

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

Лабораторна робота №6  
з дисципліни «Програмування мікропроцесорних систем»

на тему

«Програмування мікропроцесорних систем.  
Робота з 7 - сегментними індикаторами.»

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

Київ – 2024

## Зміст

Зміст.....	2
1. Постановка задачі.....	3
2. Виконання.....	3
3. Контрольні питання.....	7
4. Висновок.....	8
5. Додатки.....	8

## 1. Постановка задачі

**Мета:** навчити роботи з 7 - сегментними індикаторами за допомогою мікроконтролера Arduino, та кодів, що включають роботу з масивами двійкових чисел.

**Завдання** по **роботі.**

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.

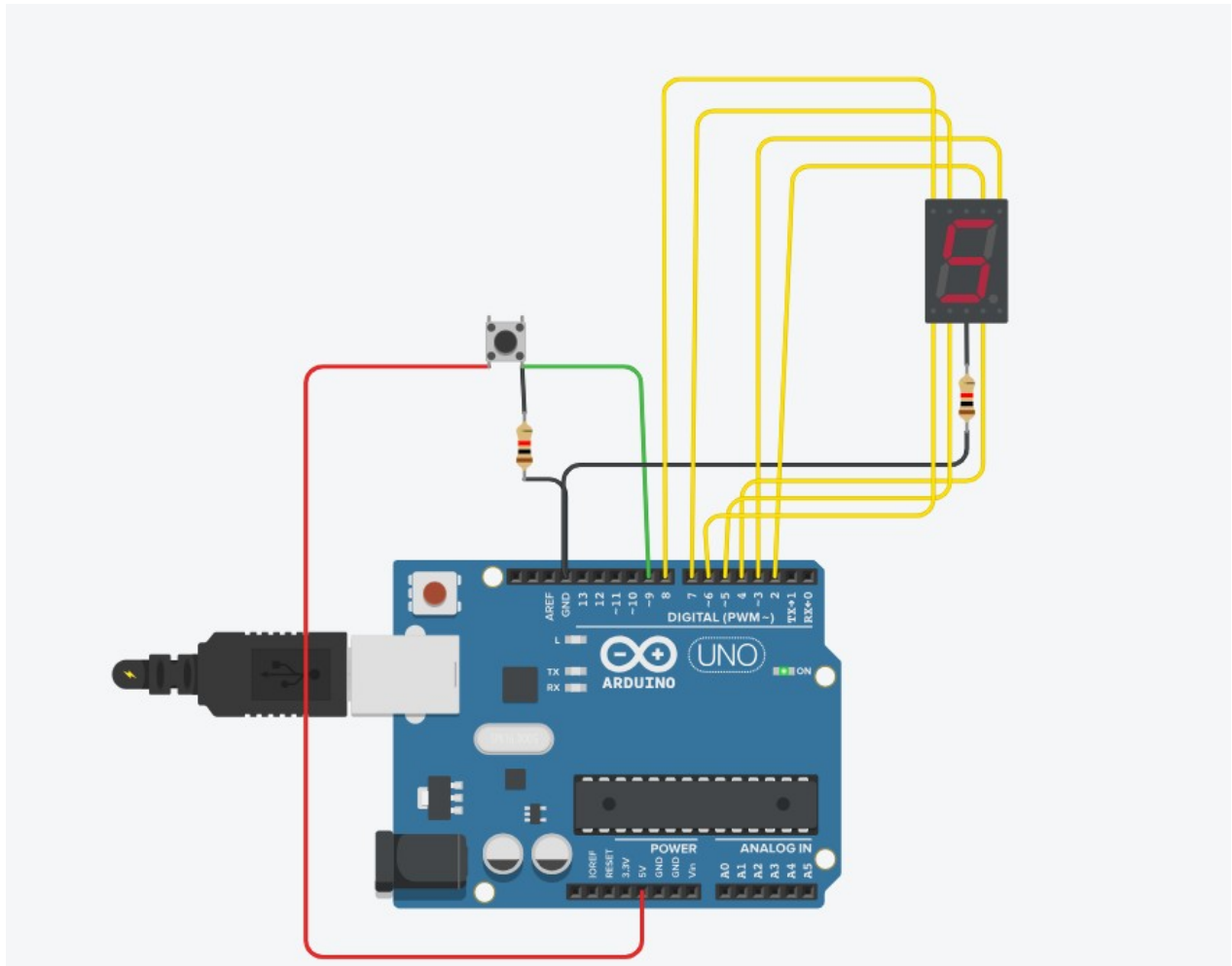
1. За допомогою сервісу tinkercad.com створити наступні проекти:

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

- 1) Скласти схему та завантажити лістинг 6.1. Впевнитись, що однорозрядний індикатор працює правильно. Розібратися з кожним рядком коду!
- 2) Написати коментар до лістингу 6.1.
- 3) Скласти схему та завантажити лістинг 6.2. Впевнитись, що чотирьохрозрядний індикатор працює правильно. Розібратися з кожним рядком коду!
- 4) Написати коментар до лістингу 6.2.
- 5) Розробити скетч з однорозрядним індикатором так, щоб при натисканні на кнопку крок за кроком виводилися цифри вашої дати народження. Наприклад, якщо дата «10.11.1998», то виводиться «1», «0», «.» і т.д.
- 6) **\*Розробити скетч з чотирьохрозрядним індикатором так, щоб справа наліво пересувалася б в нескінченному циклі дата вашого дня народження. Наприклад, число «10» у попередньому завданні.**

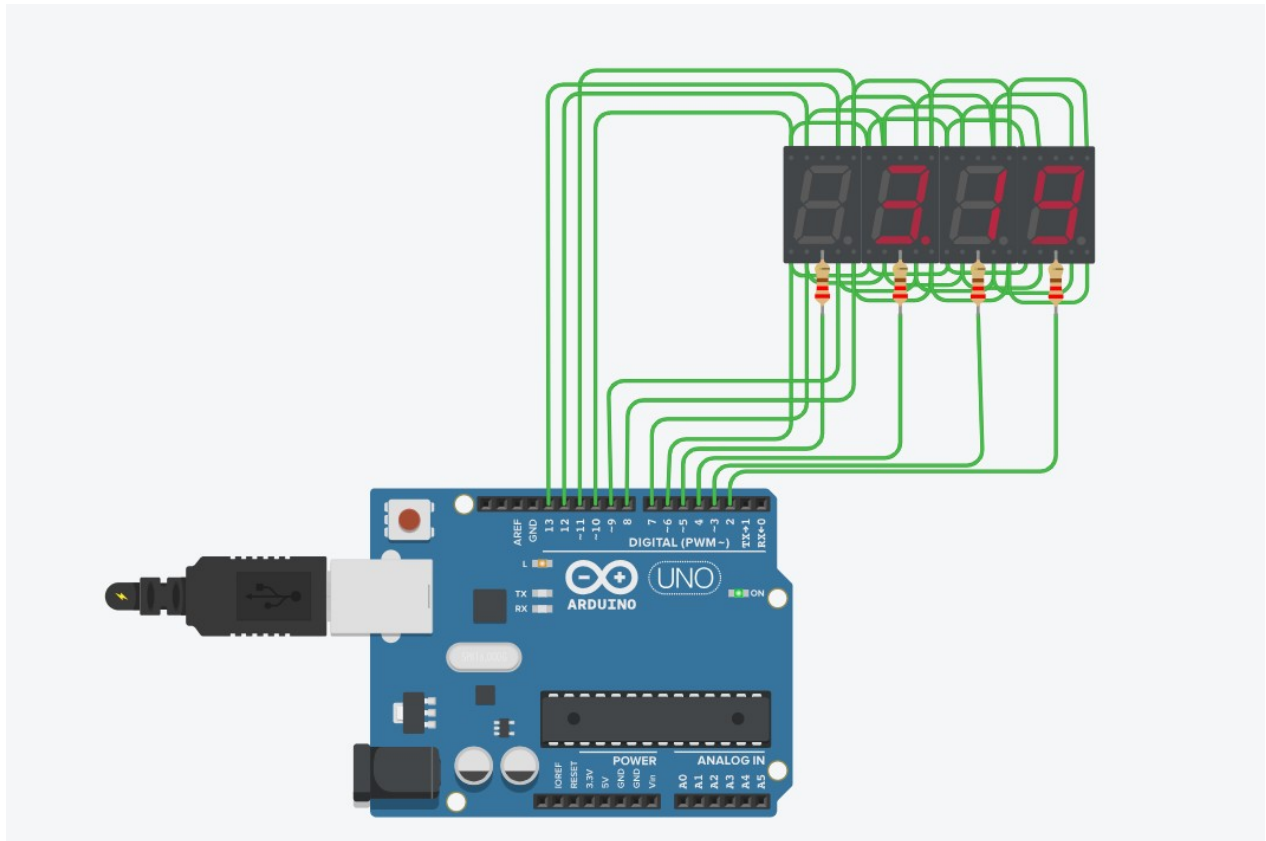
## 2. Виконання

Спочатку потрібно було скласти схему і завантажити лістинг 6.1 та впевнитись що все працює коректно, код та знімки екрану наведені нижче:

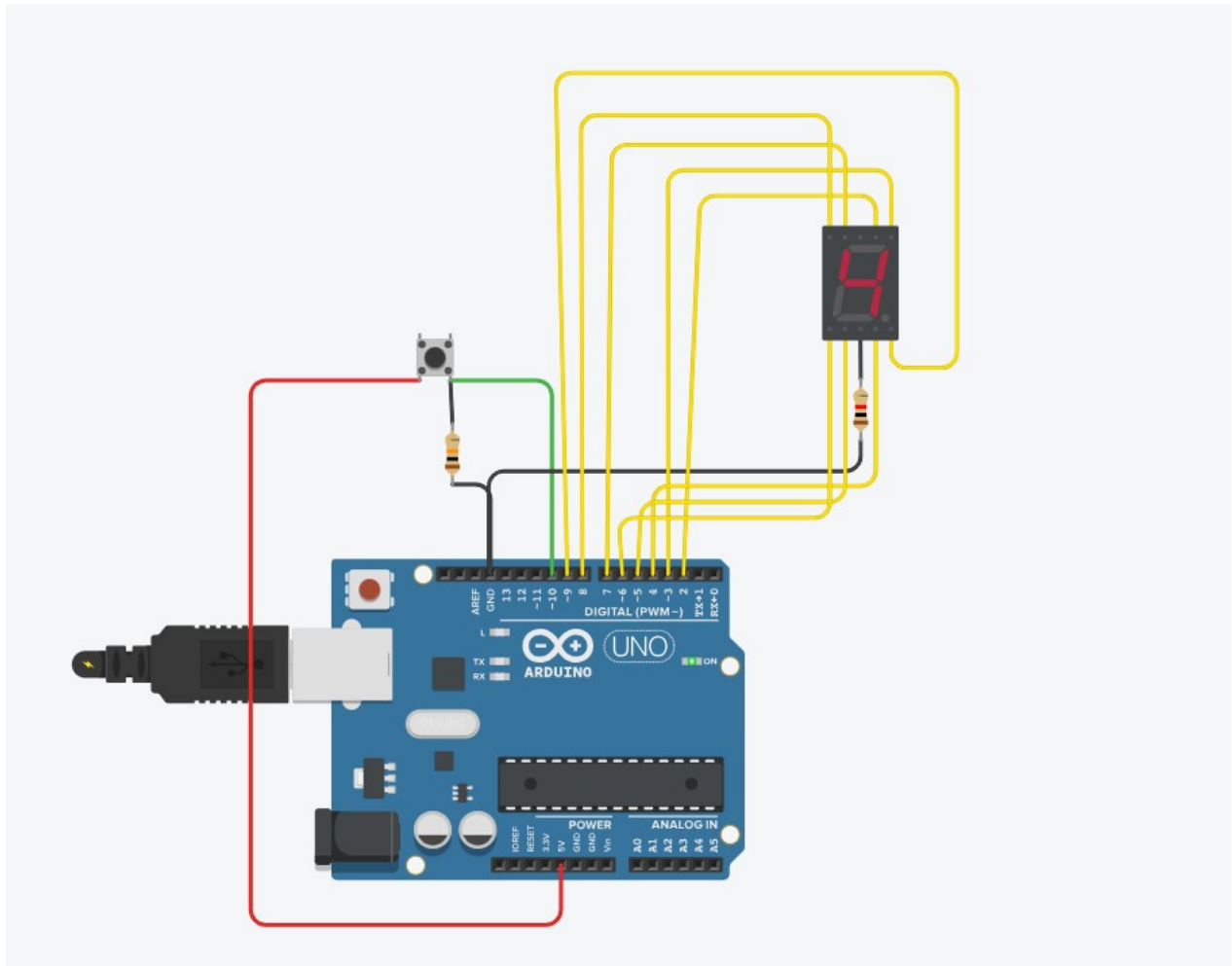


Як бачимо, усе працює, отже можна переходити до створення другого проекту та завантаження лістингу 6.2.

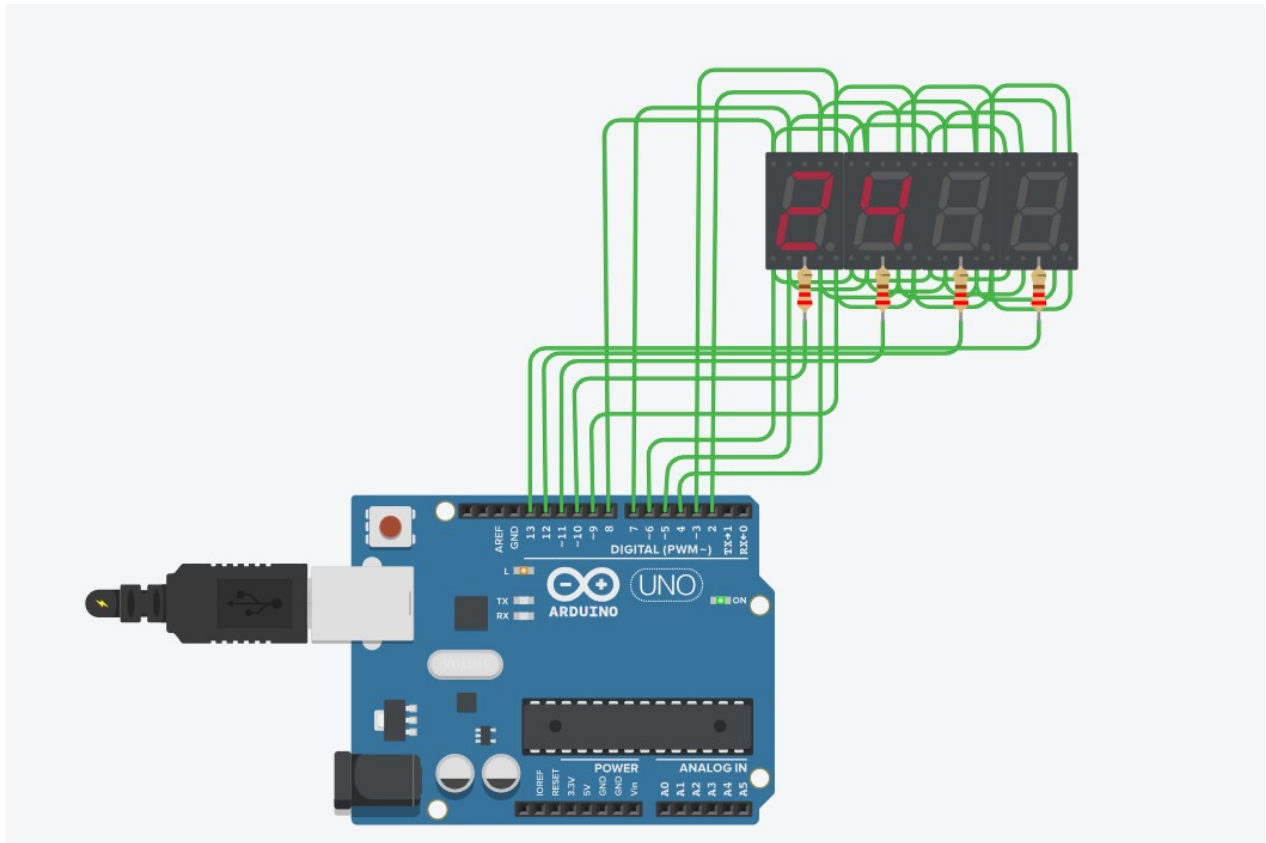
Результати роботи наведено на знімку екрану нижче, повний код програми можна знайти у додатках, також там знаходиться і посилання на проект.



Як бачимо на екрані дисплею, усе працює, отже можна переходити до створення третього проекту, в якому цифри дати народження при натисканні на кнопку будуть по чергово крок за кроком виводитися на однорозрядному індикаторі.



Як бачимо на екрані дисплею, усе працює, отже можна переходити до створення четвертого проекту, де на чотирьохрозрядному індикаторі дата мого дня народження буде пересуватися в нескінченному циклі справа наліво.



### 3. Контрольні питання

- 1) Що представляє собою однорозрядний семисегментний індикатор? Однорозрядний семисегментний індикатор — це пристрій, що складається з семи світлодіодів, які формують цифри від 0 до 9.
- 2) Як з'єднані ніжки індикатора та його сегменти? Ніжки індикатора з'єднані так, що кожна з них відповідає за активацію певного сегмента індикатора.
- 3) Що таке тип даних «byte»? Яке максимальне двійкове число можна записати за допомогою типу даних «byte»?

Тип даних «byte» — це 8-бітний тип, який може приймати значення від 0 до 255. Максимальне двійкове число, яке можна записати, — 11111111.

4) Навіщо в лістингу 6.1 використовується конструкція «mask & currentPinMask»?

У лістингу 6.1 конструкція «mask & currentPinMask» використовується для перевірки стану конкретного біта в змінній.

5) Що представляє собою чотирьохрозрядний семисегментний індикатор?

Чотирьохрозрядний семисегментний індикатор — це пристрій, що складається з чотирьох однорозрядних семисегментних індикаторів, який може відображати числа від 0 до 9999.

6) В яких рядках лістингу 6.2 видно, що один з контактів 6, 8, 9, 12 під'єднується до «землі»?

У лістингу 6.2 рядки, де видно підключення до «землі», містять вказівку на логічний нуль або заземлення, це можуть бути рядки з коментарями або визначеннями пінів.

## 4. Висновок

У даній лабораторній роботі я навчилася роботі з 7 - сегментними індикаторами за допомогою мікроконтролера Arduino, та кодів, що включають роботу з масивами двійкових чисел. Усі результати наведені на знімках екрану вище, код програми та посилання на сам проект.

## 5. Додатки

Посилання на перший проект:  
<https://www.tinkercad.com/things/4QCxT9u5VcW-lab61>

Код проекту:

```
byte numbers[10] = {  
    B11111100, B01100000, B11011010, B11110010, B01100110,  
    B10110110, B10111110, B11100000, B11111110, B11100110  
};  
  
const int BUTTON = 9; // пін 9 для кнопки  
boolean lastButton = LOW; // lastButton - для збереження попереднього стану //кнопки  
boolean currentButton = LOW; // currentButton - для збереження поточного //стану кнопки  
int counter = 0; // counter - відраховує скільки разів натиснули кнопку.
```



```

void setup()
{
    for(int i = 2; i <= 8; i++)
    {
        pinMode(i, OUTPUT);
    }
    pinMode(BUTTON, INPUT);
}

void loop()
{
    currentButton = debounce (lastButton);
    if (lastButton == LOW && currentButton == HIGH) // Якщо натискання

    {
        counter = (counter + 1) % 10;
    }
    lastButton = currentButton;
    writeNumber(counter);
}

void writeNumber(int number)
{
    if(number < 0 || number > 9)
    {
        return;
    }
    byte mask = numbers[number];
    byte currentPinMask = B10000000;
    for(int i = 2; i <= 8; i++)
    {
        if(mask & currentPinMask) digitalWrite(i,HIGH);
        else digitalWrite(i,LOW);
        currentPinMask = currentPinMask >> 1;
    }
}

boolean debounce (boolean last) // Подавлення брязкоту кнопки
{
    boolean current = digitalRead (BUTTON); // змінній current присвоюється // стан кнопки: HIGH кнопка
    натиснута, LOW кнопку не натиснуто
    if (last != current) // якщо змінився стан кнопки
    delay (5); // чекаємо 5 мс
    current = digitalRead (BUTTON); // читаємо стан кнопки після затримки
    return current; // повертаємо стан кнопки
}

```

Посилання на другий проект : <https://www.tinkercad.com/things/4J8bVUbfpvm-lab63>

Код проекту :

```
void ledDigitDisplay(float num, float time)
{
    unsigned long ltime = millis();
    int pin[] = {6, 7, 8, 9, 10, 2, 11, 3, 4, 12, 13, 5}; // піни
    int settingsSegments[] = {pin[10], pin[6], pin[3], pin[1], pin[0], pin[9], pin[4],
pin[2]}; // Порядок сегментів
    int segments[] = {0b00111111, 0b00000110, 0b01011011, 0b01001111,
0b01100110, 0b01101101,
    0b01111101, 0b00000111, 0b01111111, 0b01101111,
    0b10000000, 0b01000000}; //0, 1, 2, 3, 4, 5, 6, 7, 8, 9, '!', '-'
    // визначаємо піни як вихід
    for(int i = 0; i < 12; ++i) pinMode(pin[i], OUTPUT);
    int floatingPoint = 0, minus = 4;
    if(num > -1000 && num < 0) // працюємо з від'ємними числами
    {
        minus--;
        if(num > -100) minus--;
        if(num > -10) minus--;
        num = -num;
    }
    // працюємо з дробовими числами
    for(int i = 0; num < 1000 && minus == 4; ++i)

    {
        if(int(num * 10) != int(num)*10)
        {
            floatingPoint++;
            num *= 10;
        }
    }
    else
        break;
```

```

}
for(int i = 0, temp; millis() - ltime <= time * 1000; i++)
{
    if(i == 4) i = 0;
    temp = int(num / pow(10, i)) % 10; // Цифра, що передаємо індикатору
    if(num >= 10000 || num <= -10000 || minus == i)
// якщо мінус або переповнення, передаємо '-'
        temp = 11;

    // працюємо з 4 розрядом
    if(i == 3 && (num >= 1000 || floatingPoint == i || minus == i))
        pinMode(pin[11], OUTPUT);
    else pinMode(pin[11], INPUT);
// працюємо з 3 розрядом
    if(i == 2 && (num >= 100 || floatingPoint == i || minus == i))
        pinMode(pin[8], OUTPUT); else pinMode(pin[8], INPUT);

// працюємо з 2 розрядом
    if(i == 1 && (num >= 10 || floatingPoint == i || minus == i)) pinMode(pin[7],
OUTPUT);
    else pinMode(pin[7], INPUT);

// працюємо з 1 розрядом
    if(i == 0) pinMode(pin[5], OUTPUT); else pinMode(pin[5], INPUT);

    for(int j = 0; j < 8; j++) // Передаємо число
        if(segments[temp] & (1 << j))
            digitalWrite(settingsSegments[j], HIGH);
    if(floatingPoint && floatingPoint == i) // Передаємо точку
        digitalWrite(settingsSegments[7], HIGH);
    delay(1); // невеличка пауза, щоб світлодіоди «розігрілися»
    for(int j = 0; j < 8; j++) digitalWrite(settingsSegments[j], LOW);
// вимикаємо усі світлодіоди
}
}

```

```
void setup()
{
}

void loop()
{
  ledDigitDisplay(3.14, 2);
  ledDigitDisplay(123, 2);
  ledDigitDisplay(-5, 2);
}
```

Посилання на третій проект :

Код проекту : <https://www.tinkercad.com/things/9cXJRY2dM6E-lab64>

```
// Патерни для цифр 0-9 та крапки (.)
byte numbers[11] = {
  B11111100, // 0
  B01100000, // 1
  B11011010, // 2
  B11110010, // 3
  B01100110, // 4
  B10110110, // 5
  B10111110, // 6
  B11100000, // 7
  B11111110, // 8
  B11100110 // 9
};

const char birthDate[] = "24.01.2004";
const int dateLength = 10; // Довжина рядка дати

const int BUTTON = 10;
const int DP_PIN = 9;
boolean lastButton = LOW;
boolean currentButton = LOW;
```

```
int dateIndex = 0; // Індекс поточного символу дати

void setup() {
    // Налаштовуємо піни 2-8 як виходи для сегментів
    for(int i = 2; i <= 8; i++) {
        pinMode(i, OUTPUT);
    }
    pinMode(BUTTON, INPUT);
    pinMode(DP_PIN, OUTPUT);
}

void loop() {
    currentButton = debounce(lastButton);

    // Перевіряємо натискання кнопки
    if (lastButton == LOW && currentButton == HIGH) {
        if (dateIndex < dateLength) {
            displayDateChar(birthDate[dateIndex]);
            dateIndex++;
        } else {
            // Скидаємо індекс, щоб почати спочатку
            dateIndex = 0;
            clearDisplay();
        }
    }

    lastButton = currentButton;
}

void displayDateChar(char c) {
    // Якщо символ - цифра
    if (c >= '0' && c <= '9') {
        writeNumber(c - '0', false);
    }
    // Якщо символ - крапка
```

```

    else if (c == '.') {
        writeNumber(10, true); // Використовуємо патерн для крапки
    }
}

void writeNumber(int number, boolean showDot) {
    if (number < 0 || number > 10) {
        return;
    }

    byte mask = numbers[number];
    byte currentPinMask = B10000000;

    for(int i = 2; i <= 8; i++) {
        if(mask & currentPinMask) {
            digitalWrite(i, HIGH);
        } else {
            digitalWrite(i, LOW);
        }
        currentPinMask = currentPinMask >> 1;
    }
    // Окремо керуємо крапкою
    if (showDot) {
        digitalWrite(DP_PIN, HIGH);
    } else {
        digitalWrite(DP_PIN, LOW);
    }
}

void clearDisplay() {
    // Вимикаємо всі сегменти
    for(int i = 2; i <= 8; i++) {
        digitalWrite(i, LOW);
    }
    digitalWrite(DP_PIN, LOW);
}

```

```

}

boolean debounce(boolean last) {
    boolean current = digitalRead(BUTTON);
    if (last != current) {
        delay(5);
        current = digitalRead(BUTTON);
    }
    return current;
}

```

Посилання на четвертий проект :

Код проекту : <https://www.tinkercad.com/things/dDoi0t0i3nA-lab66>

```

int segmentsPins[] = {2, 3, 4, 5, 6, 7, 8, 9};
int digitPins[] = {10, 11, 12, 13};
// Числа для відображення на індикаторі (0 тепер повний 0)
byte numbers[10][8] = {
    {1, 1, 1, 1, 1, 1, 0, 0}, // 0
    {0, 1, 1, 0, 0, 0, 0, 0}, // 1
    {1, 1, 0, 1, 1, 0, 1, 0}, // 2
    {1, 1, 1, 1, 0, 0, 1, 0}, // 3
    {0, 1, 1, 0, 0, 1, 1, 0}, // 4
    {1, 0, 1, 1, 0, 1, 1, 0}, // 5
    {1, 0, 1, 1, 1, 1, 1, 0}, // 6
    {1, 1, 1, 0, 0, 0, 0, 0}, // 7
    {1, 1, 1, 1, 1, 1, 1, 0}, // 8
    {1, 1, 1, 1, 0, 1, 1, 0} // 9
};

int birthdayNumber[] = {0, 0, 2, 4}; // День народження: 0024

void setup() {
    for (int i = 0; i < 8; i++) {
        pinMode(segmentsPins[i], OUTPUT);
    }
}

```

```
}  
for (int i = 0; i < 4; i++) {  
    pinMode(digitPins[i], OUTPUT);  
}  
}  
  
void loop() {  
    static int offset = 0;  
  
    for (int refresh = 0; refresh < 50; refresh++) {  
        for (int digit = 0; digit < 4; digit++) {  
            int numIndex = (digit + offset) % 4;  
  
            // Вимкнення всіх розрядів  
            for (int j = 0; j < 4; j++) {  
                digitalWrite(digitPins[j], HIGH);  
            }  
  
            // Перевірка, чи потрібно взагалі щось відображати  
            if (birthdayNumber[numIndex] > 0) {  
                // Увімкнення поточного розряду  
                digitalWrite(digitPins[digit], LOW);  
  
                // Відображення цифри  
                for (int i = 0; i < 8; i++) {  
                    digitalWrite(segmentsPins[i], numbers[birthdayNumber[numIndex]][i]);  
                    //delay(5000);  
                }  
  
                // Коротка затримка для стійкості зображення  
                delay(10);  
            }  
        }  
    }  
    delay(2000);  
}
```



```
// Оновлення зсуву  
offset = (offset + 1) % 4;  
}
```