Dr. Monk's DIY Electronics Blog

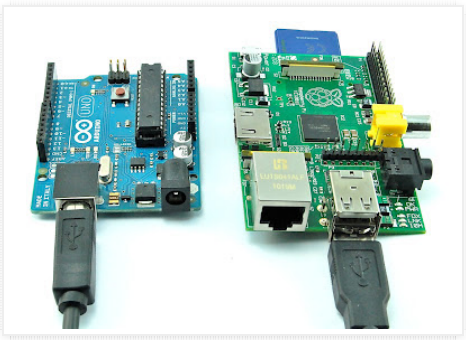
Open Source hardware, Arduino, Raspberry Pi, BeagleBone, DIY electronic construction, reviews, projects, how-tos and recipes.

Sunday, April 29, 2012

Raspberry Pi and Arduino

Note. There is now a followup to this post here.

The Raspberry Pi is creating quite a storm of interest. I have just got mine and one of the first things that I wanted to try was to get it talking to an Arduino over USB using Python.



.. and you know what? It proved to be a lot easier than I expected. This is mainly because, after all, despite its diminutive price tag, the Pi is just a Linux box. I got communication working both ways, with the Arduino sending 'Hello Pi' to the Pi and at the same time, testing for a digit coming in. When it receives a digit, it flashes the number of times indicated by the digit.

Arduino

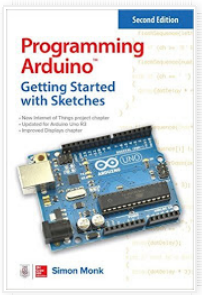
Let's start with the Arduino end. I used an Arduino Uno and Arduino software version 1.0. I haven't tried an older board, but I suspect the FTDI generation Arduinos before the Uno may have trouble with USB.

Here is the sketch - paste it into a new Arduino IDE window and load it up onto your Arduino using your regular computer.

```
const int ledPin = 13;

void setup()
{
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  Serial.println("Hello Pi");
  if (Serial.available())
  {
     flash(Serial.read() - '0');
  }
  delay(1000);
}

void flash(int n)
{
  for (int i = 0; i < n; i++)
  {
    digitalWrite(ledPin, HIGH);
    delay(100);
    digitalWrite(ledPin, LOW);
    delay(100);
  }
}
```

@simonmonk2 twitter

Learn Arduino



Learn Arduino

Topics

- arduino (32)
- raspberry pi (16)
- android (6)
- .net gadgeteer (3)
- ioio (2)
- python (2)
- handbag (1)
- processing (1)

Popular Posts

Raspberry Pi and Arduino Note. There is now a followup to this post here . The Raspberry Pi is creating quite a storm of interest. I have just got mine and one o...

Raspberry Pi and Breadboard (Raspberry Leaf)
I thought I would share this little helper I have made to simplify using the Pi with jumper wires. If you are using your Raspberry Pi...

Arduino Timer Library
I have developed a simple to use library that gets around a load of problems that arise when you start trying to do much inside 'loop&#...

External Links

- eCalc Calculator
- MonkMakes.com
- Adafruit
- Shield List

PAGE VIEWS (LAST 30 DAYS)

1 8 6 8 2

Blog Archive

Blog Archive

Raspberry Pi

There is a Python library for serial communications called 'pySerial' which has history with Arduino. So, I stood on the shoulders of giants and adapted the instructions found here.

Step 1. If you are not reading this page on your Pi, then switch now, so you can copy and paste.

Step 2. Browse to here and download **pyserial-2.5.tar.gz (106.3 kB)** and save it somewhere convenient. I saved it to the 'other' folder on the Desktop.

Step 3. This is a gziped tar file. Which needs unzipping and untaring. To unzip it open a Terminal, which you will find from the 'start menu' under 'accessories'. Now paste the following commands into it.

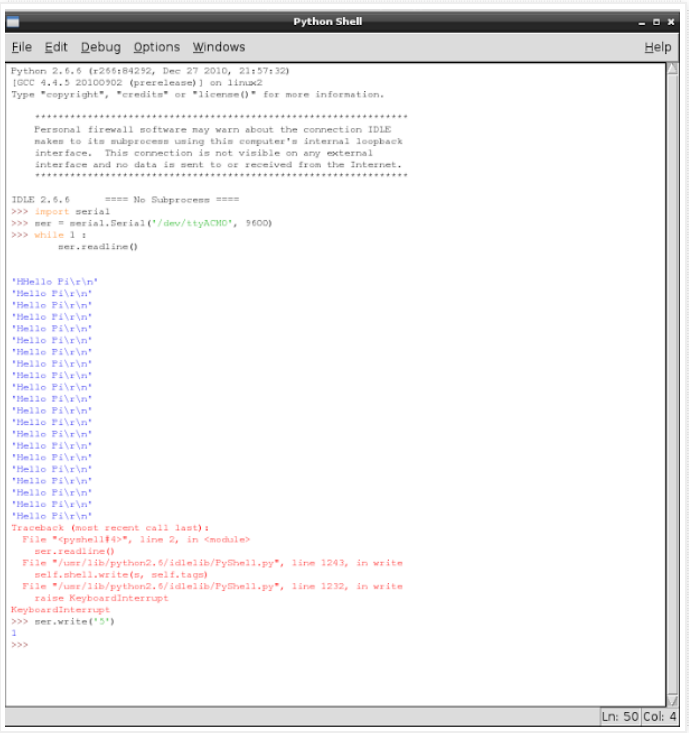
```
cd /home/pi/Desktop/other
gunzip pyserial-2.5.tar.gz
tar - xvf pyserial-2.5.tar
```

Step 4. Install pySerial, by typing these lines in your terminal window:

```
cd pyserial-2.5
sudo python setup.py install
```

Step 5. Run Python 2. You will find this from the menu under Programming - Use Python 2 not 3.

Thats it! Now we just need to write some Python to access the Serial port. So type the commands shown in the transcript below.



You type the parts after >>>

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

Note that the second argument here (9600) is the baud rate and should match whatever you put in your Arduino sketch.

/dev/ttyACM0 is the name for the USB interface to the Uno, at least it was for my Uno. The way to discover the port name is to run the following command in the terminal without the Uno plugged in.

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

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

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 should now start to appear!

You can see in the Blue writing where the Arduino is talking to the Pi. Then some error trace as you press ctrl-C to interrupt the messages coming from the Arduino.

When you type

```
ser.write('5')
```

you should see the LED on the Arduino flash 5 times.

There are many possibilities here, we could put a motor shield or LCD shield onto the Arduino and control it from your Pi.

About the Author
These are my books. Click on the image below to find out more about them.

Posted by Simon Monk at 2:27 PM
Labels: arduino, python, raspberry pi

37 comments:

MagPi Magazine said...
Hi Simon

Thanks for an interesting blog. We would be interested in covering this in our magazine the MagPi.

If you would be interested in working with us, you can contact us at editor@themagpi.com

Thanks

The MagPi team
April 30, 2012 at 12:24 PM

Miles said...
Great article, will creat link to it on our forum, I'm sure people will be interested.

Miles from Ciseco
May 1, 2012 at 7:36 AM

boci said...
Can you compare the power consumption ?
May 7, 2012 at 8:00 AM

ezstartup said...
How do you connect the two cylons, one usb cable between arduino and raspi?
May 8, 2012 at 3:59 AM

ezstartup said...
How do you connect the two cylons, one usb cable between arduino and raspi?
May 8, 2012 at 3:59 AM

Simon Monk said...
Power consumption of Pi:
Startup 100mA settling to 70mA after 30 seconds.

Yes, connected by USB, Pi providing power to Arduino (25mA)
May 8, 2012 at 5:47 AM

Simon Monk said...
Just glanced at my power supply again and Pi drawing 370 mA now.
May 8, 2012 at 1:47 PM

Tim said...

Hi Simon,

I found the article really interesting. I was wondering whether you could use the same/similar commands to read an analogue input off the arduino from the pi? Esentially what i would lke to do is use the pi for back end data logging/processing.

May 29, 2012 at 4:02 AM

Tim said...

Hi Simon,

I found the article really interesting. I was wondering whether you could use the same/similar commands to read an analogue input off the arduino from the pi? Esentially what i would lke to do is use the pi for back end data logging/processing.

May 29, 2012 at 4:02 AM

Lee Tickett said...

Any reason why i get no output when i use your code?
I changed it to sys.stdout.write(ser.readline()) and it works :)

June 3, 2012 at 2:27 AM

W8XR said...

Works great - thanks!

When I tried it with my Arduino Uno plugged into a powered USB hub attached to the RPi, I couldn't see ttyACM0.

It works fine directly plugged into the RPi.

Should it work with the hub? (I'm thinking there's something wrong with the hub...)

June 9, 2012 at 1:58 PM

W8XR said...

Works great - thanks!

When I tried it with my Arduino Uno plugged into a powered USB hub attached to the RPi, I couldn't see ttyACM0.

It works fine directly plugged into the RPi.

Should it work with the hub? (I'm thinking there's something wrong with the hub...)

June 9, 2012 at 1:59 PM

DougEdey said...

Hey Simon, thanks for the writeup.

With regard to the older boards (FTDI) I can confirm that my Arduino Minis work fine using this method (they can even be programmed from the Pi)

dmesg output:

```
[ 6.152837] usb 1-1.2: new full speed USB device number 4 using dwc_otg
[ 6.279186] usb 1-1.2: New USB device found, idVendor=0403, idProduct=6001
[ 6.295385] usb 1-1.2: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 6.310300] usb 1-1.2: Product: FT232R USB UART
[ 6.323101] usb 1-1.2: Manufacturer: FTDI
[ 6.334651] usb 1-1.2: SerialNumber: A600aPpA
```

and program:

```
doug@raspberrypi ~/arduino_dev/sketchbook/initial_playing $ sudo make upload
for STTYF in 'stty -F' 'stty --file' 'stty -f' 'stty <' ; \
do $STTYF /dev/tty >/dev/null 2>/dev/null && break ; \
done ;\
$STTYF /dev/ttyUSB0 hupcl ;\
(sleep 0.1 || sleep 1) ;\
$STTYF /dev/ttyUSB0 -hupcl
/usr/bin/avrdude -q -V -p atmega168 -C /etc/avrdude.conf -c arduino -b 19200 -P /dev/ttyUSB0 \
-U flash:w:build-cli/initial_playing.hex:i

avrdude: AVR device initialized and ready to accept instructions
avrdude: Device signature = 0x1e9406
avrdude: NOTE: FLASH memory has been specified, an erase cycle will be performed
To disable this feature, specify the -D option.
avrdude: erasing chip
avrdude: reading input file "build-cli/initial_playing.hex"
avrdude: writing flash (4836 bytes):
avrdude: 4836 bytes of flash written

avrdude: safemode: Fuses OK

avrdude done. Thank you.
```

July 5, 2012 at 6:03 AM

watching bird said...

run step 1 and then my browser (midori),crashes when I visit this page from my raspi.

Raspberry with debian and midori as a browser, the log says "out of memory", I have also installed apache, php, Mysql.

Am I the only one with this malfunction?

July 9, 2012 at 9:37 AM

watching bird said...

Unable to finish the tutorial, I run step 1 and then my browser crashes when I visit this page from my raspi.

Raspberry with debian and midori as browser, the log says "out of memory", I have also installed apache, php, Mysql.

Am I the only one with this malfunction?

July 9, 2012 at 9:42 AM

Andre Miller said...

Thank you for this article! It inspired me to try and connect the Raspberry Pi and Arduino using the GPIO UART on the Raspberry Pi and SoftSerial on two other pins on the Arduino. This way you still have the USB available for your serial monitor to do debugging with.

Here is my article: Raspberry Pi and Arduino via GPIO UART

July 10, 2012 at 12:38 PM

scruss said...

Alternatively, you could run Firmata on your Arduino, and use its interface to Python for control. Here's a simple example I put together (with a pretty GUI, too): Raspberry Pi, Python & Arduino *and* a GUI …

August 16, 2012 at 5:06 AM

Paul Ross said...

I can open the USB port apparently, but when I get to the read or write, it complains than the module (serial???) doesn't have an attirbute -- doesn't apparently know how to read or write. I'm following your example, but something seems to be haywire. Thanks! /paul W3FIS

August 16, 2012 at 1:22 PM

Ashman said...

Cant get this to work on my Duemilnove. The led on the Duemilnove flashes when the while loop is running, but no message back to the Pi. Nothing at all happens when I try the ser.write.

any idea's,, hoping I dont need to replace my Duemilnove with an Uno.

thanks

September 14, 2012 at 5:46 AM

Ace said...

Thanks for this article.

But something strange happens to me. The code is workin but when i put it in a file.py and execute that is not working the write('5')

Somebody have an idea what this can be?

Thanks!

September 18, 2012 at 8:40 AM

asleep_in_verona said...

Is there any way to make this script-able?

I'd like to be able to read the Arduino via a Python but I can't get anything that I am sending.

The post mentions you have to press enter twice to see the data. Is there anyway to include this in the script. Also, why is it necessary to press enter twice?

December 18, 2012 at 10:58 AM

asleep_in_verona said...

Is there any way to make this script-able?

I'd like to be able to read the Arduino via a Python but I can't get anything that I am sending.

The post mentions you have to press enter twice to see the data. Is there anyway to include this in the script. Also, why is it necessary to press enter twice?

December 18, 2012 at 11:00 AM

asleep_in_verona said...

Is there any way to make this script-able?

I'd like to be able to read the Arduino via a Python but I can't get anything that I am sending.

The post mentions you have to press enter twice to see the data. Is there anyway to include this in the script. Also, why is it necessary to press enter twice?

December 18, 2012 at 11:00 AM

Michael Horne said...

Thanks for the info Dr Monk. I've got my Leonardo clone sending back serial output and printing that out via Python on the Pi

January 3, 2013 at 2:16 PM

James Phillips said...

My Raspberry PI Doorbell server written with Python, activates randomly - the detection of the button press makes the voltage on the switch go from 1 down to 0 in small increments - the code looks for change rather than either 1 or 0 - is the code at fault or do you think I need a shield - or go for something like This

January 10, 2013 at 2:42 PM

TechMaster said...

Hi, Dr. Monk

Your article was just what I was looking for: I have mounted my Arduino to a 4-wheel chassis, connected my Pi to the Arduino and am now able to control the motors via SSH in a python shell.

Now the next step for me is to make a python script that I can call with an argument so that the argument is sent to the Arduino, so that I do not have to go into a python shell to control it but I have encountered a problem:

When I execute the script, the return value of ser.write() is 1, but the Arduino does not react (although the RX LED blinks).

This is my script so far:

```
#!/usr/bin/python

import sys
import serial

baudrate = 9600
direction = sys.argv[1]

ser = serial.Serial('/dev/ttyUSB1', baudrate)
# Writes 'F' as in Foreward. Works in python shell.
ret = ser.write('F')

# This prints '1'
print(ret)
```

Any ideas as of what might be wrong?

February 3, 2013 at 2:49 AM

Splitlocked said...

Hi, Dr. Monk

Your article was just what I was looking for: I have mounted my Arduino to a 4-wheel chassis, connected my Pi to the Arduino and am now able to control the motors via SSH and a python shell.

Now the next step for me is to make a python script that I can call with an argument so that the argument is sent to the Arduino, so that I do not have to go into a python shell to control it but I have encountered a problem:

When I execute the script, the return value of ser.write() is 1, but the Arduino does not react (although the RX LED blinks).

This is my script so far:

```
#!/usr/bin/python

import serial

baudrate = 9600

ser = serial.Serial('/dev/ttyUSB1', baudrate)
ret = ser.write('B')

# This prints '1'
print(ret)
```

Any ideas as of what might be wrong?

February 3, 2013 at 2:54 AM

Sergio Gordaliza said...

hi, i'm Sergio.
What do you do with this:
(Serial.read() - '0') ?
Thanks and regards from Spain

March 6, 2013 at 8:58 PM

NUTSgoWEEE said...

Hi, thank you so much for posting this. I am having some trouble though on Raspbian.

It installs fine, but there is no "Python 2" under programming.

Any suggestions?
Thanks again!

March 23, 2013 at 3:12 AM

Bruce Fleming said...

Hi. I should remind myself every time I have a problem that I actually know very little.

This article is now in the "Programming the Raspberry Pi" book by Dr. Monk and that book is what I am using to get going with the RPi. I had the following error message when I tried to execute everything in the IDLE editor:
**********8
>>> import serial
>>> ser = serial.Serial('dev/ttyACM0', 9600)

Traceback (most recent call last):
File "", line 1, in
ser = serial.Serial('dev/ttyACM0', 9600)
File "/usr/local/lib/python2.7/dist-packages/serial/serialutil.py", line 260, in __init__
self.open()
File "/usr/local/lib/python2.7/dist-packages/serial/serialposix.py", line 276, in open
raise SerialException("could not open port %s: %s" % (self._port, msg))
SerialException: could not open port dev/ttyACM0: [Errno 2] No such file or directory: 'dev/ttyACM0'
****
I searched the RPi forums with nothing useful to me as I still know little about using Linux. Then I searched Yahoo! using the full error message and came across the following link:
http://www.tracemyworld.com/site/en/support/documentation.html?start=7

I found that using the ls /dev/ACM0 command did not give me what I wanted but then later tried just this:
ls /dev/
which showed me that there was a USB device present where there was none when the Arduino was not plugged in.

At this point, it is helpful to know I am using and Arduino Duemilanove and a Freeduino SB, V 2.I which is pretty much the same thing as the Duemilanove. Once I figured the new USB device listing out, I started experimenting with the code and using two Arduino's. The LEDs blink so quickly so I slowed down the blinking. When I started to enter in numbers with more than 1 digit, I found out the Arduino sketch will blink each digit in series - fun!

Dr. Monk, if you have a forum or other venue for posting the differences that people have with hardware and software, it would be appreciated if you posted a link. I have not found it yet. The RPi forum would be a good place to search for that - I might start a thread.

April 7, 2013 at 1:06 PM

Luigi Liceo said...

Thanks for this article.
Works great!

I'm waiting next chapters of Your book... :-))

Luigi from Italy

July 20, 2013 at 12:17 AM

Leo said...

Is there any way to set a static port name for the arduino or configure so it automatically knows the port name ?

January 7, 2014 at 12:47 PM

Koty Bashford said...

For those having issues making this into a script, I found that you need to issue the following commands

time.sleep(1)
ser.setDTR(level=0)
time.sleep(1)

Reason:
The Arduino resets when a serial connection is opened. So you need to make your script wait until it is finished resetting.

January 10, 2014 at 1:21 PM

Antonia Creswell said...

I am getting:

Permission denied: 'dev/ttyAMAO'

I have read that I can "free my pi serial port" but would like not to have to do this... how can I solve this?

Thank you

March 1, 2014 at 9:09 AM

adithya said...

hi mr.simon
i have a issue with my arduino basically i use a arduino NG with atmega8 controller and for a few months i was able to upload programs but this week it shows soe errors like
Arduino: 1.5.6-r2 (Windows 8), Board: "Arduino NG or older, ATmega8"

In file included from C:\Users\Aditya\arduino1.6\Arduino\hardware\arduino\avr\cores\arduino
\HardwareSerial.cpp:32:
C:\Users\Aditya\arduino1.6\Arduino\hardware\arduino\avr\cores\arduino\HardwareSerial_private.h: In member function 'void HardwareSerial::_rx_complete_irq()':
C:\Users\Aditya\arduino1.6\Arduino\hardware\arduino\avr\cores\arduino\HardwareSerial_private.h:98: error: 'UPE' was not declared in this scope

This report would have more information with
"Show verbose output during compilation"

enabled in File > Preferences.
even if i just hit complie without any program it shows the error
could you please elp me .....
my email id is adithyaad96@gmail.com

April 8, 2014 at 4:15 AM

adithya said...

hi simon i was using my arduino NG with atmega 8 for two months but for a week shows some errors like
Arduino: 1.5.6-r2 (Windows 8), Board: "Arduino NG or older, ATmega8"

In file included from C:\Users\Aditya\arduino1.6\Arduino\hardware\arduino\avr\cores\arduino
\HardwareSerial.cpp:32:
C:\Users\Aditya\arduino1.6\Arduino\hardware\arduino\avr\cores\arduino\HardwareSerial_private.h: In member function 'void HardwareSerial::_rx_complete_irq()':
C:\Users\Aditya\arduino1.6\Arduino\hardware\arduino\avr\cores\arduino\HardwareSerial_private.h:98: error: 'UPE' was not declared in this scope

This report would have more information with
"Show verbose output during compilation"
enabled in File > Preferences.

can you please reply me about the issue please

April 8, 2014 at 4:17 AM

The Ninaad Das Channel said...

Hi Mr.Simon
I really liked this blog post, and I was very happy when I saw the setup working!
but I have one question. Why does the command ser.write('n') doesn't work with more than one digit numbers? Is there any improvisation to the code to fix this output bug?
Thank you
Ninaad Das
from India

January 27, 2015 at 1:41 AM

Post a Comment

Links to this post

Create a Link

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)