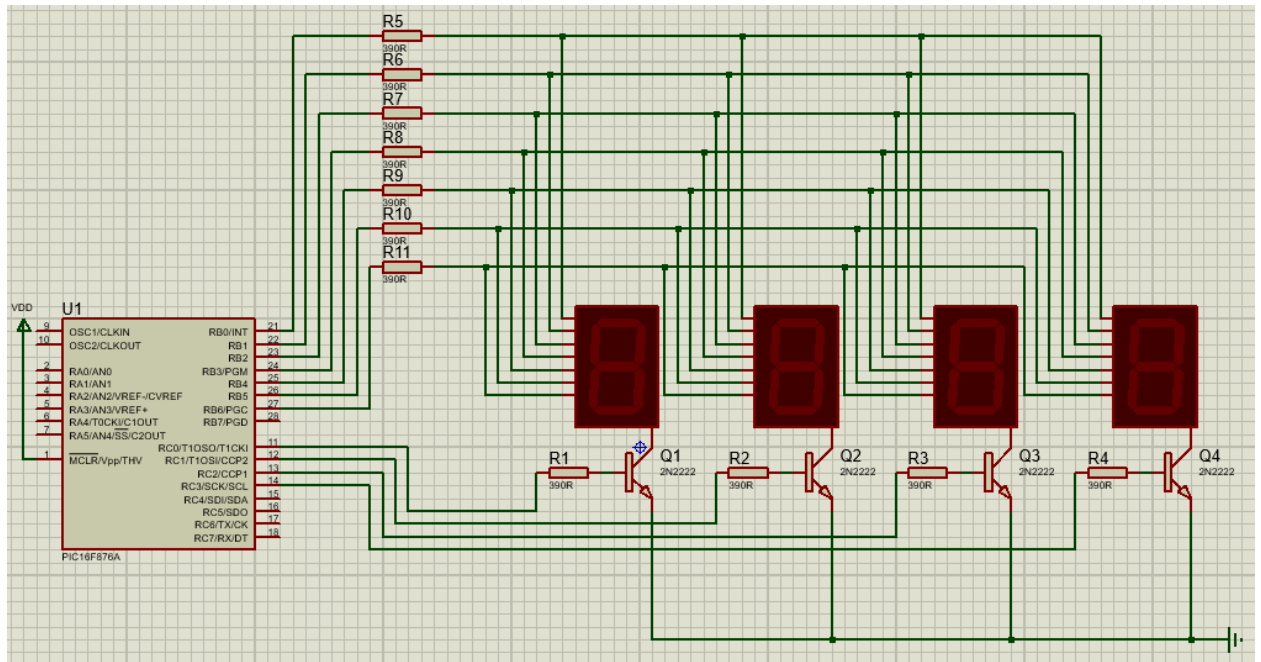


Рис. 6.1. Схема макетної плати для реалізації динамічної індикації

У середовищі Proteus дана схема виглядає наступним чином:



**Рис. 6.2. Схема в симуляції Proteus**

2. У середовищі MPLAB створити проєкт з кодом для виводу на індикатор чисел у зростаючому порядку починаючи з нуля.

```
#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
#define dig1 RC0
#define dig2 RC1
#define dig3 RC2
#define dig4 RC3
```

```
unsigned int a,b,c,d,e,f,g,h;
```

```

int i = 0;
int flag = 0;
unsigned int seg[]={0b00111111,
                    0b00000110,
                    0b01011011,
                    0b01001111,
                    0b01100110,
                    0b01101101,
                    0b01111101,
                    0b00000111,
                    0b11111111,
                    0b01101111
                    };

void main(void){
    TRISC = 0b00000000;
    TRISB = 0b00000000;
    PORTC = 0b00000000;
    PORTB = 0b00000000;

    while(1)
    {
        a=i%10;
        b=i/10;
        c=b%10;
        d=b/10;
        e=d%10;
        f=d/10;
        g=f%10;
        h=f/10;

        PORTB=seg[g];
        dig1=1;
        __delay_ms(5);
    }
}

```

```

    dig1=0;
    PORTB=seg[e];
    dig2=1;
    __delay_ms(5);
    dig2=0;
    PORTB=seg[c];
    dig3=1;
    __delay_ms(5);
    dig3=0;
    PORTB=seg[a];
    dig4=1;
    __delay_ms(5);
    dig4=0;

    if(flag>=10)
    {
        i++;
        flag=0;
    }
    flag++;

}
}

```

### 3. Перевірити роботу коду на симуляторі Proteus.

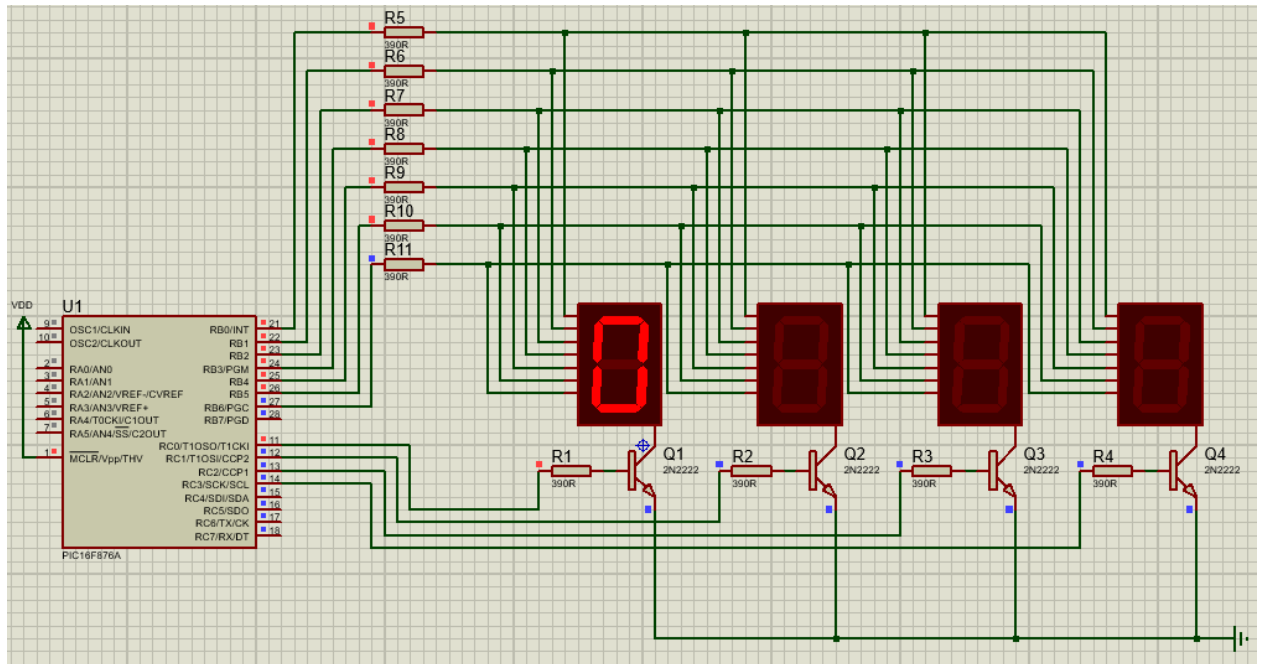


Рис. 6.3. Симуляція коду в програмі Proteus

#### Контрольні запитання

1. В чому полягає динамічна індикація багаторозрядних чисел з допомогою семисегментних індикаторів?

Для збільшення кількості індикаторів і зменшення апаратних засобів для реалізації індикації в мікропроцесорних системах використовується динамічна індикація. Суть її полягає в тому, що один і той самий управляючий сигнал керує відображенням інформації на кількох елементах індикації в різні, строго визначені, проміжки часу.

2. Опишіть алгоритм роботи динамічної індикації?

Розглянемо алгоритм динамічної індикації. Нехай маємо деяке число  $x$ , яке треба вивести на індикатор. Кількість розрядів даного числа –  $k$ .

1. Знаходимо значення першого розряду ( $n = x \% 10$ ) і включаємо відповідні сегменти для його індикації ( $PORTB = \text{“відповідний код”}$ ). Включаємо перший розряд індикатора, тобто на вивід  $RC7$  мікроконтролера подаємо логічну “1”.

2. Робимо деяку часову затримку для нормального візуального сприйняття виведеної цифри.

3. Очищаємо порт В ( $PORTB = 0$ ) і відключаємо перший розряд ( $RC7 = 0$ ). Після чого робимо невеличку затримку для повної релаксації свічення першого розряду.

4. Повторюємо пункти 1-3 для наступних розрядів, після чого весь процес повторюється знову.

3. Як впливає інерційність зору на частоту повторення засвічування розрядів числа?

Необхідно забезпечити достатній час світіння одного індикатора, для того щоб не зменшувалась яскравість. Також необхідно забезпечити таку частоту перебору індикаторів, щоб не було помітно мерехтіння. Тому для відтворення даних на різних 7SEG необхідним є почергове включення цих індикаторів. Період включення слід обрати так, щоб для людського сприйняття усі індикатори здавались увімкненими.

### **Висновки**

Під час виконання лабораторного практикуму я написала алгоритм для виводу на індикатор чисел у зростаючому порядку починаючи з нуля. Даний алгоритм був протестований у симуляторі Proteus на відповідній схемі.