

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №14

РАБОТА ЗАЩИЩЕНА С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

старший преподаватель  
должность, уч. степень, звание

подпись, дата

Сыщиков А. Ю.  
инициалы, фамилия

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

по дисциплине: Интерфейсы автоматизированных систем обработки  
информации и управления

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР.

1742

подпись, дата

Коробков Д.В.

Седов В.А.

инициалы, фамилия

Санкт-Петербург 2020

## 1. Цель работы:

Передача данных между устройствами с помощью UART-соединения

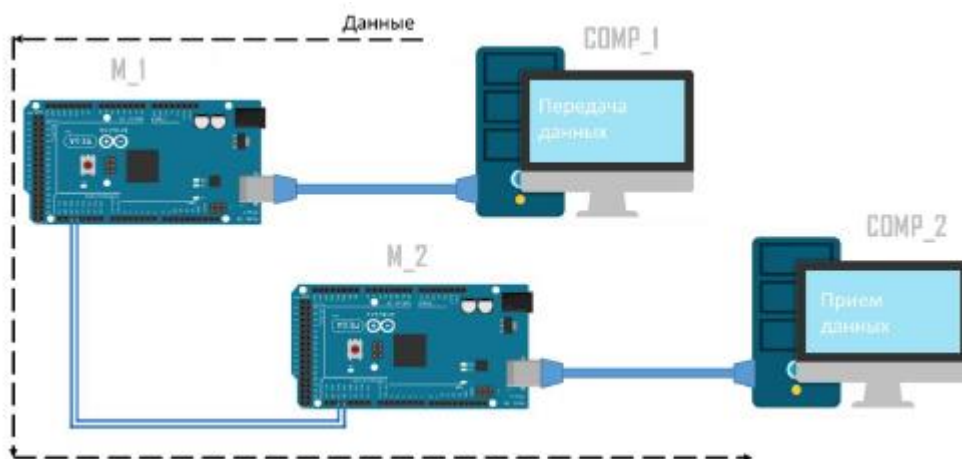
## 2. Практическая часть:

Реализовать дуплексное соединение устройств М\_1 и М\_2 по UART-соединению, моделирующего работу «чата». Устройство М\_1 получает данные от компьютера COMP\_1 и передает на устройство М\_2. М\_2, в свою очередь, принимает эти данные и передает на компьютер COMP\_2. То же самое происходит и в обратную сторону. Таким образом, происходит общение двух пользователей. При этом, необходимо синхронизировать устройства таким образом, чтобы оба пользователя получали данные с одним и тем же «штампом» времени, а также предусмотреть возможность «горячего» подключения одного из устройств, т.е. одно устройство уже работает и принимает данные от COMP\_1, в то время как второе устройство «не в сети». При подключении второго устройства к первому необходимо передать накопленные сообщения (максимально – 10 сообщений по 250 символов в каждом). Выбор библиотеки для передачи и приема данных – на усмотрение разработчика (*SoftwareSerial* или *HardwareSerial*).

Используемые устройства:

- Плата Arduino UNO x2;
- Соединительные провода (папа-папа) x4;

Схема подключения устройств:



## 3. Программы для взаимодействующих устройств:

### а. Программа для «ведущего» устройства

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(6, 7); // RX, TX
```

```

int flag_online=1;
char strings[10][15];
void setup() {
  Serial.begin(9600);
  Serial.println("ARD1: I'm ready.");
  mySerial.begin(4800);
  pinMode(9,OUTPUT);
  digitalWrite(9, 1);
  if (digitalRead(8)==0){
    flag_online=0;
    Serial.println("ARD2 офлайн."); //второй контролдер отключен
  }
  else{
    flag_online=1;
    Serial.println("ARD2 онлайн."); //второй контролдер отключен
  }
}
int flag2=0;
int j =0;
int i=0;
int flag3 =0;

void loop() {
  if ((digitalRead(8)==0)&&(flag_online==1)){
    Serial.println("ARD2 вышел из чата."); //второй контролдер отключен
    flag_online=0;
    delay(200);
  }
  if ((digitalRead(8)==1)&&(flag_online==0)){
    Serial.println("ARD2 вернулся в чат."); //второй контролдер отключен
    flag_online=1;
    delay(200);
  }

  if (Serial.available()){
    if ((flag_online==1)&&(flag3==0)){
      //Serial.println("Положительно. Передача."); //второй контролдер отключен
      //for (int j = 0; j < Serial.available(); j++){
        //Serial.write("первый"); //второй контролдер отключен
        //mySerial.write(buffer);
        int buffer2 = Serial.read();
        mySerial.write(buffer2);
        Serial.write(buffer2);
      }
    }
  }
}

```

```

    flag2=0;
    if (!(Serial.available()>0)){
        delay(100);
    }
    //}
}
if (flag_online==0){
    //Serial.println("Отрицательно. Сохранение в буфер.");
    //Serial.println("Заполняем буфер"); //второй контроллер отключен
    //for (int j = 0; j < Serial.available(); j++){
    // int buffer = Serial.read();
    // strings[i][j]=buffer;
    //}
    strings[i][j]=Serial.read();
    //Serial.write(strings[0]);
    j++;
    if (!(Serial.available()>0)){
        flag2=1;
        delay(1000);
    }
}
}
else{
    delay(100);
    if (flag2==1) {
        Serial.print(strings[i]);
        Serial.print("Сохранено в буфер.");
        Serial.println();
        flag3=1;
        i++;
        j=0;
        flag2=0;
    }
    if ((flag3==1) &&(flag_online==1)){
        /*Serial.write("Выводим буфер");
        Serial.write(strings[0][0]);
        Serial.write(strings[0][1]);
        Serial.write(strings[0][2]);
        Serial.write(strings[0][3]);
        Serial.write(strings[0][4]);
        Serial.write(strings[0][5]);*/
        /*for (j = 0; j<i; ++j){
        for (int jj = 0; jj<15; ++jj){

```

```
//Serial.write(" Что в буфер");
Serial.write(strings[0][jj]);
// mySerial.write(strings[0][jj])
}
// }*/
flag3=0;
for (int jj =0; jj<i; jj++){
    //Serial.println(i);
    //Serial.println("Отправка отложенных сообщений.");
    //Serial.println(strings[jj]);
    mySerial.write(strings[jj]);
    delay(100);
}
i=0;
j=0;
}
}

if (mySerial.available()){
    Serial.write(mySerial.read());
}
}
```

## b. Программа для «ведомого» устройства

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(6, 7); // RX, TX
int flag_online=1;
char strings[10][15];
void setup()
{
  Serial.begin(9600);
  Serial.println("ARD2: I'm ready.");
  mySerial.begin(4800);
  pinMode(8,OUTPUT);
  digitalWrite(8, 1);
  if (digitalRead(9)==0){
    flag_online=0;
    Serial.println("ARD1 офлайн."); //второй контролдер отключен
  }
  else{
    flag_online=1;
    Serial.println("ARD1 онлайн."); //второй контролдер отключен
  }
}

int flag2=0;
int j =0;
int i=0;
int flag3 =0;

void loop() {
  if ((digitalRead(9)==0)&&(flag_online==1)){
    Serial.println("ARD1 вышел из чата."); //второй контролдер отключен
    flag_online=0;
    delay(200);
  }
  if ((digitalRead(9)==1)&&(flag_online==0)){
    Serial.println("ARD1 вернулся в чат."); //второй контролдер отключен
    flag_online=1;
    delay(200);
  }

  if (Serial.available()){
    if ((flag_online==1)&&(flag3==0)){
      //Serial.println("Положительно. Передача."); //второй контролдер отключен
```

```

//for (int j = 0; j < Serial.available(); j++){
//Serial.write("первый"); //второй контролдер отключен
//mySerial.write(buffer);
int buffer2 = Serial.read();
mySerial.write(buffer2);
Serial.write(buffer2);
flag2=0;
if (!(Serial.available()>0)){
    delay(100);
}
//}
}
if (flag_online==0){
//Serial.println("Отрицательно. Сохранение в буфер.");
//Serial.println("Заполняем буфер"); //второй контролдер отключен
//for (int j = 0; j < Serial.available(); j++){
// int buffer = Serial.read();
// strings[i][j]=buffer;
//}
strings[i][j]=Serial.read();
//Serial.write(strings[0]);
j++;
if (!(Serial.available()>0)){
    flag2=1;
    delay(1000);
}
}
}
else{
    delay(100);
    if (flag2==1) {
        Serial.print(strings[i]);
        Serial.print("Сохранено в буфер.");
        Serial.println();
        flag3=1;
        i++;
        j=0;
        flag2=0;
    }
    if ((flag3==1) &&(flag_online==1)){
        /*Serial.write("Выводим буфер");
        Serial.write(strings[0][0]);
        Serial.write(strings[0][1]);

```

```

Serial.write(strings[0][2]);
Serial.write(strings[0][3]);
Serial.write(strings[0][4]);
Serial.write(strings[0][5]);*/
/*for (j = 0; j<i; ++j){
for (int jj = 0; jj<15; ++jj){
    //Serial.write(" Что в буфер");
    Serial.write(strings[0][jj]);
    // mySerial.write(strings[0][jj])
}
// }*/
flag3=0;
for (int jj =0; jj<i; jj++){
    //Serial.println(i);
    //Serial.println("Отправка отложенных сообщений.");
    //Serial.println(strings[jj]);
    mySerial.write(strings[jj]);
    delay(100);
}
i=0;
j=0;
}
}

if (mySerial.available()){
    Serial.write(mySerial.read());
}
}

```

#### 4. Демонстрация работы в роликах:

Демонстрация работы у Вадима: <https://www.youtube.com/watch?v=vZ3ZAbEsDQM>

Демонстрация работы у Данилы: <https://yadi.sk/d/PxHGuY1hdgNa-Q>