

Міністерство освіти і науки України  
Українська академія друкарства

**ЗВІТ**  
з лабораторної роботи № 8  
на тему «Вивчення алгоритму виведення слів,  
багаторозрядних чисел та нестандартних символів на  
дисплей LCD1602A»

Виконала:

студ. групи КН-41

Гончарук Т. О.

Перевірив:

к. ф-м. н., доц. Пушак А. С.

**Тема роботи:** Вивчення алгоритму виведення слів, багаторозрядних чисел та нестандартних символів на дисплей LCD1602A

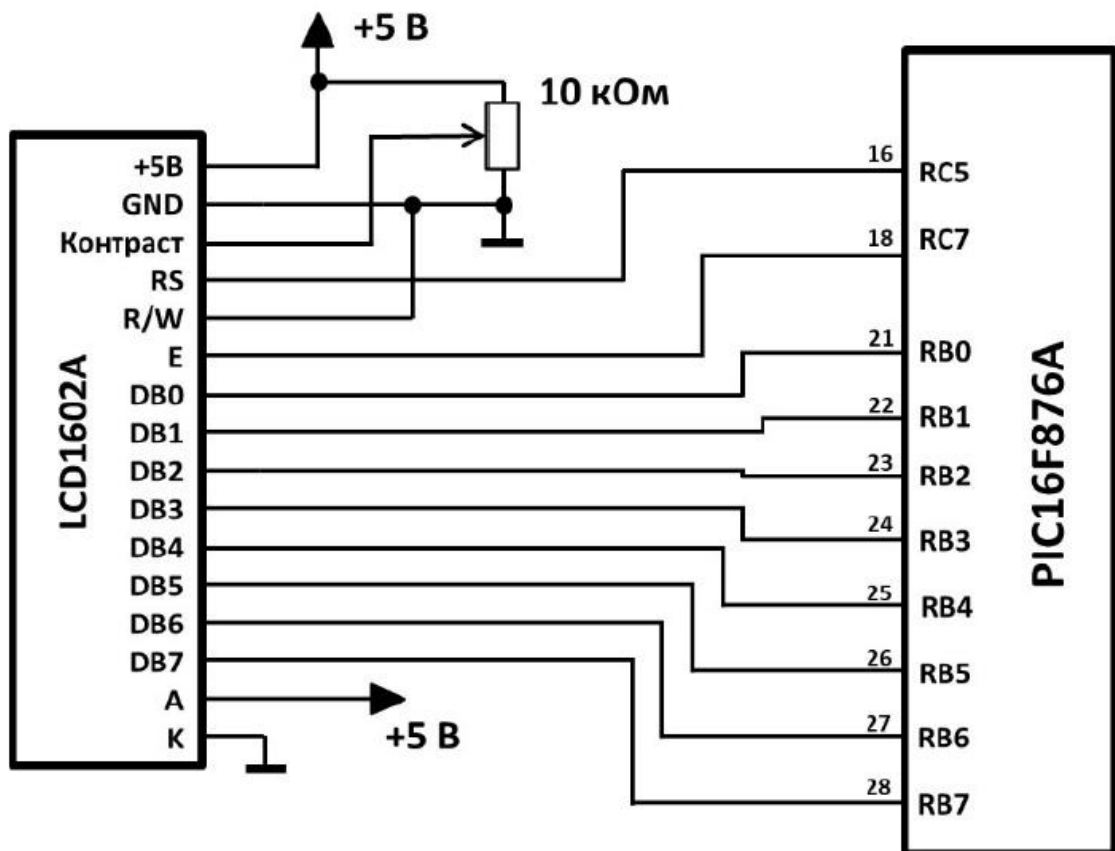
**Мета роботи:** Вивчення алгоритму виведення на дисплей слів та багаторозрядних чисел.

### Хід роботи

1. У середовищі Proteus складіть схему приведену на рис. 8.1.
2. У середовищі MPLAB створити проєкт з кодом для біжучого символу на дисплей зі зміною напрямку руху на останніх позиціях.
3. Перевірити його роботу у середовищі Proteus.

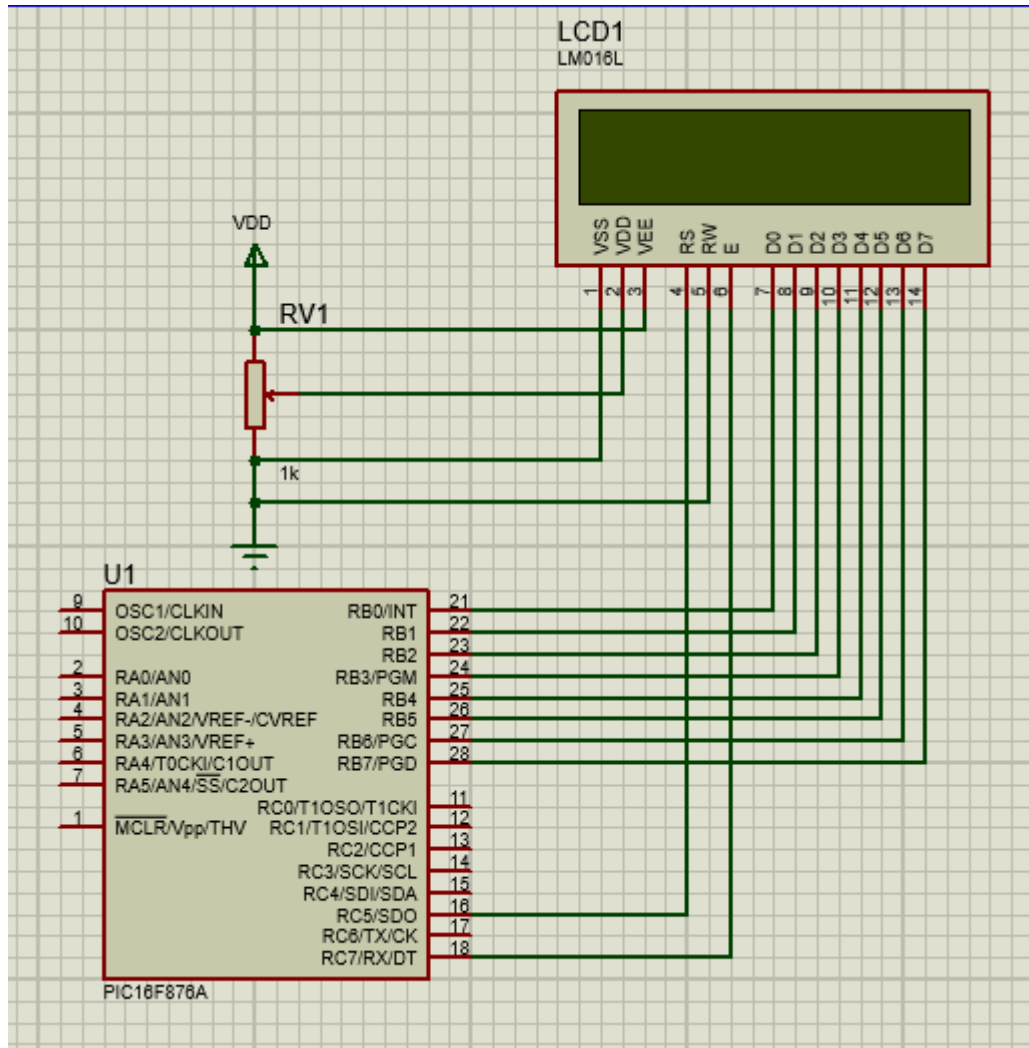
### Результати виконання

1. У середовищі Proteus складіть схему приведену на рис. 8.1.



**Рис. 8.1.** Схема макетної плати для роботи дисплея у 8-бітному режимі

У середовищі Proteus дана схема виглядає наступним чином:



**Рис. 8.2. Схема в симуляторі Proteus**

2. У середовищі MPLAB створити проєкт з кодом для біжучого символу на дисплей зі зміною напрямку руху на останніх позиціях.

```
#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 strob (void)
{
    RC7 = 0;
    __delay_us (50);
    RC7 = 1;
    __delay_us (50);
}

char x;
char* v;
unsigned int i;
int A[16]={128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143};
void main (void)
{
    TRISB = 0b00000000;
    TRISC = 0b00000000;
    PORTB = 0b00000000;
    PORTC = 0b00000000;

    __delay_ms (20);
    RC5=0;
    RC7=1;
    __delay_us (100);

    PORTB=0b00110000;
    __delay_ms (5);
    strob ();

    PORTB=0b00110000;
    __delay_ms (5);
    strob ();
}

```

```
PORTB=0b00110000;  
__delay_ms (5);  
strob ();
```

```
PORTB=0b00111000;  
__delay_us (100);  
strob ();
```

```
PORTB=0b00001100;  
__delay_us (100);  
strob ();
```

```
PORTB=0b00000000;  
__delay_us (100);  
strob ();
```

```
RC5 = 0;  
PORTB = 128;  
__delay_us (100);  
strob ();
```

```
i = 0;  
RC5 = 0;  
PORTB = A[i];  
__delay_us (50);  
strob();
```

```
while (1)  
{  
    RC5 = 0;  
    PORTB = A[i];  
    __delay_us (50);  
    strob();
```

```
RC5 = 1;
PORTB = 0b11110011;
__delay_us (50);
strob();
__delay_ms (100);
```

```
RC5 = 0;
PORTB = A[i];
__delay_us (50);
strob();
```

```
RC5 = 1;
PORTB = ' ';
__delay_us (50);
strob();
__delay_ms (1);
i++;
if(i==16)
{
    while(i>=1)
    {
        RC5 = 0;
        PORTB = A[i];
        __delay_us (50);
        strob();
```

```
RC5 = 1;
PORTB = 0b11110011;
__delay_us (50);
strob();
__delay_ms (100);
```

```

RC5 = 0;

PORTB = A[i];

__delay_us (50);

strob();

RC5 = 1;

PORTB = ' ';

__delay_us (50);

strob();

__delay_ms (1);

i--;

}

}

}

}

```

3. Перевірити роботу коду на симуляторі Proteus.

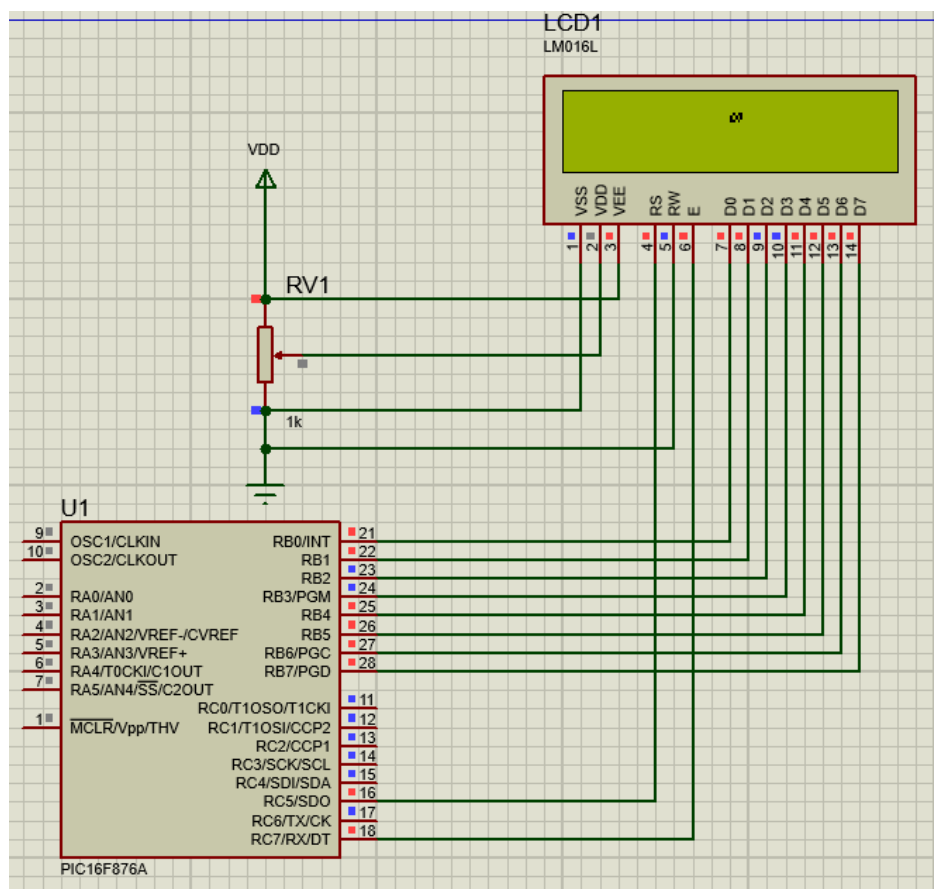


Рис. 8.3. Симуляція коду в програмі Proteus

## Контрольні запитання

1. Як вивести на дисплей символи грецького алфавіту?

В технічній документації дисплея приведено таблицю кодів різних символів, які можна вивести на дисплей.

2. Як вивести на дисплей значення певної змінної?

3. Опишіть алгоритм виводу на дисплей слів та речень?

Для виводу слів на дисплей необхідно ввести певну змінну типу `char*` або `string`, якій буде присвоюватись слово або набір символів (коротке речення) і змінну для перебору символів у заданому слові. Алгоритм виведення слів на дисплей зводиться до почергового виведення символів заданого слова по порядку з першого по останній. Рахунок символів починається з нуля.

4. Опишіть алгоритм виводу на дисплей багаторозрядних чисел?

У цьому випадку також методом перебору розрядів числа виводиться на дисплей їх значення, однак спочатку значення кожного розряду необхідно перетворити у символьний тип. У мові C++ для перетворення цілого числа у символьний тип використовується наступний оператор: `x=a+'0'`;

де `a` – число цілого типу, а `x` – значення цього числа як змінна символьного типу.

## Висновки

Під час виконання лабораторного практикуму я написала алгоритм для виводу біжучого символу на дисплей зі зміною напрямку руху на останніх позиціях. Даний алгоритм був протестований у симуляторі Proteus на відповідній схемі.