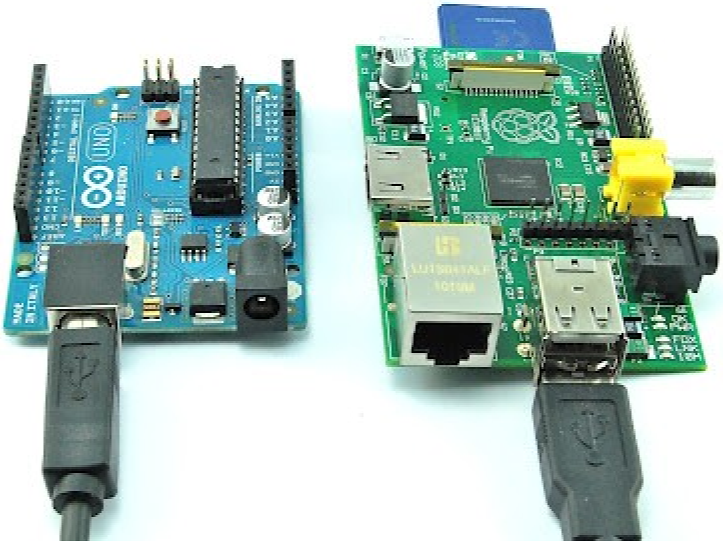
OscarLiang.com
Share Knowledge and Ideas

# CONNECT RASPBERRY PI AND ARDUINO WITH SERIAL USB CABLE



**Share this:**

Facebook | Google | Twitter | Reddit

## Using USB Cable Between Raspberry Pi and Arduino

There are many ways of connecting the Raspberry Pi and Arduino, such as using the GPIO with the Arduino. To Setup your Raspberry Pi, check out this article.

To Demonstrate how this works, I will be doing two little projects, one for data going to Raspberry Pi from Arduino, the other one for the opposite. First of all, make sure you have installed pySerial, which gives you the ability to read from and write to the serial port with Python Programming language. People have used it before with Arduino, so it’s been proven to be working, you can check this out.

## Arduino Talking to Raspberry Pi via USB cable

We will send ‘Hi’ from the Arduino to the Raspberry Pi every 2 seconds. Here is the Arduino source code.

```
[sourcecode language=”cpp”]
void setup(){
Serial.begin(9600);
}

void loop(){
Serial.println(“Hello Pi”);
delay(2000);
}
[/sourcecode]
```

Run Python 2 on Raspberry Pi. You will find this from the menu under Programming, you should use Python 2 not 3.

Type the following after >>>

```
import serial
ser = serial.Serial('/dev/ttyACM0', 9600)
```

The first argument – /dev/ttyACM0 is the name for the USB interface used. To find out the port name, we need to run this command in terminal without Arduino plugged in:

```
ls /dev/tty*
```

Now plug in your Arduio and run the command again. If a new name appears, then this is the name of your port.

The second argument – 9600 is the baud rate and should match with what you set in the Arduino program.

Now lets start a loop listening for messages from the Arduino.

```
while 1 :
    ser.readline()
```

You will need two hit enter twice after you type the second line. Messages ‘Hi’ should now start to appear every 2 seconds. You can press Ctrl + C to stop (interrupt) the Python program.

## Raspberry Pi Sending Data To Arduino

In this example, Raspberry Pi will be sending back a single number, and the Arduino will turn on and off the LED on Pin 12 so many times.

[sourcecode language=”cpp”]

const int ledPin = 12;

void setup(){

```
Serial.begin(9600);
}

void loop(){
if (Serial.available()) {
light(Serial.read() – ‘0’);
}
delay(500);
}

void light(int n){
for (int i = 0; i < n; i++) {
digitalWrite(ledPin, HIGH);
delay(100);
digitalWrite(ledPin, LOW);
delay(100);
}
}
[/sourcecode]
```

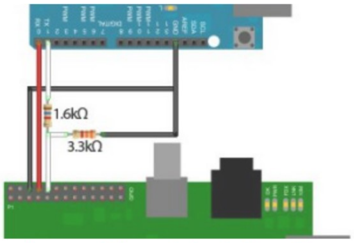
On the Raspberry Pi Side, you need to type

```
ser.write('3')
```

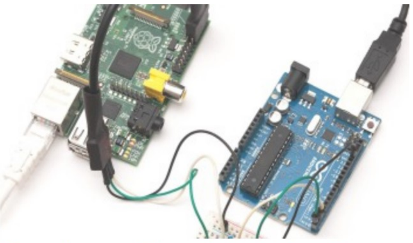
Now you should see the LED on the Arduino light up 3 times.

There you go, be creative and you will find there are so many things you can do. For example we could control some motor or LCD on the Arduino from the Raspberry Pi.

**Related**



Raspberry Pi and Arduino Connected Over Serial GPIO
21st May 2013
In "Electronics"



Raspberry Pi and Arduino Connected Using I2C
25th May 2013
In "Electronics"

Arduino Yun - Arduino on the Cloud Raspberry Pi's Competitor?
22nd September 2013
In "Reviews and News"

Posted in Electronics, Featured, Raspberry Pi and tagged arduino, raspberry pi on 20th May 2013 [https://oscarliang.com/connect-raspberry-pi-and-arduino-usb-cable/] . 49 Replies

## 49 thoughts on “Connect Raspberry Pi and Arduino with Serial USB Cable”

Miku
1st January 2017 at 2:59 pm

HI Oscar
can you explain why you need to use Python 2 and not 3. I have some code working on 3 hence need it to work on python 3.

jack liu
30th September 2016 at 4:19 am

thats good for my job, thank you

Felix
23rd August 2016 at 11:32 am

Hello

when I execute

ser.write('3')

Raspberry returns: "1"
And doesn't work :/

Ardulink
8th July 2015 at 7:26 am

Hi Oscar,

Your article is very clear and useful. Can advice you about my Java library for PC/arduino communication called Ardulink? Whit it I've connected a PI with arduino via USB cable like you but with Java. I'll take a look to your other guides in order to understand if Ardulink can be used also with I2C or GPIO.

Thank you
Luciano

Hi Oscar , I want to ask. What if the raspberry pi sending many data to arduino? do you have some reference code? thank you

wildha
7th July 2015 at 6:31 am

I want to ask. What if the raspberry pi sending many data or number to arduino? do you have some reference code? thank you.

Jeff
30th March 2015 at 7:19 pm

Hi Oscar,
Thanks for useful example which got me going easily. Uno3 is gathering times for activities and sending through to Pi for further processing as required.

I can identify the usb port as you explain. The new Pi has 4 ports. Different users are plugging in different things in different ports (memory sticks) at times so the serial.Serial is not constant.

How can I programmatically trap a situation where the port 'no longer exists' and discover the right label? (Using Python)

Thanks, Jeff

Brian

I just hacked away at this for the past few hours to get it working. I am sending data from the Raspberry Pi to the Arduino via the USB cable. I wanted to do this because I WAS using a line level converter board and I wanted to simplify my project: It eliminates a board, a pile of wires, and a wall wart power supply! Like most tutorials, there are a few things that are missing. I wish people who posted tutorials would try to execute them EXACTLY as written to make sure they work before releasing them into the wild. Unfortunately, the world is an imperfect place. Without further ado, here is what I did:

On the Raspberry Pi in Python I created a file called serial_test.py. BTW, DO NOT name your file serial.py as that will cause problems when it ties to “import serial”. It can’t import itself! Here is what the file contains:

```
import serial
import time
ser = serial.Serial(“/dev/ttyACM0’, 9600)
//read up on how to get your raspberry pi to tell you what the correct serial port name is!
//Yours could be ACM0 or it could be something else.

time.sleep(2) //this is required because the arduino resets when a serial connection is established
//then, send the data
ser.write(‘9’) //using then number 9 just as an example. This program only transmits ONE byte!
```

On the Arduino, here is what I have (among many other things in a program file)

```
//this is the LED on the UNO board, no need to wire one up externally with a resistor.
const int ledPin = 13;
```

In void setup() I have:

```
pinMode(ledPin, OUTPUT);
Serial.begin(9600);
```

```
int n = (Serial.read() – '0'); //must subtract the ASCII value of zero (48) to get the true value from the character that is sent
for (int i = 0; i < n; i++) {
digitalWrite(ledPin, HIGH);
delay(500);
digitalWrite(ledPin, LOW);
delay(500);
}
```

When the python script is executed on the Raspberry Pi, the LED's should blink a few times rather quickly (too fast to count) on the arduino as the serial connection is established. Then, after the 2 second timer has expired you should be able to count the LED blinking (in this example, 9 times). Love it when it finally works! Now I have to figure out how I can send more than one byte.

I know I struggled getting this to work. So, I hope this info helps reduce the stress level of at least one person so the world can be a happier place! =)

---

Michael
28th July 2016 at 1:48 pm

How do you send data that is more than one byte long? Can I send and receive a multi-byte message with:

```
ser.write('message')
```

and receive it with just:

```
string message = "";
```

```
}
```

Filip
23rd January 2017 at 8:22 pm

Dude , saved me a ton of time trying to figure out the import bug , a bit new to programing so didnt think of that being a problem , thank you very much !!!

vivek
15th February 2015 at 7:44 am

i hav a problem with pi serially communicating with arduino using bash script
i want to send a character to arduino…but it seems dat the echo command is not doin the trick…it initiates the communication which i knw as the arduino blinks initially….i guess the arduino is getting reset as the echo command closes the port after sending the character and the arduino has little time….here is my code …help me out!!!!!!!!!!!!!!!!!!!!!!!!!!

```
#!/bin/bash
echo a > /dev/ttyUSB0
echo “status :204 No Content”
echo “Content-type : text/plain”
echo “”
```

Jim
4th February 2015 at 11:00 am

whitespace away…

Jim
4th February 2015 at 10:57 am

Replying somewhat late to Jays comment. I have modified the code to the following

```
import serial
ser = serial.Serial('/dev/ttyACM0', 9600)
while 1 :
textln = ser.readline()
print( textln )
```

## very important to indent in python since this will put the 2 lines inside of the while loop …

Vinicius Souza
17th January 2015 at 3:08 pm

Very helpful !!!

Jessica Hart
16th January 2015 at 8:25 am

Arduino to Raspberry Pi with USB Serial Connection. Quick Guide to Connecting your Raspberry Pi to Arduino via USB Cable.

Gavin Bath
8th January 2015 at 7:37 am

Thanks so much for writing this tutorial. This helped me get up and running quickly with comms between pi and Arduino for the irrigation controller I'm building. Much appreciated!

---

Jay
15th December 2014 at 3:04 am

Hi,

I am trying to capture output from an Uno R3 on a Pi B+ and am at a loss because your sample code is not working. I verified that ttyACM0 is the correct device and used both the sample sketch you provide and the Python code. I see no errors in either code, and everything appears to run properly, but "Hello World" never appears on the Pi. Here is the code that I am using for confirmation purposes.

```
void setup() {
Serial.begin(9600);
}

void loop() {
Serial.println("Hello World");
delay(2000);
}
```

And here is the Python code.

```
import serial
```

ser.readline()

What am I doing wrong? Is here something about B+ that invalidates this code?

Thank you!

taotao
28th February 2015 at 8:04 pm

your program is right，but your pi’s program donot let pi print anything then it recieved the massage that send by arduino，here is my code
message= ser.readline（）
print message

Taki Uddin
18th June 2016 at 6:25 pm

Please remove double quotes. Single quotes only..lol it’s damn to late but why not..:p

Serial.println(‘Hello World’);

Chris
25th November 2014 at 8:28 pm

Hi,
but what, if I want to let the led blink 10x or 25x?

P.S. With you code, the LED flashes 2x when I send 25

Thanks

NicoHood
18th April 2014 at 9:35 pm

Hi,

thx for all your helpful tutorials! I created a way to communicate between Raspberry and Arduino very easily. Its a Protocol library for Arduino and Raspberry. Have a look if you want to and give me feedback :)

nicohood.wordpress.com/2014/04/18/arduino-raspberry-pi-serial-communication-protocol-via-usb-and-cc/

Nico

Patrick
8th April 2014 at 1:54 pm

Hi Oscar,

inside the call of the light function, you substract a null from the incomming byte. Is this null a part of the ascii-string, wich was send by the RasPi? I guess, it is the null terminator and I have to delete it from the received string, because if I don't, the value of the incomming byte is interpreted as decimal ascii string, so the LED will blink 49 times in case of one.
Am I right?

Greetz

Patrick

---

**Oscar** Post author
8th April 2014 at 2:30 pm

Hi Patrick

Subtracting character '0', you can convert the input char into integer. For example char '7' has ASCII 55, and char '0' has ASCII 48. so '7' subtract '0' you get int 7.

cheers

Oscar

---

Hitesh Giri
2nd April 2014 at 10:28 am

Sadar pranaam _/_ (Namaste in Marathi)

Thank you so much for this helpful tip. Saved me a ton of time and effort!

Roseanna
28th February 2014 at 1:53 am

I like the helpful info you supply in your

articles. I'll bookmark your weblog and check again right here regularly.

I am slightly sure I will be told plenty of new stuff

proper here! Best of luck for the next!

Muriel
17th February 2014 at 11:58 am

Highly descriptive article, I enjoyed that a lot.

Will there be a part 2?

antoine9298
24th January 2014 at 9:28 pm

is it a library for php? or in command line?

or maybe you can execute phyton code in php?

Lubomir Spacek

Hi,

nice blog.

Need Raspberry Pi collocation ? I found on new Raspberry Pi collocation project, link-blocked

Enjoy with RPi !

Ashok Subramaniam
16th January 2014 at 5:30 pm

Can I use this to communicate with an arduino due board?

Oscar
16th January 2014 at 8:50 pm

yes.

Corazon
15th January 2014 at 4:29 pm

Woah! I'm really loving the template/theme of this
website. It's simple, yet effective. A lot of times it's very difficult to
get that "perfect balance" beteen superb usability and appearance.
I must say that you've done a verty good job with this. In addition,
the blog loads extremely fast for me on Internet explorer.
Exceptional Blog!

3rd January 2014 at 2:42 pm

Hi Oscar, thanks for your tutorial which I'm using to control my heating. Now I finally understand how things work!

Gohper
18th December 2013 at 7:42 am

Great!
I have not thought about the usb connection between Pi and Arduino before, maybe to simple!
For a few days I have been locking at wifi shield with antennas for the Aurdino, but a Pi with a wifi dongle is a more powerfull and cheaper wifi connection :) !! And in addition the Pi can do other things aswell!

pete
29th November 2013 at 4:24 pm

I wonder if anyone can help. With two arduino Leonardos connected via a powered USB hub to a raspberry pi (rev A), and communicating perfectly (via usb serial) with bash scripts running on the Pi, after several hours i lose ethernet coms on the pi and it subsequently reboots (maybe watchdog is doing this, not sure). i have over 1 amp of 5v supplied to the pi, and the Leos are powered by the hub and continue to run just fine after coms are lost, so i don't think it's a power supply issue on the pi. any other ideas where i might look for the cause of this frustrating problem? thanks.

30th November 2013 at 8:54 am

Hi Pete, did you see my reply to the last comment?
have you tried powering the arduino Leonardos using external source? see if that fix your problem.
is there any other components you are running on the Arduino?

T.J.
21st November 2013 at 7:16 am

This may be a silly question, but is the Pi powering the Arduino via USB also? If I don't want to run the risk of making the Pi brown-out and reset can I apply DC power to the Arduino in this case or is that a bad idea?

Much thanks, this tutorial may help make possible a project I'm working on currently.

Oscar
21st November 2013 at 9:28 am

In theory you can. Recommended current output of the USB pin from RPI is around 100mA, and the current input of the Arduino Uno is 70mA, so we have just enough current to run an extra LED on the arduino. But if you want to add any more current driven devices on the Arduino, you will need an extra power source (e.g. 9V battery) to the Arduino.

Yun
28th October 2013 at 1:21 pm

Hello Oscar,

Thanks for your great post. I followed your tutorial and connect Raspberry Pi and Arduino Nano V3 with a micro USB cable, and it works fine! However I meet a problem after rebooting the RPi: the serial port of Arduino is not listed when I try “ls /dev/tty*”. I pressed the reset button on Arduino board several times, and it didn’t help. I have to unplug the Arduino and plug in again to make RPi recognize it. I wonder if you have ever met this kind of issue, and if you by any chance that know a workaround for it? Thanks in advanced.

Max
1st June 2016 at 2:05 am

I have had a similar issue. Did you find a solution?

Yun
27th October 2013 at 9:49 am

Thanks for the very useful tutorial, that’s really the one I am looking for. Just two comments:

1) to print out the data from Arduino to RPi, you need to print it in the loop (which is missing in the post):
print ser.readline()

2) if your RPi trys to send data to Arduino, and it doesn’t work, that most probably because the write command comes before the serial device get initialized. You can find these information from http://playground.arduino.cc/Interfacing/Python:

“ the arduino serial device takes some time to load, and when a serial connection is estab-

Any write() commands issued before the device initialised will be lost. A robust server side script will read from the serial port until the arduino declares itself ready, and then issue write commands. Alternatively It is possible to work around this issue by simply placing a 'time.sleep(2)' call between the serial connection and the write call. ”

For me the 'time.sleep(2)' could do the trick (remember to import time in py file). A better application should detect the serial device before sending data to it.

GCL
27th October 2013 at 6:04 am

Hello Oscar!
What pray tell is that shield on top of the Arduino (any) doing? That is what is it wired to do. It looks rather like the prototype shield that Limor Freed also known as Lady Ada designed. Yours is the first easy to use article on the subject of connecting the two together.

Oscar
27th October 2013 at 9:54 am

That's shield used in one of the application that utilize the Serial connection between RPi and Arduino, it's not relevant to the tutorial :-)
sorry about the confusion.

Elia
4th October 2013 at 6:20 pm

msg = serial.Serial('/dev/ttyACM0')

msg.write("Riga 1&$Riga 2&")

But not write in the serial port, why? thank you

Jes
16th September 2013 at 5:09 pm

Hi!
Great tutorial! One question: to run code on both the pi and the Arduino when sending information from the Pi, do I need to have the Arduino IDE installed on my Pi? or do I just pre-load the code I want onto the Arduino using my laptop before connecting them?
Thanks!

Angus-pangus
26th July 2013 at 11:17 pm

Great stuff on your pages

I dunno if this is of any help to people out there but I struggled to do this with my Arduino Due over the 'native' micro-USB port (i.e. not the 'programming' micro-USB port that I connect to my laptop).

I realized that

Serial.println("Serial data not outputted");

does not active the native port on Arduino Due, only the Rx/Tx pins and the programming

SerialUSB.println(“Oscar Liang kicks ass”);

does. This solved all of my problems. Again thanks for your good tutorials, keep up the good work.

Alex
25th July 2013 at 3:41 pm

Hey

Great Tutorial, but I have on problem:

ImportError: No modul named serial

Have anyone a idea?

Thank you!

Alex

Oscar
25th July 2013 at 3:53 pm

sounds like you are missing the “pyserial” library on your Pi. Install that and try again?

Golan Gabay
7th July 2013 at 9:26 pm

Hi there,

I followed your tutorial and it’s working great but I have another problem…

I’m trying to send data through the USB port from my Pi to my Arduino using the php serial

When I'm sending it, I see the Arduino's serial LEDs blink which means It receives something but the data is invalid since I try showing it on a display and I see nothing…
Once I run this python script as an endless loop:

```
import serial
ser = serial.Serial('/dev/ttyACM0', 9600)
While 1:
1=1
```

And I'm sending the SAME data using the php class I get valid bytes and I see them on the display.
When I stop this script the data is invalid again and I don't see anything on the screen.
I think it's something to do with baudrate or something like that (stop bits or something else) even though the php is sending in the right baudrate since this Arduino guy is showing serial data flow (the LEDs are blinking) but no output is showed…
Maybe the python serial setup overrides the default Pi's serial setup and that's how it might work…?
I tried anything I could find on the net but nothing helps…
Please help!
Thank you in advance!
Golan

---

Gérard
31st May 2013 at 8:14 pm

Hello Oscar, very well done. I was just looking for that. I am new to raspberrypi and python and want to connect a CellLog8s with usb to the pi to read out the data stream the CellLog sends. I used the following code in python:

```
import serial
ser = serial.Serial('/dev/ttyUSB0', 128000)
```

ser.readline()

After line 2 i get an error. When changing baudrate to 9600 there is no error, but i can't see anything on the screen after the last line and 2 time enter.
What is the maximum baudrate i can use in python and how can i initiate databits and stopbit?
The CellLog is sending with 128000 baud, 8 databits and 1 stopbit.

**I only check blog comments once or twice a week, if you want a quick reply you can post your question on this forum IntoFPV.com... You might get a faster response from me there (multirotor related only).**