



**«Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)»**

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ

Задание к лабораторной работе №4

по курсу «Основы микропроцессорной техники»

«Организация обмена данными через I2C и SPI»

СОДЕРЖАНИЕ

1 I2C. Подключение дисплея	3
1.1 Схема подключения	3
1.2 Необходимые файлы	3
1.3 Задание	4
1.4 Самостоятельная работа.....	4
2 SPI. Подключение дисплея	5
2.1 Схема подключения	5
2.2 Необходимые файлы	6
2.3 Задание	9
2.4 Самостоятельная работа.....	9
3 Отчет к лабораторной работе.....	9
4 Контрольные вопросы	9

1 I2C. Подключение дисплея

1.1 Схема подключения

Создайте проект на Arduino Uno на сайте <https://wokwi.com>. На макете необходимо разместить, в соответствии со схемой ниже (Рисунок 1), дисплей экран с модулем I2C (lcd1602 i2c).

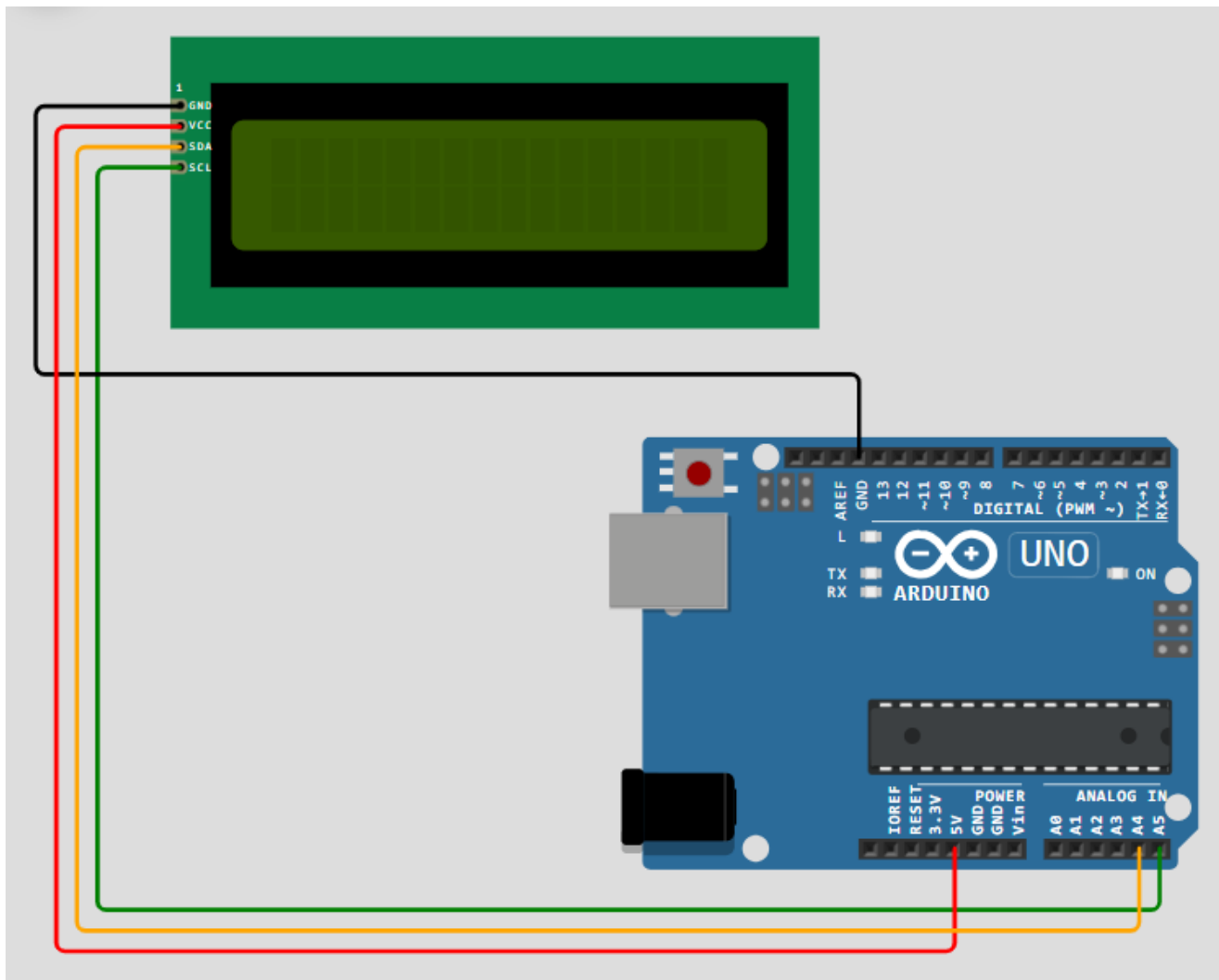


Рисунок 1 – Схема подключения

Далее необходимо повторить 2 и 3 пункты из раздела 1.2.4 Лабораторной работы №1. Также требуется установить библиотеку для дисплея, пример установки был приведен в Лабораторной работы №2.

1.2 Необходимые файлы

Добавьте в проект следующий файл и код:

main.cpp

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

LiquidCrystal_I2C lcd(0x27, 16, 2);
char message[] = "I2C one love ";
unsigned long startTime;
int counter = 0;

void setup() {
    lcd.init();
    lcd.backlight();
    startTime = millis();
}

void loop() {
    lcd.setCursor(0, 0);
    for(int i = 0; i < 16; i++) {
        lcd.print(message[(counter + i) % strlen(message)]);
    }

    lcd.setCursor(0, 1);
    switch(counter % 4) {
        case 0: lcd.print("|"); break;
        case 1: lcd.print("/"); break;
        case 2: lcd.print("-"); break;
        case 3: lcd.write(92); break;
    }

    lcd.setCursor(4, 1);
    lcd.print("Time: ");
    lcd.print((millis() - startTime) / 1000);
    lcd.print(" sec ");

    counter++;
    delay(200);
}
```

1.3 Задание

Студенту необходимо создать и собрать проект, разобраться в логике работы системы, проверить корректность работы на симуляции.

1.4 Самостоятельная работа

В макет первой части лабораторной работы студенту необходимо добавить одну кнопку. Требуется разработать систему мониторинга, считающую количество

нажатий на кнопку. Изначально на первой строке дисплея выводится сообщение/эмодзи отображающее нейтральное состояние системы, по достижении 20 нажатий эта строка должна измениться. Далее через 2 секунды счетчик нажатий должен сброситься в ноль, а сообщение в начальное состояние. На второй строке дисплея все время должно выводиться актуальное число нажатий.

2 SPI. Подключение дисплея

2.1 Схема подключения

Для выполнения данного пункта необходимо запрограммировать два микроконтроллера, поэтому создайте пару новых проектов на Arduino Nano на сайте <https://wokwi.com>. На макете для первого МК необходимо разместить кнопку, в соответствии со схемой ниже (Рисунок 2).

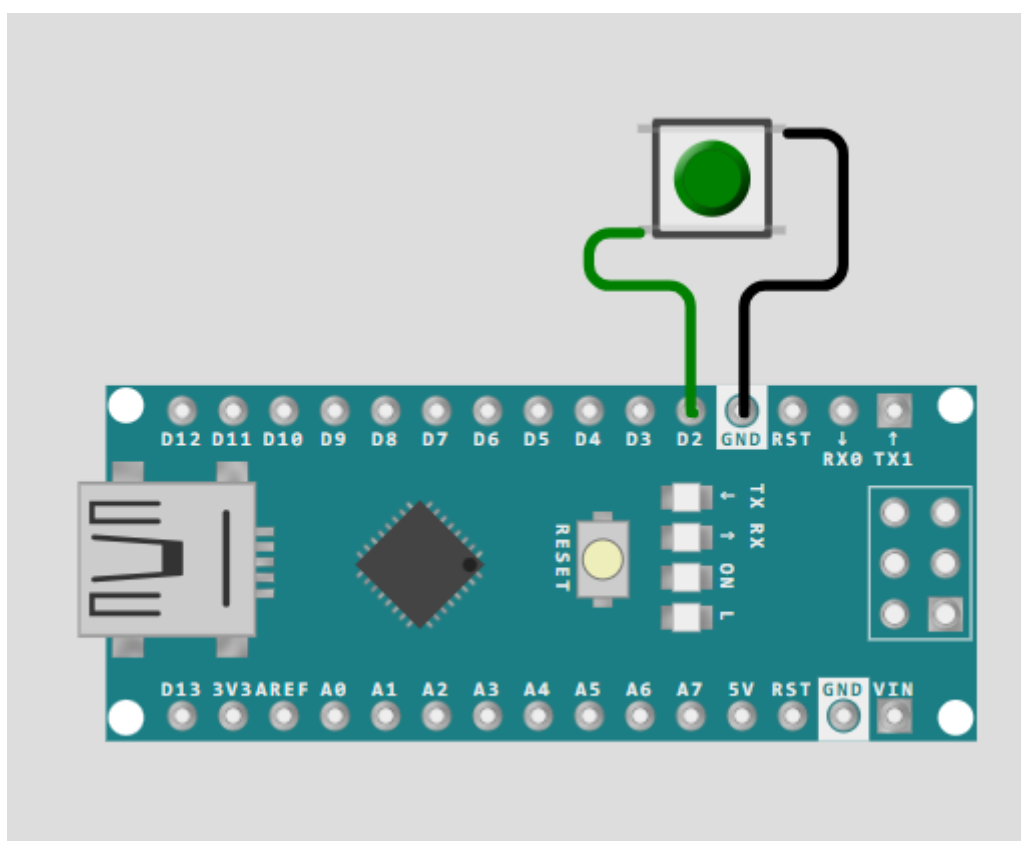


Рисунок 2 – Схема подключения мастера

Макет второго и третьего МК выглядит следующим образом (Рисунок 3).

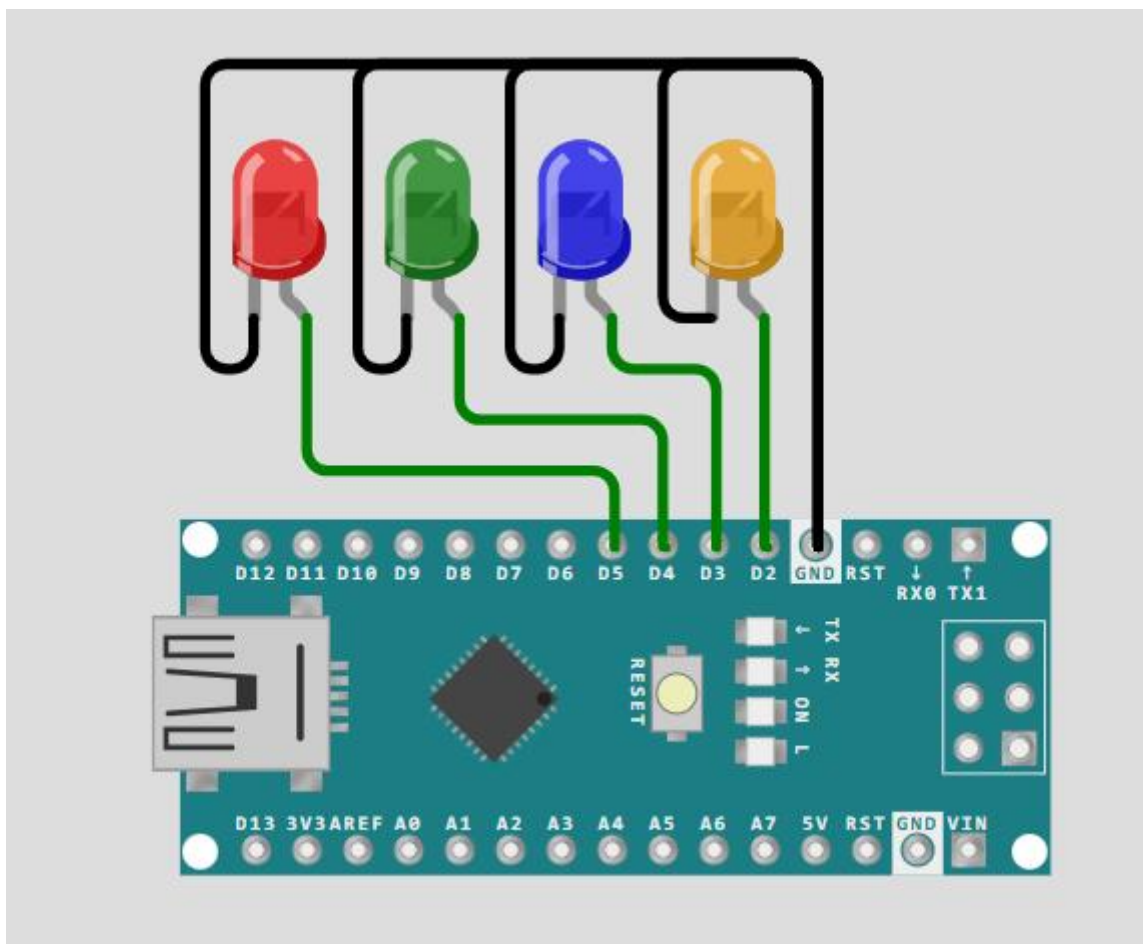


Рисунок 2 – Схема подключения слейвов

Далее в трех проектах необходимо повторить 2 и 3 пункты из раздела 1.2.4 Лабораторной работы №1.

2.2 Необходимые файлы

Добавьте в проект MASTER микроконтроллера следующий файл и код:

main.cpp

```
#include <SPI.h>
#define SS1 10
#define SS2 9
#define BUTTON_PIN 2

int activeSlave = 1;
bool lastButtonState = HIGH;
int counter = 0;

void sendToActiveSlave(byte data) {
    if (activeSlave == 1) {
        digitalWrite(SS1, LOW);
```

```

        SPI.transfer(data);
        digitalWrite(SS1, HIGH);
    } else {
        digitalWrite(SS2, LOW);
        SPI.transfer(data);
        digitalWrite(SS2, HIGH);
    }
    Serial.print("Sent to Slave");
    Serial.print(activeSlave);
    Serial.print(": ");
    Serial.println(data);
}

void setup() {
    pinMode(SS1, OUTPUT);
    pinMode(SS2, OUTPUT);
    pinMode(BUTTON_PIN, INPUT_PULLUP);
    digitalWrite(SS1, HIGH);
    digitalWrite(SS2, HIGH);
    SPI.begin();
    Serial.begin(115200);
    Serial.println("Press button to switch slaves");
}

void loop() {
    if (digitalRead(BUTTON_PIN) == LOW && lastButtonState == HIGH) {
        activeSlave = (activeSlave == 1) ? 2 : 1;
        Serial.print("Active Slave: ");
        Serial.println(activeSlave);
        delay(300);
    }
    lastButtonState = digitalRead(BUTTON_PIN);

    if (millis() % 1000 == 0) {
        sendToActiveSlave(counter);
        counter++;
    }
}

```

Добавьте в проект SLAVE 1 микроконтроллера следующий файл и код:

main.cpp

```

#include <SPI.h>
#define SS_PIN 10
int ledPins[] = {2,3,4,5};

void setup() {
    for(int i=0; i<4; i++) pinMode(ledPins[i], OUTPUT);
    pinMode(SS_PIN, INPUT);
}

```

```

    SPCR |= _BV(SPE);
    SPI.attachInterrupt();
}

ISR (SPI_STC_vect) {
    byte data = SPDR;

    if (data % 2 == 0) {
        for(int i = 0; i < 4; i++) {
            digitalWrite(ledPins[3 - i], HIGH);
            delay(200);
            digitalWrite(ledPins[3 - i], LOW);
        }
    } else {
        for(int i = 0; i < 4; i++) {
            digitalWrite(ledPins[i], HIGH);
        }
    }
}

void loop() {}

```

Добавьте в проект SLAVE 2 микроконтроллера следующий файл и код:

main.cpp

```

#include <SPI.h>
#define SS_PIN 9
int ledPins[] = {2,3,4,5};

void setup() {
    for(int i=0; i<4; i++) pinMode(ledPins[i], OUTPUT);
    pinMode(SS_PIN, INPUT);
    SPCR |= _BV(SPE);
    SPI.attachInterrupt();
}

ISR (SPI_STC_vect) {
    byte data = SPDR;

    if (data % 2 == 1) {
        for(int i = 0; i < 4; i++) {
            digitalWrite(ledPins[i], HIGH);
            delay(200);
            digitalWrite(ledPins[i], LOW);
        }
    } else {
        for(int i = 0; i < 4; i++) {
            digitalWrite(ledPins[i], LOW);
        }
    }
}

```



```
}  
}  
  
void loop() {}
```

2.3 Задание

Студенту необходимо создать и собрать три проекта, разобраться в логике работы каждого из них. Далее требуется прошить физическое устройство и проверить корректность работы системы.

2.4 Самостоятельная работа

Создание игровой системы "Угадай число" на трех Arduino с использованием SPI.

Создайте систему из трех Arduino Nano, где один ведущий (Master) и два ведомых (Slave) взаимодействуют через SPI. Master загадывает случайное число от 1 до 100. Два Slave пытаются его угадать - один перебирает числа снизу вверх, другой сверху вниз. При каждом обращении Master отправляет загаданное число Slave, а они возвращают свои текущие догадки. Когда один из Slave угадывает число, Master останавливает игру и объявляет победителя через Serial Monitor.

3 Отчет к лабораторной работе

Составьте отчет к лабораторной работе.

Требования:

1. Описание лабораторной работы и основных этапов её выполнения, с соответствующими скриншотами.
2. Выполнение самостоятельного задания со скриншотами и результатами измерений.
3. Ответы на вопросы.
4. Выводы к лабораторной работе.

4 Контрольные вопросы

1. В чем разница между lcd.print и lcd.write?

2. Почему вместо \ на дисплей экране выводится ¥?
3. Что такое SPCR и что именно выполняет команда `SPCR |= _BV(SPE)`?
4. Что выполняет команда `SPI.attachInterrupt()`?
5. Что такое SPDR и когда вызывается ISR (`SPISTCvect`)?
6. Почему в файле `platformio.ini` мы не вносили настройки скорости, как это происходило в Лабораторной работе №3?