

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных систем
Факультет автоматики и вычислительной техники
Кафедра электронных вычислительных машин

Отчет по лабораторной работе №4
по дисциплине
«Информатика»
«Пьезоэлемент, мотор и текстовый дисплей»
Вариант 6

Выполнил студент гр. ИВТб-1301-05-00	_____ /Макаров С.А./
Руководитель преподаватель	_____ /Шмакова Н.А./

Киров 2025

Цель работы

Цель работы: Закрепление основ работы с Arduino.

Задание

1. Кнопочные ковбои. Начальное состояние светодиодов выключено. 3 игрока, для каждого по два светодиода. После окончания игры она начинается сначала. Выиграш игорка сигнализируется свеодиодом. игрок зарабатывает бонусы за нажатие, по нажатию на кнопку, если игрок нажал первый после звукового сигнала, то он зарабатывает бонус.

Решение

Задание 1

Выполнена живая сборка. Исходный код программы:

```
#define SHOOT_LED A5
#define PLAYER_COUNT 3
int buttonPins[PLAYER_COUNT] = {11, 12, 13};
int shootLeds[PLAYER_COUNT] = {8, 9, 10};
int scoreLeds[PLAYER_COUNT * 2] = {2, 3, 4, 5, 6, 7};
int scoreCounts[PLAYER_COUNT] = {0, 0, 0};
void setup() {
    pinMode(SHOOT_LED, OUTPUT);
    for (int i = 0; i < PLAYER_COUNT; i++) {
        pinMode(buttonPins[i], INPUT_PULLUP);
        pinMode(shootLeds[i], OUTPUT);
        pinMode(scoreLeds[i], OUTPUT);
        pinMode(scoreLeds[i + 1], OUTPUT);
        pinMode(scoreLeds[i + 2], OUTPUT);
    }
}
void loop(){
    delay(random(2000, 7000));
    while(!digitalRead(buttonPins[0]) ||
        !digitalRead(buttonPins[1]) ||
        !digitalRead(buttonPins[2])) {}
    digitalWrite(SHOOT_LED, HIGH);
    for (int player = 0; ; player = (player + 1) % PLAYER_COUNT) {
        if (!digitalRead(buttonPins[player])) {
            digitalWrite(shootLeds[player], HIGH);
            scoreCounts[player]++;
            for (int i = 0; i < scoreCounts[player]; i++)
                digitalWrite(scoreLeds[i + player * 2], HIGH);
            if (scoreCounts[player] == 2) {
                delay(4000);
                scoreCounts[0] = 0;
                scoreCounts[1] = 0;
                scoreCounts[2] = 0;
                for (int i = 0; i < PLAYER_COUNT * 2; i++)
                    digitalWrite(scoreLeds[i], LOW);
            } else {
                delay(2000);
            }
            digitalWrite(shootLeds[player], LOW);
            digitalWrite(SHOOT_LED, LOW);
            break;
        }
    }
}
```

Задание 2

Выполнена живая сборка. Исходный код программы:

Задание 3

Выполнена живая сборка. Исходный код программы:

```
#include <LiquidCrystal_I2C.h>
#define BUTTON_PIN 3
LiquidCrystal_I2C lcd(0x27, 16, 2);
String message = "hello computer science ";
volatile boolean direction = true;
int length = message.length();
int position = 0;
void setup() {
    lcd.init();
    lcd.backlight();
    pinMode(BUTTON_PIN, INPUT_PULLUP);
    attachInterrupt(1, changeDirection, FALLING);
}
void changeDirection() {
    direction = !direction;
}
void loop() {
    lcd.clear();
    lcd.setCursor(0, 0);
    String displayMessage = message.substring(
        position, min(position + 16, length));
    lcd.print(displayMessage);
    if (displayMessage.length() < 16 && length > 16)
        lcd.print(message.substring(0, 16 - displayMessage.length()));
    if (direction) {
        position++;
        if (position == length)
            position = 0;
    } else {
        position--;
        if (position == 0)
            position = length;
    }
    lcd.setCursor(0, 1);
    lcd.print(millis() / 1000);
    delay(200);
}
```

Задание 4

Выполнена живая сборка. Исходный код программы:

Задание 5

Выполнена живая сборка. Исходный код программы:

```
#include <Wire.h>
#include <TroykaLedMatrix.h>
#define SDA_PIN 5
#define SCL_PIN 4
TwoWire customWire = TwoWire(1);
TroykaLedMatrix matrix;
const uint8_t sprite_1[] {
    0b00111100,
    0b00111100,
    0b00111100,
    0b11111111,
    0b01011010,
    0b01111110,
    0b01000010,
    0b00111100,
};
const uint8_t sprite_2[] {
    0b00111100,
    0b00111100,
    0b00111100,
    0b11111111,
    0b01111010,
    0b01111110,
    0b01111000,
    0b00111100,
};
const uint8_t sprite_3[] {
    0b00111100,
    0b00111100,
    0b00111100,
    0b11111111,
    0b01111110,
    0b01111110,
    0b01111110,
    0b00111100,
};
const uint8_t sprite_4[] {
    0b00111100,
    0b00111100,
    0b00111100,
    0b11111111,
    0b01011110,
    0b01111110,
    0b00011110,
    0b00111100,
};
void setup() {
    customWire.begin(SDA_PIN, SCL_PIN, 100000);
    matrix.begin(customWire);
}
void loop() {
    matrix.clear();
    matrix.drawBitmap(sprite_1);
    delay(300);
    matrix.clear();
}
```

```

matrix.drawBitmap(sprite_2);
delay(300);
matrix.clear();
matrix.drawBitmap(sprite_3);
delay(300);
matrix.clear();
matrix.drawBitmap(sprite_4);
delay(300);
}

```

Задание 6

Выполнена живая сборка. Исходный код программы:

```

int columnPins[5] = {2, 3, 4, 5, 6};
int rowPins[7] = {7, 8, 9, 10, 11, 12, 13};
byte frame1[7] = {
    0b11111,
    0b10000,
    0b10000,
    0b11111,
    0b10001,
    0b10001,
    0b11111
};
byte frame2[7] = {
    0b10001,
    0b11011,
    0b10101,
    0b10001,
    0b10001,
    0b10001,
    0b10001
};
int lastSwitch = 0;
int interval = 1000;
boolean toggle = false;
void setup() {
    for (int i = 0; i < 5; i++) {
        pinMode(columnPins[i], OUTPUT);
        digitalWrite(columnPins[i], LOW);
    }
    for (int i = 0; i < 7; i++) {
        pinMode(rowPins[i], OUTPUT);
        digitalWrite(rowPins[i], HIGH);
    }
}
void loop() {
    unsigned long currentTime = millis();
    if (toggle) {
        displayFrame(frame1);
    } else {
        displayFrame(frame2);
    }
    if (currentTime - lastSwitch >= interval) {
        toggle = !toggle;
    }
}

```

```

        lastSwitch = currentTime;
    }
}
void displayFrame(byte frame[7]) {
    for (int row = 0; row < 7; row++) {
        digitalWrite(rowPins[row], LOW);
        for (int col = 0; col < 5; col++) {
            if (bitRead(frame[row], 4 - col)) {
                digitalWrite(columnPins[col], HIGH);
            } else {
                digitalWrite(columnPins[col], LOW);
            }
        }
    }
    for (int col = 0; col < 5; col++) {
        digitalWrite(columnPins[col], LOW);
    }
    digitalWrite(rowPins[row], HIGH);
}
}

```

```

int columnPins[5] = 2, 3, 4, 5, 6; int rowPins[7] = 7, 8, 9, 10, 11, 12, 13;
byte sprite1[] 0b01110, 0b01110, 0b11111, 0b10101, 0b11111, 0b10001,
0b01110 ;

```

```

byte sprite2[7] 0b01110, 0b01110, 0b11111, 0b11101, 0b11111, 0b11100,
0b01110 ;

```

```

byte sprite3[7] 0b01110, 0b01110, 0b11111, 0b11111, 0b11111, 0b11111,
0b01110 ;

```

```

byte sprite4[7] 0b01110, 0b01110, 0b11111, 0b10111, 0b11111, 0b00111,
0b01110 ;

```

```

int lastSwitch = 0; int interval = 500; int i = 0;
void setup() for (int i = 0; i < 5; i++) pinMode(columnPins[i], OUTPUT);
digitalWrite(columnPins[i], LOW); for (int i = 0; i < 7; i++) pinMode(rowPins[i],
OUTPUT); digitalWrite(rowPins[i], HIGH);

```

```

void loop() int currentTime = millis(); if (i == 0) displayFrame(sprite1);
else if (i == 1) displayFrame(sprite2); else if (i == 2) displayFrame(sprite3);
else if (i == 3) displayFrame(sprite4); if (currentTime - lastSwitch >= interval)
i = (i + 1) lastSwitch = currentTime;

```

```

void displayFrame(byte frame[7]) for (int row = 0; row < 7; row++)
digitalWrite(rowPins[row], LOW); for (int col = 0; col < 5; col++) if (bitRead(frame[row],
4 - col)) digitalWrite(columnPins[col], HIGH); else digitalWrite(columnPins[col],
LOW); for (int col = 0; col < 5; col++) digitalWrite(columnPins[col], LOW);

```

```
digitalWrite(rowPins[row], HIGH);
```

Вывод

В ходе лабораторной работы были изучены основы работы в среде Arduino IDE, а также собраны схемы с семисегментными индикаторами (с общим анодом и катодом), регистром 595, дешифратором 4511, драйвером 4026 в соответствии с вариантом задания.