

Національний технічний університет України «КПІ ім. Ігоря Сікорського»  
Факультет Інформатики та Обчислювальної Техніки  
Кафедра інформаційних систем та технологій

Лабораторна робота №5  
з дисципліни «Програмування мікропроцесорних систем»

на тему

«Програмування мікропроцесорних систем.  
Робота з LCD - дисплеєм.»

Виконав:  
студент групи ІІІ-11  
Дякунчак І.  
Викладач:  
доц. Голубєв Л. П.

Київ – 2024

## Зміст

Зміст.....	2
1. Постановка задачі.....	3
2. Виконання.....	4
3. Контрольні питання.....	8
4. Висновок.....	9
5. Додатки.....	9

# 1. Постановка задачі

**Мета:** навчити роботі з Alphanumeric I2C Liquid Crystal Display (1602 LCD), за допомогою мікроконтролера Arduino.

## Завдання до роботи:

**В кожній з робіт потрібно розробити схеми та заставити її працювати за правилами, що викладені в задачах. До кожної із задач у звіті повинні бути намальовані відповідні схеми.**

За допомогою сервісу tinkercad.com створити наступні проекти:

В середовищі tinkercad створити на малий макетній платі принципіальну схему пристрою, а в розділі «Текст» написати програму.

### Задача №1.

Записати ваше прізвище у першому рядку дисплею, ім'я – у другому. Дані на екрані повинні прокручуватися. Для парних варіантів справа наліво, для непарних зліва направо.

### Задача №2.

Створити проект, в якому на екрані LCD - дисплею у першому рядку висвітлювалася б температура по Цельсію, у другому по Фаренгейту.

### Задача №3.

Розташуйте інформацію з вашим прізвищем так, щоб непарні літери стояли на першому, парні на другому рядку дисплею, як кажуть «шахматкою».

### Задача №4.

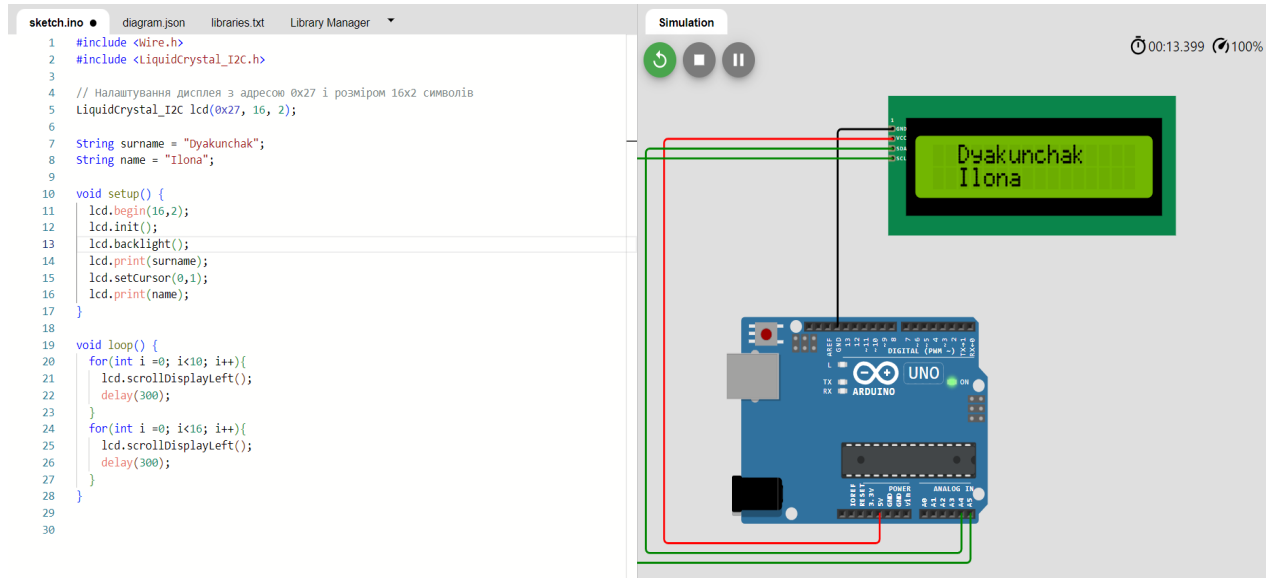
Написати у другому рядку дисплею своє прізвище кирилицею, створивши відповідні літери української абетки. Якщо у прізвищі більше 8 літер, обмежитися першими 8 літерами.

### Задача №5

За допомогою інтернет-сервісу WokWi створити проект, який виводить на OLED-екран прізвище виконавця та номер групи на протязі 3 сек., а потім за допомогою потенціометра змінювати значення на аналоговому виході та відображати це на OLED-екрані у вигляді графіка.

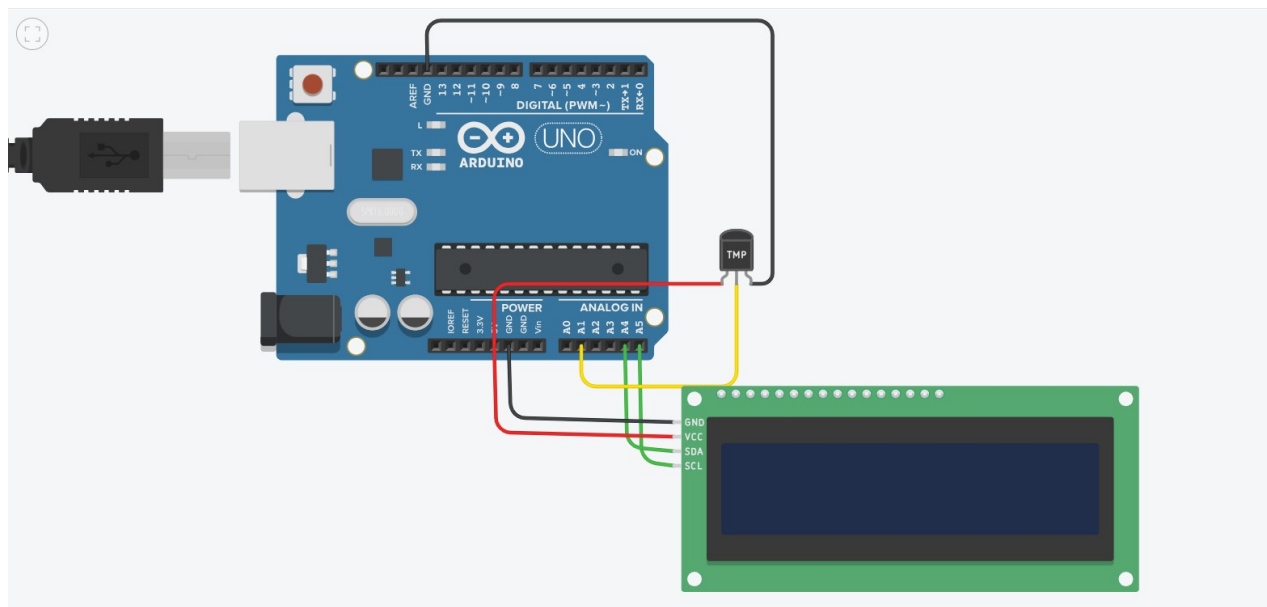
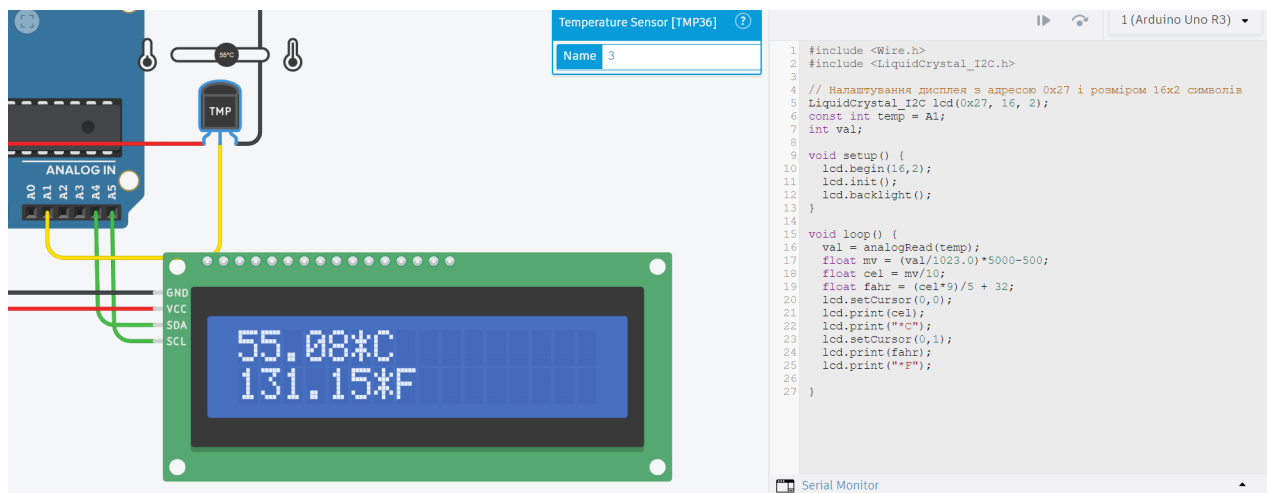
## 2. Виконання

Спочатку потрібно було створити перший проект, суть якого полягає в записі на дисплеї свого імені та прізвища та прокручування їх справа на ліво, код та знімки екрану наведені нижче:

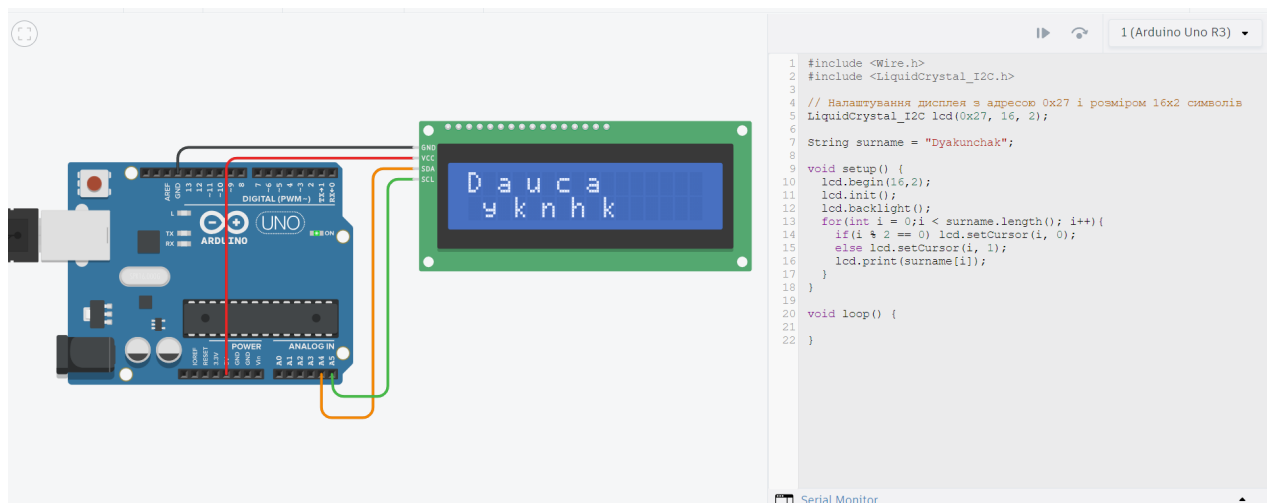


Як бачимо, усе працює, отже можна переходити до створення другого проекту, в якому на екрані LCD - дисплею у першому рядку висвітлювалася б температура по Цельсію, у другому по Фаренгейту.

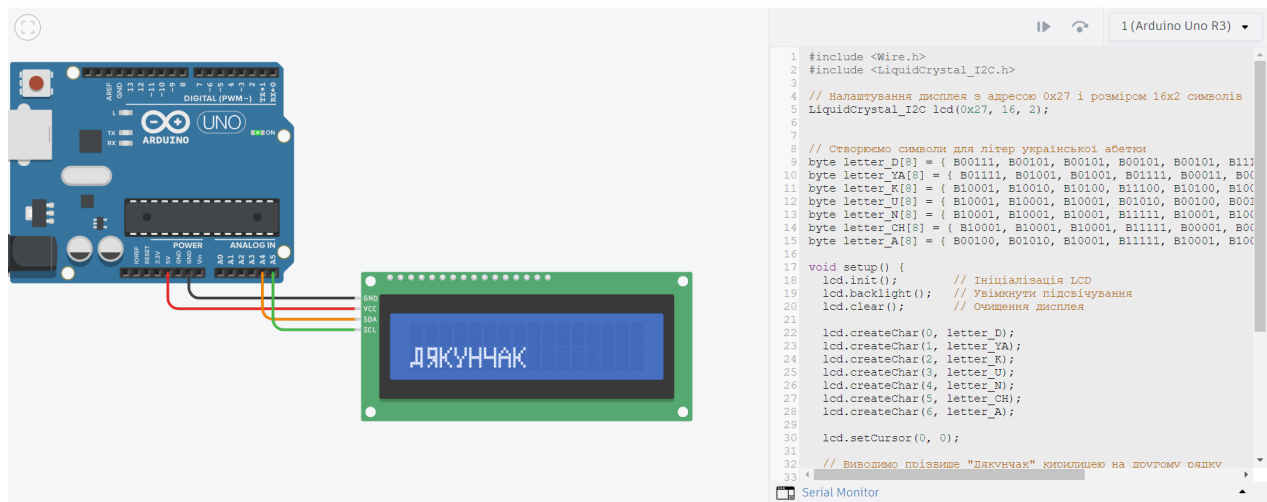
Результати роботи наведено на знімку екрану нижче, повний код програми можна знайти у додатках, також там знаходиться і посилання на проект.



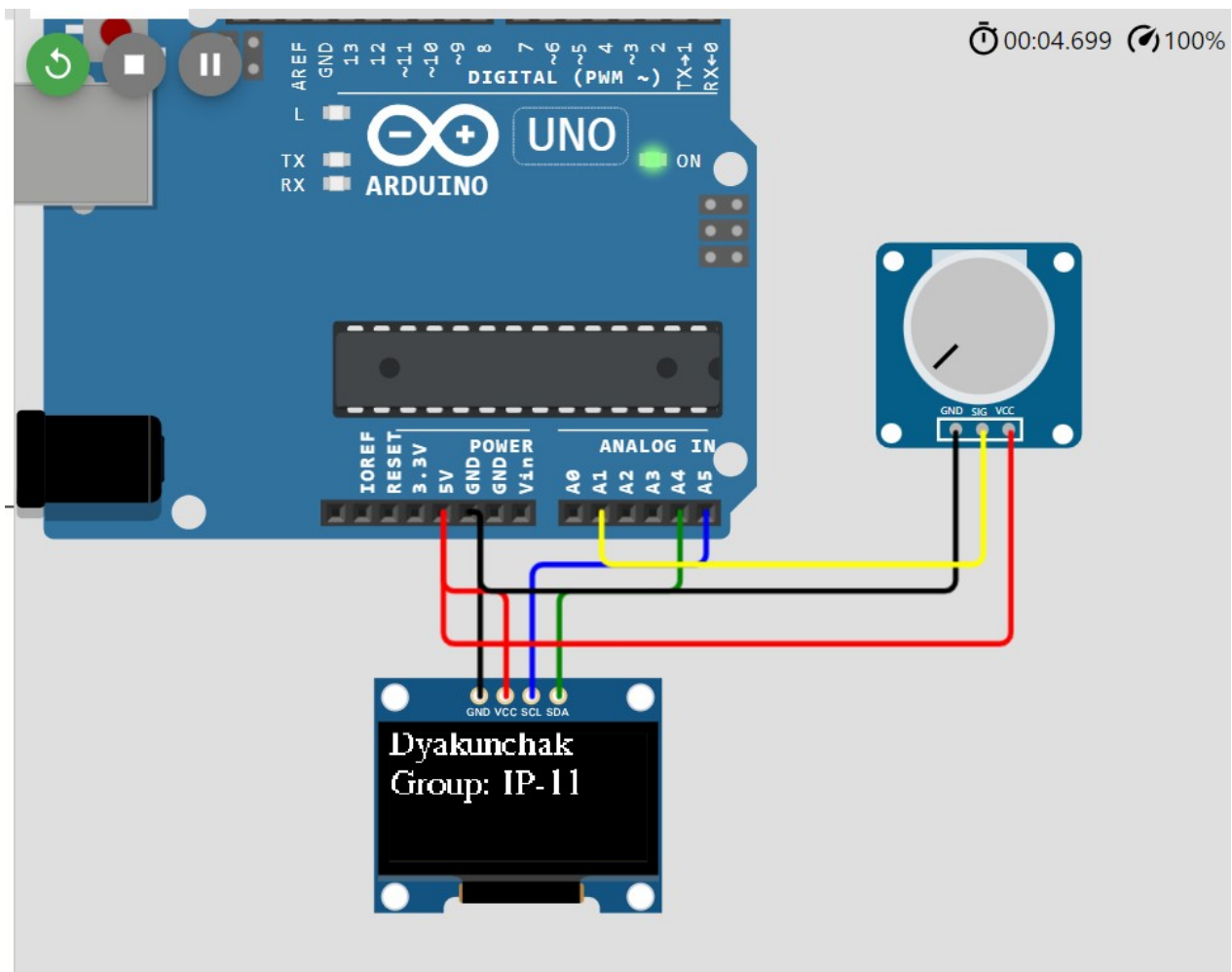
Як бачимо на екрані дисплею, усе працює, отже можна переходити до створення третього проекту, в якому літери прізвища будуть розташовуватися на екрані “шахматкою”.

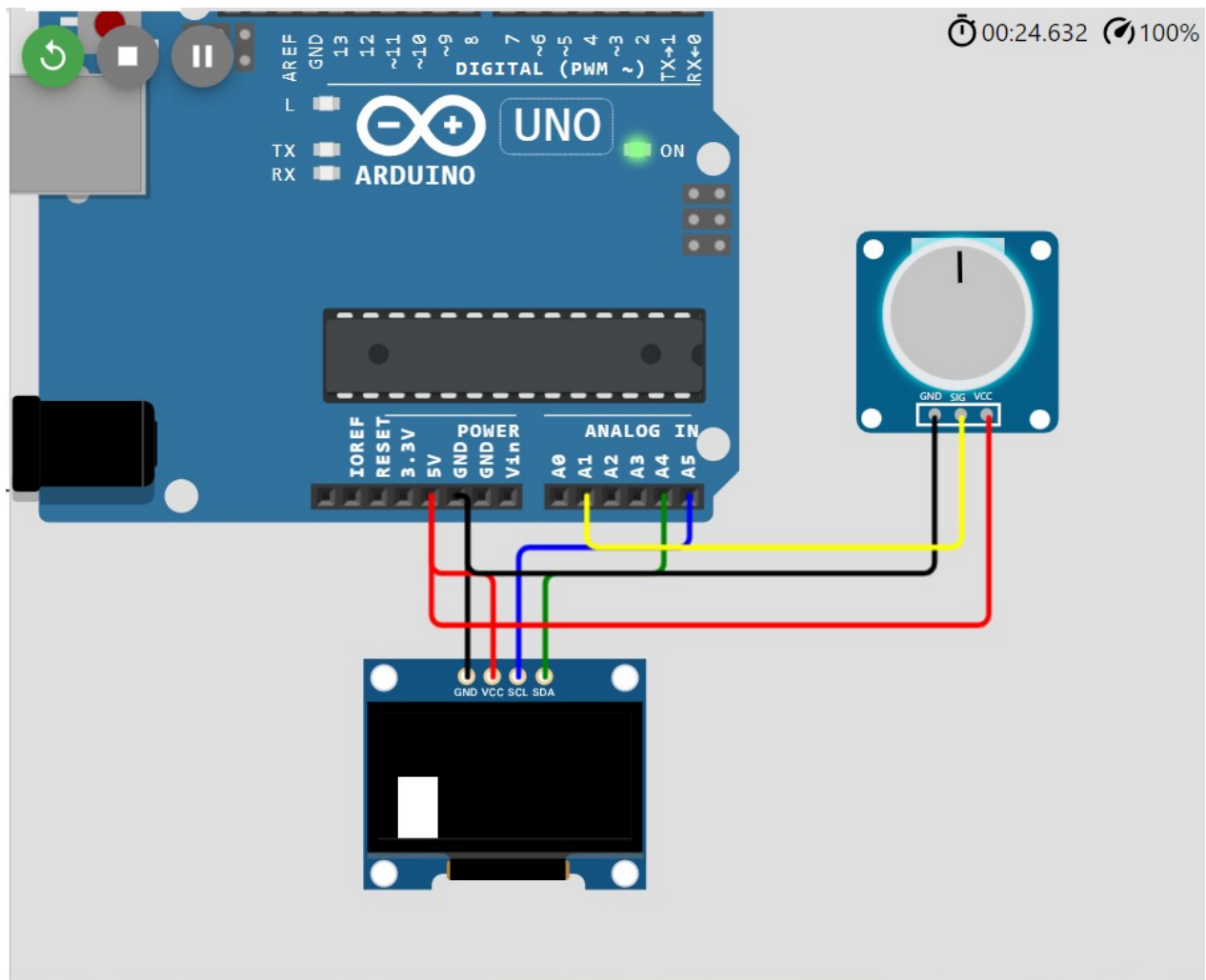


Як бачимо на екрані дисплею, усе працює, отже можна переходити до створення четвертого проекту, де на екрані дисплею прізвище відображається українськими літерами.



Тепер переходимо до п'ятого проекту, який виводить на OLED-екран прізвище та номер групи на протязі 3 сек., а потім за допомогою потенціометра змінювати значення на аналоговому виході та відображати це на OLED-екрані у вигляді графіка.





### 3. Контрольні питання

1. Що таке шина I<sup>2</sup>C? Навіщо вона потрібна? Скільки з'єднань потрібно для роботи з нею?

Шина I<sup>2</sup>C - це двопровідний протокол для зв'язку між мікроконтролерами та периферійними пристроями. Для її роботи потрібно лише два з'єднання: SDA (дані) та SCL (тактування).

2. Які дані передаються по шині I<sup>2</sup>C?

Дані, що передаються по шині I<sup>2</sup>C, включають команди та інформацію між пристроями, зокрема адреси, дані для запису та дані для читання.

3. Коли і хто запропонував протоколи I<sup>2</sup>C?

Протоколи I<sup>2</sup>C були запропоновані компанією Philips (тепер NXP Semiconductors) у 1982 році.

4. Як йде з'єднання шини I<sup>2</sup>C в Arduino?



З'єднання шини I2C в Arduino здійснюється через вбудовані пінові з'єднання SDA і SCL (наприклад, A4 і A5 на Arduino Uno).

5. Як потрібно з'єднати LCD – дисплей без шини I<sup>2</sup>C в Arduino?

LCD-дисплей без шини I2C з'єднується з Arduino через 16 піна (стандартний 4-бітний або 8-бітний інтерфейс), включаючи пін даних та контрольні піни.

6. Як під'єднати бібліотеку, якою немає серед бібліотек у «вашому» Arduino?

Для підключення бібліотеки, якої немає серед стандартних бібліотек Arduino, потрібно завантажити її з Інтернету (наприклад, з GitHub) і додати в папку "libraries" вашого проекту.

7. Як включити потрібну бібліотеку, що є у наявності, до вашого проекту?

Щоб включити потрібну бібліотеку у проект, використовуйте команду `#include <назва_бібліотеки.h>` на початку вашого скетчу.

8. Що і як потрібно вказати у програмі, щоб працювати з LCD – дисплеєм? Які саме налаштування потрібно зробити на початку скетчу?

Для роботи з LCD-дисплеєм на початку скетчу потрібно ініціювати дисплей, встановити порти та викликати функцію налаштування (наприклад, `lcd.begin(16, 2)`; для 16x2).

9. Які функції бібліотеки для роботи з дисплеєм lcd ви знаєте?

Функції бібліотеки для роботи з дисплеєм LCD включають: `lcd.begin()`, `lcd.print()`, `lcd.clear()`, `lcd.setCursor()`, `lcd.home()`, `lcd.noDisplay()`, `lcd.display()`.

## 4. Висновок

У даній лабораторній роботі я навчилась роботі з Alphanumeric I2C Liquid Crystal Display (1602 LCD) та, за допомогою мікроконтролера Arduino. В процесі виконання я створила 5 проектів з використанням OLED-дисплею та рідкокристалічного дисплею. Усі результати наведені на знімках екрану вище, код програми та посилання на сам проект

## 5. Додатки

Посилання на перший проект: <https://wokwi.com/projects/410183120424114177>

Код проекту:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Налаштування дисплея з адресою 0x27 і розміром 16x2 символів
LiquidCrystal_I2C lcd(0x27, 16, 2);

String surname = "Dyakunchak";
String name = "Ilona";

void setup() {
  lcd.begin(16,2);
  lcd.init();
  lcd.backlight();
  lcd.print(surname);
  lcd.setCursor(0,1);
  lcd.print(name);
}

void loop() {
  for(int i =0; i<10; i++){
    lcd.scrollDisplayLeft();
    delay(300);
  }
  for(int i =0; i<16; i++){
    lcd.scrollDisplayLeft();
    delay(300);
  }
}
```

Посилання на другий проект : <https://www.tinkercad.com/things/kgjQuUpERSl-lab52>

Код проекту :

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Налаштування дисплея з адресою 0x27 і розміром 16x2 символів
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```

const int temp = A1;
int val;

void setup() {
  lcd.begin(16,2);
  lcd.init();
  lcd.backlight();
}

void loop() {
  val = analogRead(temp);
  float mv = (val/1023.0)*5000-500;
  float cel = mv/10;
  float fahr = (cel*9)/5 + 32;
  lcd.setCursor(0,0);
  lcd.print(cel);
  lcd.print("*C");
  lcd.setCursor(0,1);
  lcd.print(fahr);
  lcd.print("*F");
}

```

Посилання на третій проект :

Код проекту : <https://www.tinkercad.com/things/9PiVhADHIL-lab53>

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Налаштування дисплея з адресою 0x27 і розміром 16x2 символів
LiquidCrystal_I2C lcd(0x27, 16, 2);

String surname = "Dyakunchak";

```

```

void setup() {
  lcd.begin(16,2);
  lcd.init();
  lcd.backlight();
  for(int i = 0; i < surname.length(); i++){
    if(i % 2 == 0) lcd.setCursor(i, 0);
    else lcd.setCursor(i, 1);
    lcd.print(surname[i]);
  }
}

void loop() {
}

```

Посилання на четвертий проект :

Код проекту : <https://www.tinkercad.com/things/hEZeRz96duV-lab54>

```

#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Налаштування дисплея з адресою 0x27 і розміром 16x2 символів
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Створюємо символи для літер української абетки
byte letter_D[8] = { B00111, B00101, B00101, B00101, B00101, B11111,
  B10001, B00000 };
byte letter_YA[8] = { B01111, B01001, B01001, B01111, B00011, B00101,
  B01001, B00000 };
byte letter_K[8] = { B10001, B10010, B10100, B11100, B10100, B10010,
  B10001, B00000 };
byte letter_U[8] = { B10001, B10001, B10001, B01010, B00100, B00100,
  B01100, B00000 };

```

```
byte letter_N[8] = { B10001, B10001, B10001, B11111, B10001, B10001,
B10001, B00000 };
byte letter_CH[8] = { B10001, B10001, B10001, B11111, B00001, B00001,
B00001, B00000 };
byte letter_A[8] = { B00100, B01010, B10001, B11111, B10001, B10001,
B10001, B00000 };

void setup() {
  lcd.init();      // Ініціалізація LCD
  lcd.backlight(); // Увімкнути підсвічування
  lcd.clear();     // Очищення дисплея

  lcd.createChar(0, letter_D);
  lcd.createChar(1, letter_YA);
  lcd.createChar(2, letter_K);
  lcd.createChar(3, letter_U);
  lcd.createChar(4, letter_N);
  lcd.createChar(5, letter_CH);
  lcd.createChar(6, letter_A);

  lcd.setCursor(0, 0);

  // Виводимо прізвище "Дякунчак" кирилицею на другому рядку
  lcd.setCursor(0, 1);
  lcd.write(0); // Д
  lcd.write(1); // Я
  lcd.write(2); // К
  lcd.write(3); // У
  lcd.write(4); // Н
  lcd.write(5); // Ч
  lcd.write(6); // А
  lcd.write(2); // К
}

void loop() {
```

```
}
```

Посилання на п'ятий проект :

Код проекту : <https://wokwi.com/projects/410197501895904257>

```
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Fonts/FreeSerif9pt7b.h>

#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels
const int pot = A1;
int val = 0;
// Declaration for an SSD1306 display connected to I2C (SDA, SCL pins)
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);

void setup() {
  Serial.begin(115200);

  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println("SSD1306 allocation failed");
    for(;;);
  }
  delay(2000);

  display.setFont(&FreeSerif9pt7b);
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  // Виведення прізвища та номеру групи
  display.setCursor(0, 10);
  display.println("Dyakunchak");
```

```
display.setCursor(0, 30);
display.println("Group: IP-11");
display.display();
delay(3000);
display.clearDisplay();
}

void loop() {
    val = analogRead(pot);
    display.clearDisplay();
    display.setCursor(0, 0);
    // Малюємо графік значень потенціометра
    int barHeight = map(val, 0, 1023, 0, SCREEN_HEIGHT); // Масштабування до
висоти екрану
    display.fillRect(10, SCREEN_HEIGHT - barHeight, 20, barHeight,
SSD1306_WHITE); // Малюємо стовпчик

    display.display(); // Оновлюємо дисплей

    delay(100);
}
```