

## **Подключение удаленного COM контроллера к USB порту компьютера**

**Цель работы:** Изучить вопрос подключения удаленного устройства к USB-COM порту компьютера

**Задача работы:** Построение канала связи COM controller – USB компьютер.

**Приборы и принадлежности:** Персональный компьютер, USAВ/COM преобразователь, RS-232/UART преобразователь, COM контроллер, утилита COM порта.

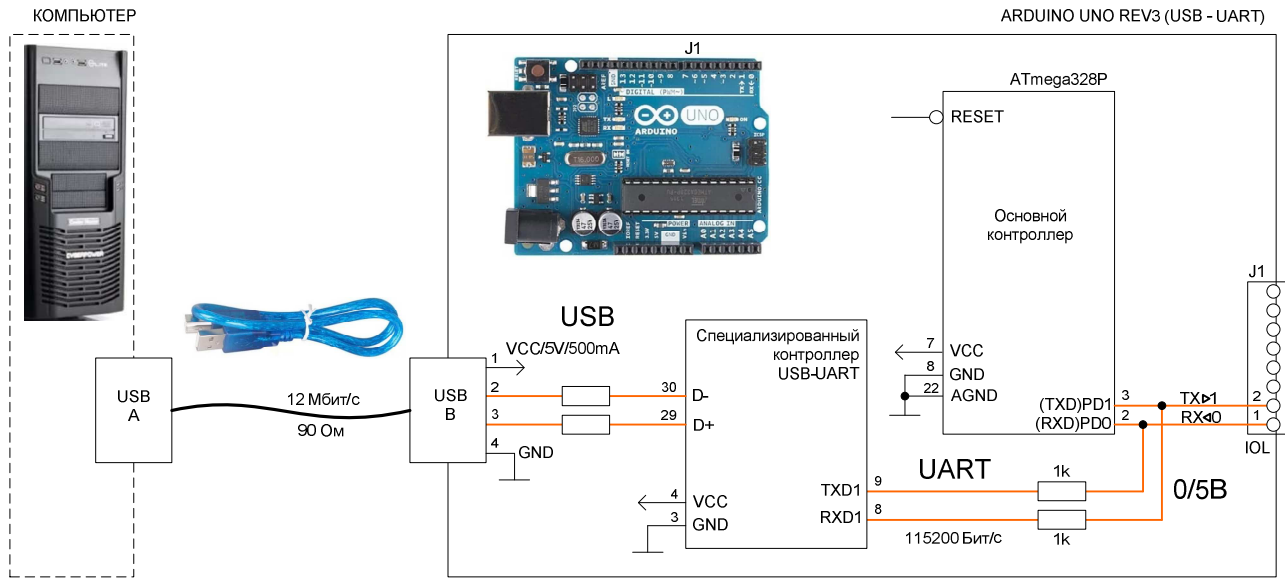
### **Введение**

До сих пор существует множество устройств, которые подключаются к компьютеру через COM порт, но, поскольку в состав современных компьютеров все реже входят COM порты, то связь с COM устройствами выполняется через USB порты дополненными специальными преобразователями сигналов. Значительное повышение частоты сигналов в USB линии (в сравнении с сигналами COM порта) накладывает ограничение на длину линий, повышает их стоимость и требует решения вопросов согласования линий. В этой работе на примере контроллера Arduino UNO рассматривается подключение COM устройства к компьютеру через протяженные несогласованные линии.

### **ОБЩИЕ СВЕДЕНИЯ**

#### **Структура канала USB – плата Arduino UNO**

Типовая схема подключения контроллера Arduino UNO к компьютеру через USB порт показана на Рисунок 1. Со стороны компьютера канал связи виден как стандартный COM порт. Но, на самом деле, это виртуальный COM порт с которым компьютер обменивается пакетами данных на частоте 12 МГц, а специализированный контроллер, расположенный на плате Arduino UNO, преобразует пакеты USB данных в последовательность бит в формате асинхронного интерфейса UART с уровнями 0/5В, которые и используются основным контроллером Arduino UNO (микросхема АТmega328Р) для загрузки программ и обмена данными с компьютером в процессе выполнения программ.

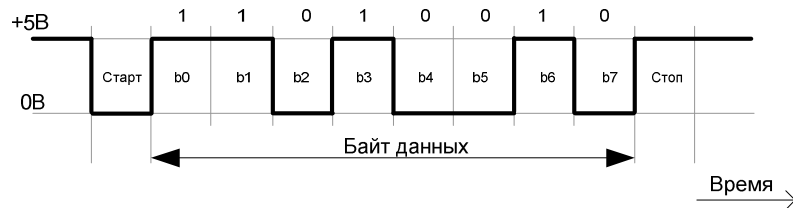


**Рисунок 1.** Типовое подключение контроллера Arduino UNO к компьютеру через USB порт.

Временная диаграмма последовательной передачи данных по правилам UART устройства с уровнем сигналов 0/5В показана на Рисунок 2. Данные передаются байтами. Помимо данных последовательность содержит стартовый и стоповый биты и может включать другие служебные биты, например, бит контроля четности, применение которого задается в настройках COM порта, там же устанавливается и одна из стандартных скоростей передачи.

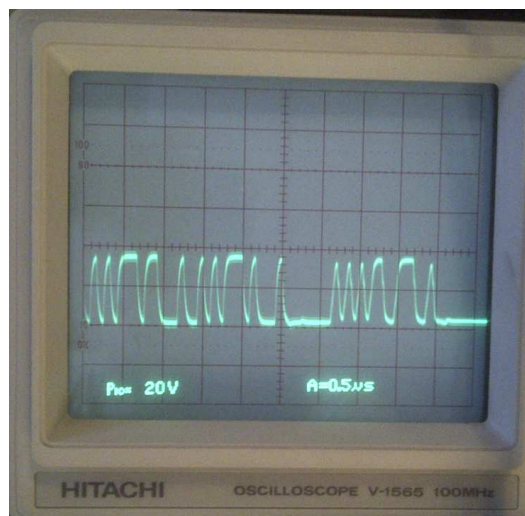
**Примечание.** В семействе асинхронного интерфейса UART наиболее известен стандарт физического уровня RS-232, применяемый COM-портом компьютера.

COM порт не имеет сигналов синхронизации, временные интервалы формируются как передатчиком так и приемником с точностью тактирования не хуже 5%.



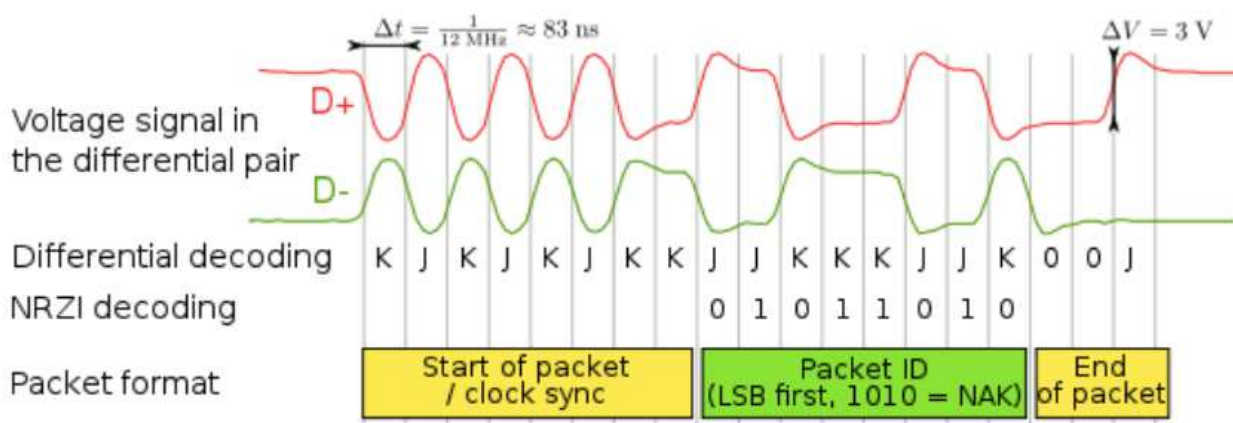
**Рисунок 2.** Временная диаграмма UART последовательной передачи данных (01001011<sub>2</sub>) микросхемы ATmega328P контроллера Arduino UNO.

Контроллер Arduino UNO содержит специализированный контроллер для преобразования UART сигналов в USB последовательность и наоборот. Порт USB компьютера осуществляющий связь с виртуальным COM портом работает в режиме *Full-speed* на частоте 12 Мбит/с (Рисунок 3). Этот режим поддерживает как USB 1.0. так и USB 2.0.



**Рисунок 3.** Измеренный 4В сигнал на дифференциальной линии USB–COM контроллера Arduino. Длина USB кабеля 2м. Частота сигналов на USB линии 12 МГц. Для формирования сигналов использовалась запись данных в COM порт контроллера. Частота USB данных 12 МГц не изменялась при записи в COM порт как на скорости 9600 бит/с так и 115200 бит/с.

Данные по шине USB передаются пакетами (Рисунок 4). Размеры пакета зависят от типа выполняемой передачи. Каждый пакет в режиме *Full-speed* содержит 8 бит синхронизации тактов приемника и передатчика (Sync), 8 бит идентификатора пакета (PID) и 2 бита конца пакета (EOP). Блок данных может составлять от 0 до 1023 байт.



**Рисунок 4.** Пример передачи пакета по дифференциальной линии USB 1.1 в режиме *Full-speed* [2]. Изменение состояние дифференциального сигнала соответствует передаче нуля, сохранение уровней - соответствует передаче единицы. Для улучшения синхронизации на единичных последовательностях принудительно вставляют нуль на каждые 6 единиц подряд.

Кроме пакета данных передаются и другие пакеты. Для выполнения всех передач по USB требуется, чтобы 2 или 3 пакета информации были переданы между хост-контроллером и приемником. Если передача оказалась успешной, пункт назначения возвращает пакет квитирования. При обнаружении ошибки во время передачи генерируется пакет отсутствия уведомления.

Дифференциальные сигналы USB передаются по витой паре экранированного 4-проводного кабеля. По стандарту, сечение сигнальных проводников высокоскоростного кабеля USB 2.0 должно быть 28 AWG и от 20 до 28 AWG для жил питания, в зависимости от длины кабеля (см. Таблица 1).

**Таблица 1.** Примерное соответствие длины и диаметра проводов USB2 кабеля.

Размер провода [3]		Сопротивление медной жилы, мОм/м	Длина кабеля, см
Номер AWG	Диаметр, мм		
28	0,321	212,9	81
26	0,405	133,9	131
24	0,511	84,22	208
22	0,644	52,96	333
20	0,812	33,31	500

Для увеличения длины USB кабеля его снабжают встроенными усилителями сигнала.

По требованию спецификации USB 2.0 для режима *High-speed* (до 480 Мбит/с) задержка распространения сигнала в кабеле не должна превышать 5,2 нс/м и быть не более 26 нс, что и определяет максимальную длину кабеля 5 м.

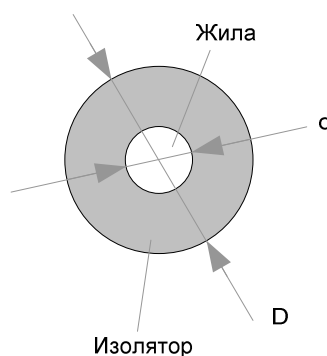
Задержка на метр длины в коаксиальном кабеле обратно пропорциональна скорости распространения волны в м/с, которая вычисляется как

$$v = \frac{c}{\sqrt{\epsilon\mu}},$$

где  $c$  – скорость света  $3 \cdot 10^8$  м/с;  $\epsilon$  - диэлектрическая проницаемость материала внутреннего изолятора;  $\mu$  - магнитная проницаемость изолятора. Для полиэтилена  $\mu = 1$  и  $\epsilon = 2,2$  фазовая скорость равна  $2 \cdot 10^8$  м/с и, соответственно, задержка 5 нс/м.

Для уменьшения потерь сигнала важно обеспечить однородность волнового сопротивления (**в.с.**) сигнальной линии. Изменение в.с. может быть связано с некачественной заделкой кабеля, плохим согласованием элементов линии, низким качеством разъёма и др.

Волновое сопротивление кабеля определяется его конструкцией. В.с. коаксиального кабеля в области высоких частот (30 кГц и выше) вычисляется по следующей формуле.



$$Z = \sqrt{\frac{L}{C}} = \frac{60}{\sqrt{\epsilon}} \ln\left(\frac{D}{d}\right), \text{ Ом,}$$

где  $L$  – продольная индуктивность закороченного кабеля, Гн;  $C$  – поперечная ёмкость разомкнутого кабеля, Ф;  $\epsilon$  - диэлектрическая проницаемость изолятора;  $D$  - диаметр изолятора;  $d$  – диаметр проводника. Величина в.с. не зависит от длины кабеля.

Диэлектрическая проницаемость  $\epsilon$  изоляторов лежит в диапазоне 1 .. 7: 1 – воздух, вакуум; 1.3 .. 2.4 – полиэтилен; 2.5..6 - резина; 5..7 – фарфор; 6..7 – слюда; 7 - стекло.

Величина в.с. витой пары USB 2.0 кабеля составляет 90 +/- 15% Ом [5]. Расчет в.с. экранированной витой пары должен учитывать и взаимное расположение проводников.

В согласованном кабеле, у которого нагрузка по концам имеет сопротивление, равное в.с., вся передаваемая электромагнитная энергия полностью поглощается приемником без

отражения. В неоднородных линиях и при несогласованных нагрузках в местах электрической несогласованности возникают отраженные волны и часть энергии возвращается к началу линии.

Коэффициент отражения волн в кабеле равен отношению

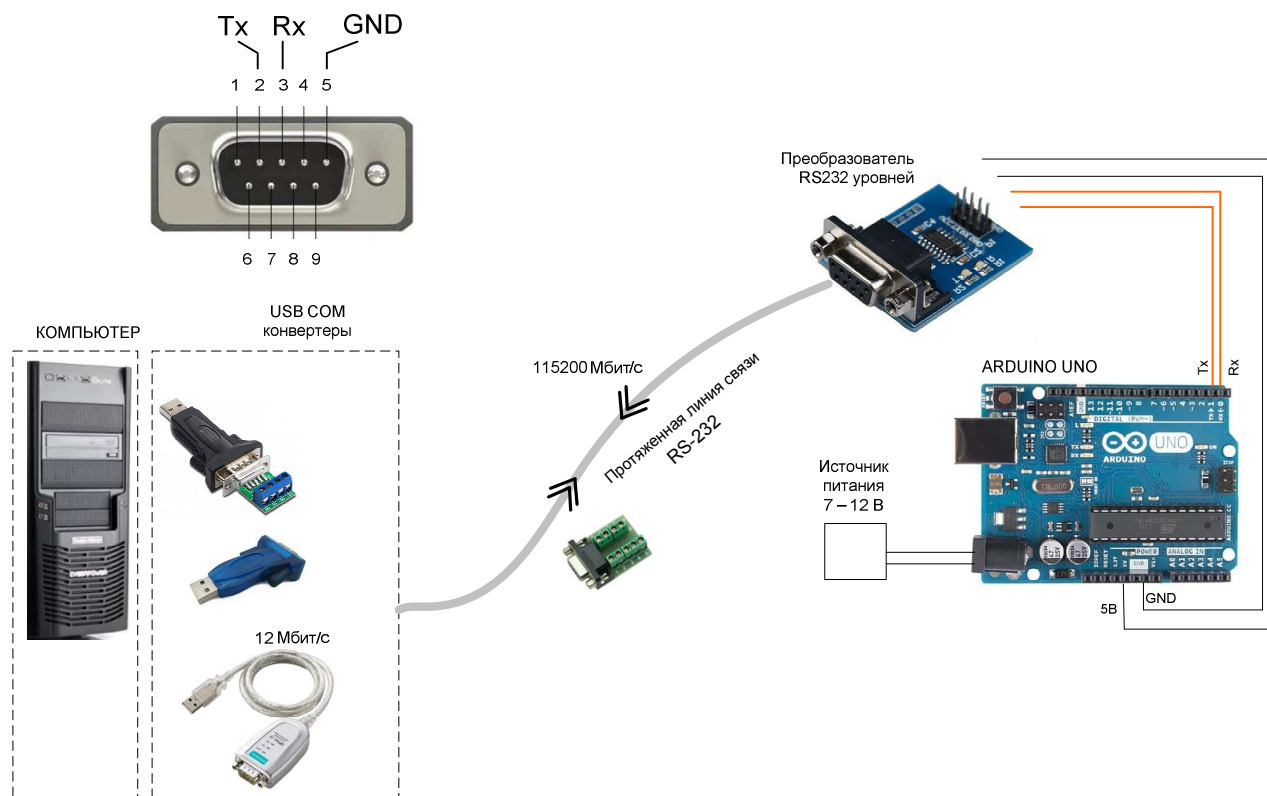
$$K = \frac{r_H - Z}{r_H + Z},$$

где  $r_H$  - сопротивление нагрузки;  $Z$  – в.с. кабеля.

Включения несогласованных элементов в USB линию может значительно исказить сигнал. Например, линия оказывается неработоспособной при включение в неё экстроводов из силового кабеля с волновым сопротивлением 10 .. 40 Ом.

### Структура канала USB – RS-232 – плата Arduino UNO

Для обеспечения устойчивой связи удаленного COM устройства с компьютером через USB порт длина USB канала сведена к минимуму, на выходе USB линии поставлен USB – RS-232 преобразователь, который через длинную линию подключен к преобразователю уровней +15/-15В == 0/5В, находящегося вблизи контроллера Arduino и подключенного к его UART порту, как показано на Рисунок 5. Скорость обмена данными в этой структуре такая же как и при подключении Arduino к компьютеру через USB кабель, но частота сигнала в протяженной линии почти в 100 раз ниже - как 0,115200 Мбит/с и 12 Мбит/с.



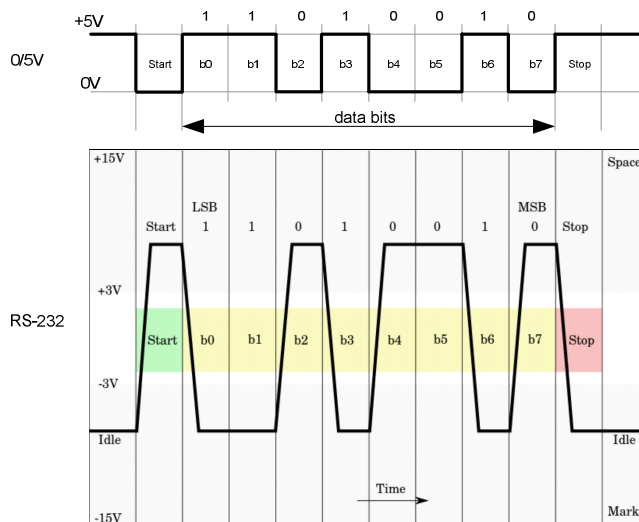
**Рисунок 5.** Схема подключения контроллера Arduino UNO к компьютеру через USB порт и длинные несогласованные линии. Обозначение контактов GND, передатчика Tx и приемника Rx на стандартном разъеме DB-9 COM порта компьютера показано сверху слева. Со стороны устройства сигналы TxD и RxD на разъема DB-9 надо поменять местами.

Интерфейс RS-232 имеет следующие характеристики [7,8].

Способ передачи сигнала	Однофазный
Максимальное количество приемников	1
Максимальная скорость передачи	460 кбит/с
Максимальная длина кабеля	15 м (для 460 кбит/с)
Синфазное напряжение на выходе	+/- 25В
Импеданс нагрузки	3 ..7 кОм
Допустимый диапазон сигналов на входе приемника	+/- 25В
Чувствительность приёмника	+/- 3В
Входное сопротивление приёмника	3 ..7 кОм
Ёмкость нагрузки	не более 2500 пкФ*

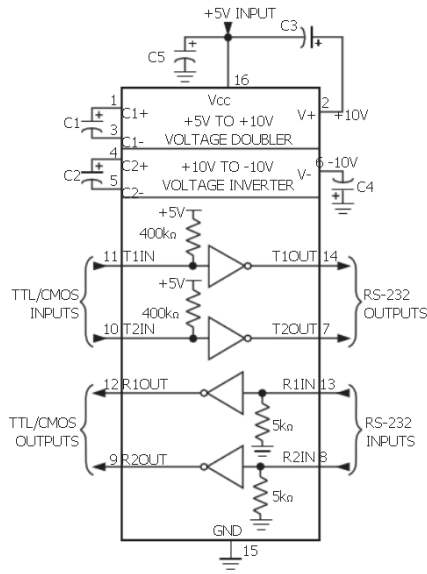
\* При использовании кабеля с малой емкостью связь может поддерживаться на расстояниях до 300 м [1].

Преобразователь RS232 уровней (Рисунок 5) не меняет последовательность бит. Он изменяет уровни сигнала 0/5 В в +12/-12 В и наоборот (Рисунок 6).



**Рисунок 6.** Временная диаграмма и уровни сигналов преобразователя RS232.

Для преобразования уровней сигналов RS232 могут использоваться микросхемы, например, MAX232 (компании Maxim Integrated Products), SP232 (Sipex), ADM232 (Analog Devices). Эти микросхемы имеют одинаковые характеристики и назначения выводов. Подключение преобразователя MAX232 показано на Рисунок 7 [6].

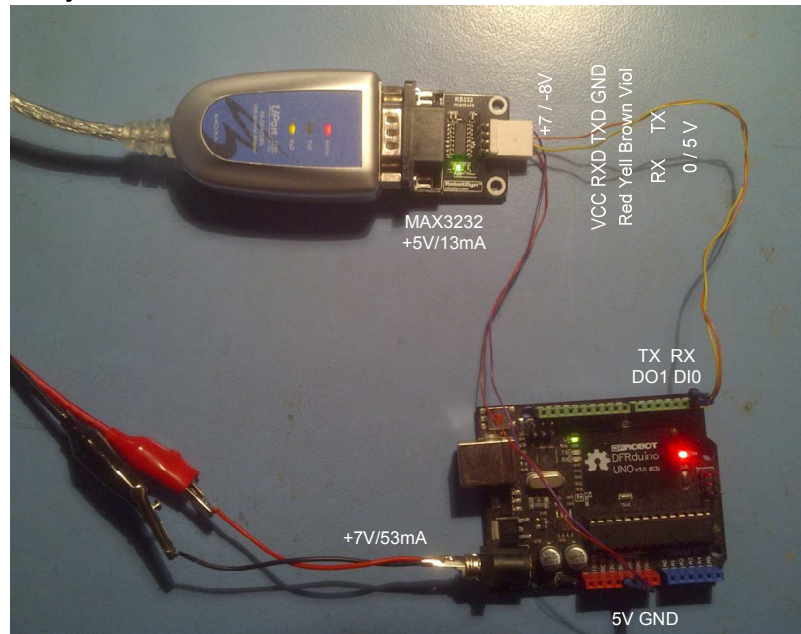


**Рисунок 7.** Схема подключения преобразователя уровней MAX232. Схема обеспечивает уровень выходного напряжения приблизительно  $\pm 7.5$  В соответствующий интерфейсу RS-232.

Рынок предлагает множество модулей преобразователей уровней построенных на базе перечисленных и других микросхем. Внешний вид одного из таких модулей показан на (Рисунок 5).

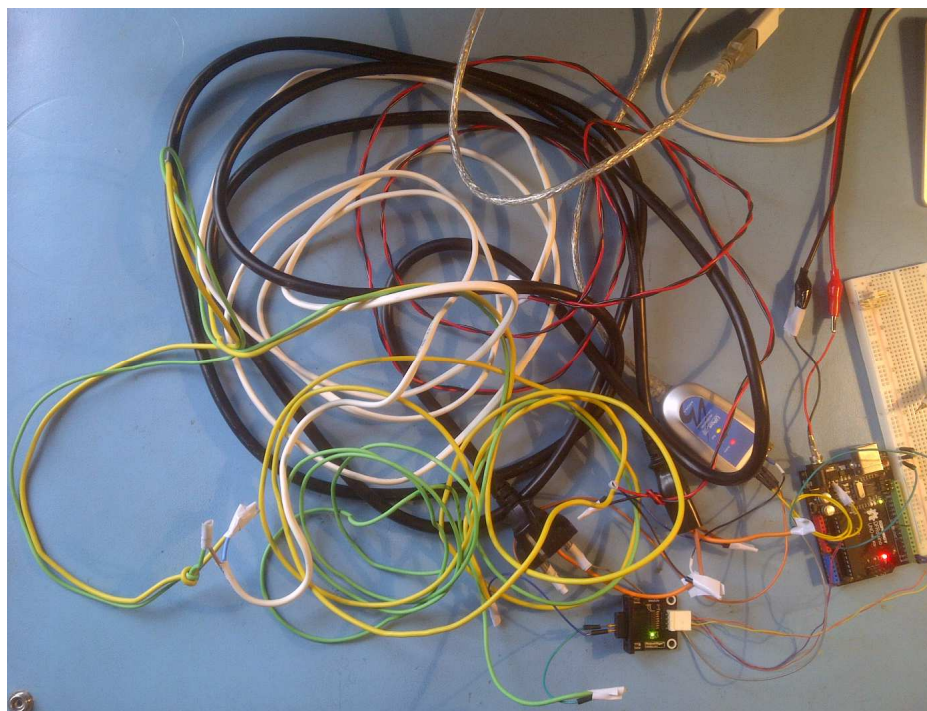
К компьютеру устройство можно подключить через стандартный COM порт, если он есть, или использовать преобразователь USB-RS232 (другие названия: USB-COM конвертеры, переходники или адаптеры), связанный с USB портом напрямую или через собственный USB кабель. Внешний вид USB преобразователей показан на Рисунок 5.

Вариант реализации макета линии COM устройство – USB порт компьютера без RS-232 линии показан на Рисунок 8.

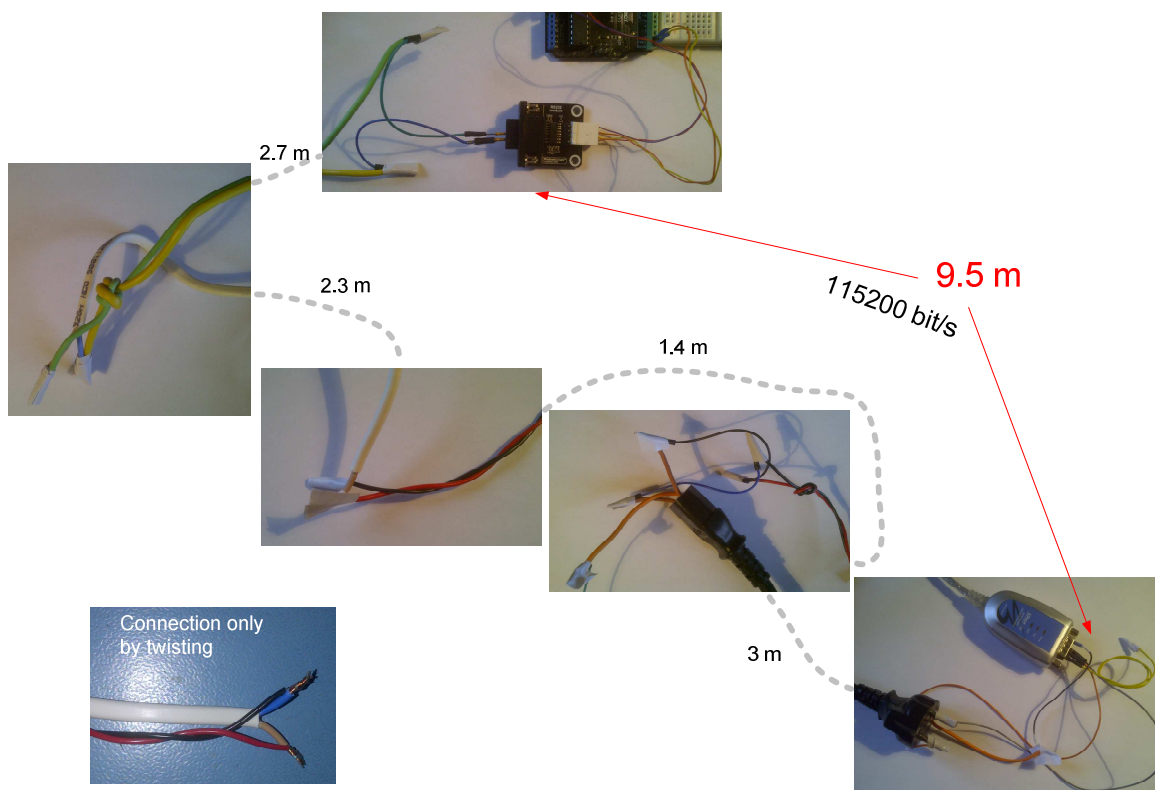


**Рисунок 8.** Вариант подключения контроллера Arduino UNO к преобразователю USB-COM компьютера.

Для проверки работоспособности канала обмена данными между контроллером Arduino UNO и компьютером через длинную несогласованную линию был собран провод показанный на Рисунок 9 и Рисунок 10. Куски провода соединялись скруткой или удерживались в гнездах разъемов на трении.



**Рисунок 9.** Канал RS232 из составного кабеля 9,5 м.



**Рисунок 10.** Куски провода канала RS232 из составного кабеля 9,5 м.



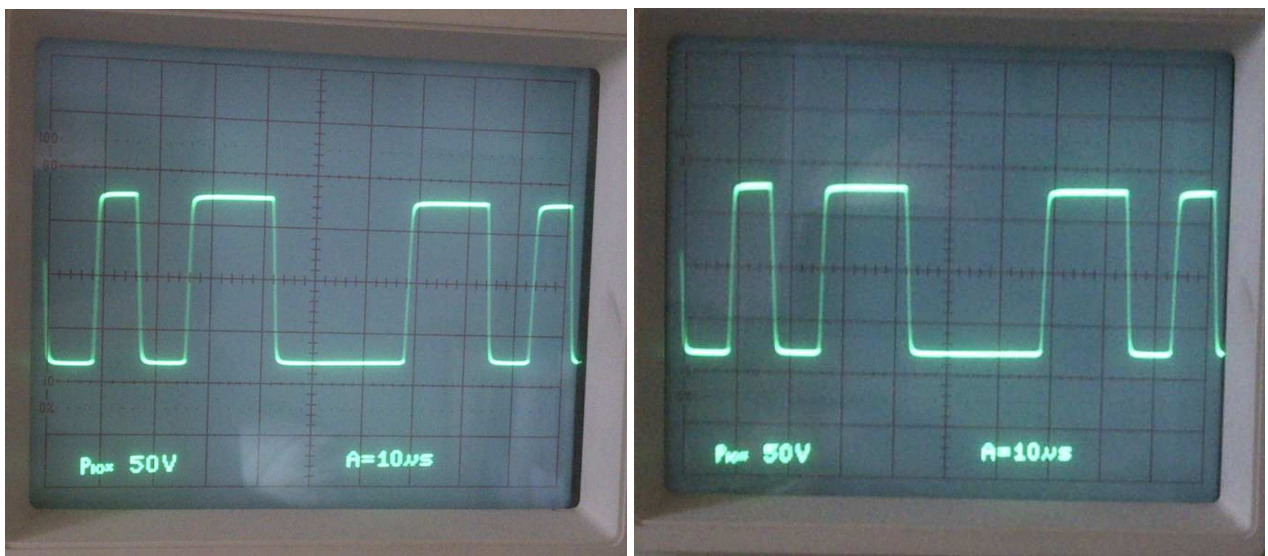
Передача и прием данных через COM порт контроллера Arduino UNO контролировалась утилитой компьютера COM Port Toolkit.

Используемая для тестирования линии программа Arduino UNO, передающая в COM порт байты данных и переключающая светодиод контроллера по приходу внешних команд, показана ниже.

```
void setup()
{
  pinMode(LED_BUILTIN, OUTPUT);
  Serial.begin(115200); // запуск последовательного порта, 9600,115200
}

void loop()
{
  if (Serial.available() >0) {
    mode = Serial.read(); // read byte
    switch (mode) {
      case 1:
        digitalWrite(LED_BUILTIN, HIGH); // turn the LED on
        break;
      case 2:
        digitalWrite(LED_BUILTIN, LOW); // turn the LED off
        break;
    }
  }
  Serial.print(5); // 6 or 9
}
```

Осциллограммы сигналов, снятые на концах состоящей из кусков линии RS-232 показаны на Рисунок 11. Данные передаются на частоте 115200 бит/с.



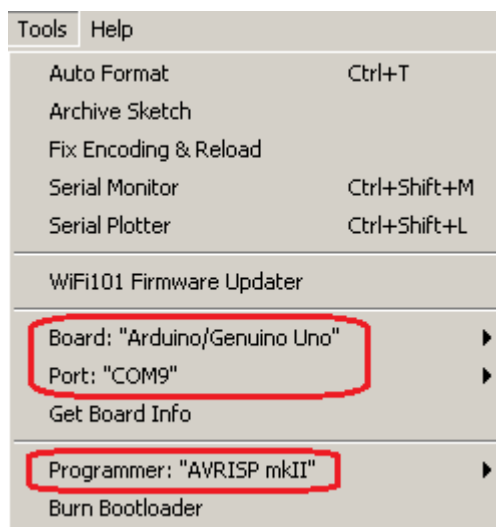
**Рисунок 11.** Сигнал амплитудой +7.5/-8 В на концах RS-232 линии составного кабеля длиной 9,5 м. Частота передачи данных 115200 бит/с. Сигнал не имеет заметных искажений.


## Прошивка контроллера Arduino UNO

Загрузка программ в контроллер Arduino выполняется при помощи его внутреннего загрузчика, который запускается сразу после включения питания контроллера, или после нажатия на кнопку reset платы, или когда компьютер через линию USB выдаёт сигнал сброса.

При подключении платы Arduino через канал RS-232 с двумя сигнальными линиями Tx и Rx при отсутствии линии сигнала сброса загрузка выполнялась в следующем порядке.

1. Запускалась среда разработки Arduino (как и в режиме загрузки через USB).



2. Загружалась программа (как и в режиме загрузки через USB).
3. Прошивка программы запускалась командой Ctrl+U или через кнопку  (как и в режиме загрузки через USB)
4. Дополнительно, после запуска прошивки и заполнения прогресс индикатора



нажималась кнопка Reset  на плате контроллера Arduino приблизительно на 0,5 секунды.



Успешная прошивка завершается сообщением

Прошивка выполнялась успешно и при кратковременном отключении питания контроллера, вместо нажатия на кнопку Reset.

Запуск загрузчика контроллера Arduino можно выполнять и в автоматическом режиме от компьютера, без нажатия на кнопку Reset или кратковременного отключения питания. Для этого необходимо, например, канал RS-232 с Tx, Rx, и GND дополнить линией RTS и подключить ее через преобразователь уровней ко входу RESET контроллера Arduino.

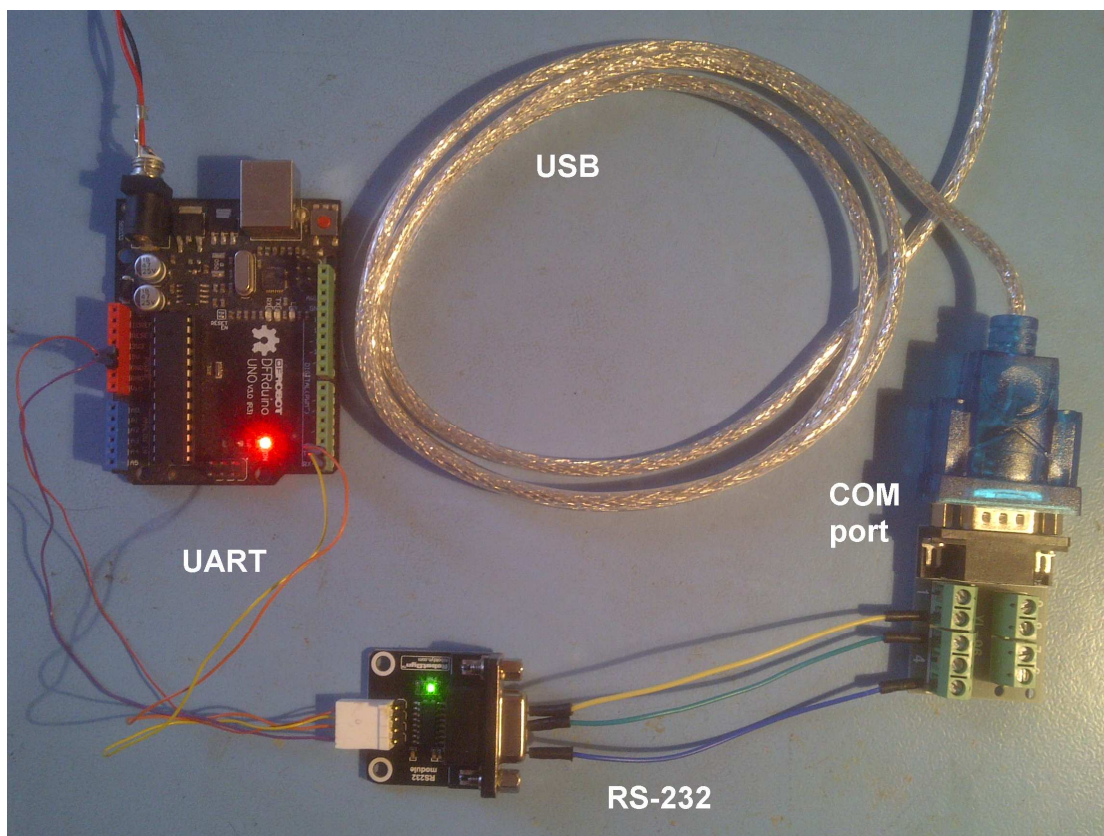
## Заключение

1. Передача COM данных через протяженную USB линию осуществляется пакетами на более высоких частотах (12 МГц) и требует согласованной линии.
2. Понижение частоты USB канала переходом с режима Full-speed (12 Мбит/с) на Low-speed (1.5 Мбит/с) выполняется на уровне BIOS или установкой соответствующего драйвера в диспетчере устройств и требует дополнительных проверок работоспособности передачи COM данных.
3. Сведение до минимума длины USB тракта и использование протяженной линии RS-232 для передачи COM данных без преобразования позволило обеспечить надежную передачу данных через куски несогласованных проводов общей длиной 9,5 м.
4. Передача RS-232 данных на низких частотах по линиям с малой ёмкостью (менее 2500 пФ) может осуществляться на расстояния до 300 м.
5. Использование дифференциальных линий RS-422 или 485 для удаленной связи с USB портом компьютера позволяет увеличить расстояние до 1200 м, но при решении вопроса согласования линий.

## ПРИМЕРЫ ПОЛУЧЕНИЯ ПРОВЕРЕННЫХ РЕЗУЛЬТАТОВ И ВАРИАНТЫ ДЛЯ САМОКОНТРОЛЯ

**Задание 1.** Проверка работоспособности канала передачи RS-232.

1. Соберите канал обмена данными по следующей схеме.



**Рисунок 12.** Канал обмена данными USB порт компьютера – COM устройство (UART порт контроллера Arduino UNO). USB кабель подключен к USB порту компьютера. Питание контроллера подается от источника питания 7-12 В или от 5В источника USB порта компьютера.

2. Используя данные раздела ОБЩИЕ СВЕДЕНИЯ, напишите для контроллера Arduino UNO программу циклической записи данных в COM порт и переключения светодиода на плате контроллера командами, считываемыми их COM порта.
3. Используя среду разработки контроллеров Arduino, пропишите рабочую программу в контроллер.
4. Используя интерфейс Serial Monitor среды разработки контроллеров Arduino, проверьте работоспособность связи контроллер-компьютер на частоте 9600 бит/с.
5. Используя утилиту COM Port Toolkit, проверьте работоспособность связи контроллер-компьютер на частоте 115200 бит/с.
6. Увеличьте длину канала RS-232 на 10 метров и вновь проверьте работоспособность связи контроллер-компьютер на частоте 115200 бит/с.

7. Выполните типовое подключение контроллера к компьютеру через USB линию. Нарастив длину USB линии соединением нескольких USB кабелей, определите длину, при которой передача данных перестает работать.

## **КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. В чем отличие подключения удаленного компьютера через USB и RS-232 шины?
2. Сравните требуемую полосу пропускания шины RS-232 и USB2 (Full-speed) в режиме приёма/передачи данных через COM порт.
3. Сравните помехоустойчивость шин UART контроллера, RS-232 и USB.

## **БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. WikipediA. The Free Encyclopedia. <https://en.wikipedia.org/wiki/RS-232>
2. WikipediA. The Free Encyclopedia. <https://en.wikipedia.org/wiki/USB>
3. WikipediA. The Free Encyclopedia. [https://ru.wikipedia.org/wiki/Американский\\_калибр\\_проводов](https://ru.wikipedia.org/wiki/Американский_калибр_проводов).
4. Способы передачи по витым парам. <https://studfiles.net/preview/2082678/>
5. USB2.0 А вилка ↔ В вилка мультимедийный кабель. [http://s3-eu-west-1.amazonaws.com/idei74/upload/instruction/instruction\\_7caa11754921437fad415d71df3d1039.pdf](http://s3-eu-west-1.amazonaws.com/idei74/upload/instruction/instruction_7caa11754921437fad415d71df3d1039.pdf)
6. +5V-Powered, Multichannel RS-232 Drivers/Receivers. <https://www.maximintegrated.com/en/products/interface/transceivers/MAX232.html>
7. Dr. Bob Davidov. Преобразователь интерфейса порта последовательной передачи данных. [http://portalnp.ru/wp-content/uploads/2013/09/08.02\\_Converter-of-serial-data-transmission\\_Ed\\_2.pdf](http://portalnp.ru/wp-content/uploads/2013/09/08.02_Converter-of-serial-data-transmission_Ed_2.pdf)
8. Dr. Bob Davidov. <http://portalnp.ru/author/bobdavidov>