

Fascinating

Computer Vision | OpenCV | Linux

Communicating with Arduino using C/C++

Posted on March 8, 2013 by salilkapur

Hello Everyone

This post is on “How to communicate with Arduino using C/C++”.

I basically have an array of integers from my C/C++ code which i need to send to an Arduino. I need to communicate between C/C++ and Arduino

Before we actually see how this is done, we should know a little about how are actually devices handled in Linux.

Under Linux and UNIX each and every hardware device is treated as a file. They are special files located in the **/dev** folder in your system which allow us to access the hardware directly.

Like any other normal file in our computer we can read/write to these device files(which means reading or writing to the device itself). That is pretty much how the OS also does it on a primitive level. It opens the file associated with the device it has to use and then reads or writes to it accordingly.

Arduino boards are also handled in the same way. We will read/write to the device file of our Arduino.

First, we need to identify the device file for our arduino.

- Plug in your arduino, and open the Arduino Software (<http://arduino.cc/en/Main/Software>).
- Go to the tools menu and look for Serial Port submenu.
- There it will show you the list of device files(Serial Ports). For Arduino Duemilanove it is **/dev/ttyUSB0**(file name can change if it is already in use), for Arduino UNO it will be **/dev/ttyACM0**. Select a file from this menu which will be used for communication.

Now, there are 2 parts to this problem, the C code and the Arduino code.

- C code will send data
- Arduino code will receive data.

C code

```

1  int main()
2  {
3      int data[] = {10,5,13}; //Random data we want to send
4      FILE *file;
5      file = fopen("/dev/ttyUSB0","w"); //Opening device file
6      int i = 0;
7      for(i = 0 ; i < 3 ; i++)
8      {
9          fprintf(file,"%d",data[i]); //Writing to the file
10         fprintf(file,"%c",','); //To separate digits
11         sleep(1);
12     }
13     fclose(file);
14 }

```

Explanation.

- We have an array of integers and we want to send this array to our arduino. First we open our arduino device file which is `/dev/ttyUSB0`.

```

1  int data[] = {10,5,13}; //Random data we want to send
2  FILE *file;
3  file = fopen("/dev/ttyUSB0","w"); //Opening device file

```

- Next we write each element of the array to the file using a simple **for** loop, after writing each element we add a *comma*(`,`) to separate elements in our file.

```

1  int i = 0;
2  for(i = 0 ; i < 3 ; i++)
3  {
4      fprintf(file,"%d",data[i]); //Writing to the file
5      fprintf(file,"%c",','); //To separate digits
6      sleep(1);
7  }
8  fclose(file);

```

Next thing is to read this data through an arduino.

Arduino Code

```

1  void setup()
2  {
3      Serial.begin(9600);
4  }
5
6  int calc()
7  {
8      int num=0,x=0;
9
10     for(x;x<=j;x++)
11         num=num+(buff[x]-48)*pow(10,j-x);
12
13     return num;
14 }
15
16 int input,num;

```

```

17 | int buff[5];
18 | int j=-1;
19 |
20 | void loop()
21 | {
22 |     if(Serial.available()>0)
23 |     {
24 |         input=Serial.read();
25 |
26 |         if(input==',')
27 |         {
28 |             num=calc();
29 |             j=-1;
30 |             Serial.println(num);
31 |         }
32 |         else
33 |         {
34 |             j++;
35 |             buff[j]=input;
36 |         }
37 |     }
38 | }

```

Explanation

- First we setup the Serial port in **void setup()**(line 3).
- After setting it up, we have to start reading, now in our C code we wrote integer values to our file

```
1 | fprintf(file, "%d", data[i]);
```

But our arduino will read them as characters, so we need to read characters and convert them back to our original number.

- So in **void loop()** we keep on reading data(which is nothing but our integers converted to characters) and store them in an array(**buff**) until we read a *comma* (','). *comma* was used to separate digits(line 10 of our C code).

```

1 | input=Serial.read();
2 |
3 | if(input==',')
4 | {
5 |     num=calc();
6 |     j=-1;
7 |     Serial.println(num);
8 | }
9 | else
10 | {
11 |     j++;
12 |     buff[j]=input;
13 | }

```

As soon as we read a *comma* we call **calc()** where we convert the read characters back to the integer.

- In **calc()** we subtract 48 from the ascii value of the character, multiply it with its *place value raised to 10* and return the final integer **num**.

```
1 | for (x; x<=j; x++)  
2 |     num=num+ (buff[x]-48) *pow(10, j-x) ;
```

- Repeat the process until all the data has been read.

Finally, we were able to send an array from C/C++ to our arduino.

This was a very basic way of communicating, many better techniques can also be used.

You May Like [About these ads \(http://en.wordpress.com/about-these-ads/\)](http://en.wordpress.com/about-these-ads/)

- 1.



This entry was posted in [Arduino](#), [Linux](#) and tagged [Arduino](#), [C/C++](#), [Linux](#), [serial](#), [serial communication](#), [serial ports](#). Bookmark the [permalink](#).

24 thoughts on “Communicating with Arduino using C/C++”

Faisal says:

on July 25, 2013 at 13:09

Hi,

I am trying to implement your code in windows machine but its not working. The PC cannot write to arduino. In serial monitor of arduino I couldn't see any output.

I have changed your code in one line only

```
file = fopen(“COM10”,”w”);
```

As arduino in Com10 so I have done that. If you have any suggestions and help or any code please help and suggest.

Reply

salilkapur says:

on July 25, 2013 at 14:36

Hi

The post was written keeping in mind a Linux Platform. So i guess the code will not be directly usable on a windows machine as windows treats devices differently.

I can help you using python.
You can do something like this

```
import serial  
  
s = serial.Serial("COM10",9600)  
s.write('hello')
```

I think this should work.

Salil

Reply

Faisal says:

on July 25, 2013 at 15:19

Thanks for your prompt reply. I will use python and lets see what will happen. I will give you update.

Reply

Faisal says:

on July 26, 2013 at 02:46

I just have small query to you. Simultaneously PC and arduino can not access the same port at the same time. At the same time PC or arduino can get access the port. So when you are writing to the port how can you see the output in the serial monitor [arduino].

Reply

salilkapur says:

on July 26, 2013 at 15:01

Hi

To answer your question, it appears that the operations are taking place simultaneously but actually they are consecutive operations. The PC first writes to the port and then the arduino reads the data.

Both arduino and PC try to gain access to the port. So lets say if the PC is granted access to the port, the PC puts a write lock on the port. This prevents the arduino from reading the data while the PC is trying to write data. Once the writing operation is complete the PC releases the write lock and then the arduino puts a read lock so that the PC cannot write again to the port while the arduino is trying to read data. It all happens one by one.

I hope this answers your query.

Salil

Reply

Faisal says:

on July 26, 2013 at 15:25

Thanks. I understood your reply. However I am wondering after PC writing data to the port will arduino get data to read from port ?

If you please give your step to watch the output. I am doing the following step.

1. Running the program in EDITOR to write data to port.
2. Closing the program

3. Watch the Serial monitor of Arduino [Program already uploaded to arduino board].

If you have any suggestion regarding the step please suggest.

Reply

salilkapur says:

on July 26, 2013 at 18:07

I guess you are trying to catch what you have send using your python/C code on the serial monitor.

- 1) So first run your C/python code and send some data.
- 2) Then in your arduino code read the serial data and print it on the monitor using `println()` and open the serial monitor.

This should work.

Reply

Faisal says:

on July 27, 2013 at 14:31

Thanks for your help and suggestion. Problem is its not working from windows machine. I have to try Linux machine.

<http://playground.arduino.cc/interfacing/python> this link ensure about it. However as far their suggestion I have added timer but not working 😞

Reply

antonio alves says:

on August 7, 2013 at 23:31

Very simple and clear. It worked at first, but I added the arduino the first line: `# include` , with arduino uno and linux raspberian.

Many thanks and good job.

Reply

salilkapur says:

on August 8, 2013 at 14:31

I'm glad you like it! 😊

Reply

michael says:

on September 6, 2013 at 14:11

hi there,

thanks for this post. I was wonering how i could read information being sent from the arduino to the pc in C/C++.

cheers,

Reply

salilkapur says:

on September 17, 2013 at 12:58

Hi

Sorry for the late reply.

So for sending information back to your computer, you first send the data from your arduino to the serial port and then in your C/C++ code you can read that data from the device file(Serial Port)

So to send data to the serial port from the arduino you can use Serial.write().
and then in your C/C++ code open the device file(i.e /dev/ttyUSB0) and read data from this file.
(Synchronizing read and write can be a bit tedious here)

Essentially all the information exchange happens using the device files.

I hope this helps.

Salil

Reply

Mike says:

on October 23, 2013 at 01:11

Firstly, Thank you very much for this tutorial. It is very helpful, I was able to implement it in my project. However, it seems like there is a ten second delay between when I write the value and when the arduino reacts to it. Do you have any ideas what would cause this kind of lag?

Reply

salilkapur says:

on October 24, 2013 at 18:29

Hi,

Can you show me your code? I am not sure what could cause this delay.

Salil

Reply

lionel says:

on March 15, 2014 at 00:16

Hello,

thank you for this post but how can i know that arduino receive data.

How can i read data sending by arduino ?

Thanks

Reply

Riemeo says:

on March 21, 2014 at 11:08

Hi

Thanks you for this code

I have some question

I coding in Xcode(Mac os)

this code be occurred error in 'sleep'

error Content is "Use of undeclared identifier 'sleep' "

Reply

salilkapur says:

on March 21, 2014 at 11:15

Hi

I haven't tried this on Macintosh. I guess you will have to look this over the internet.

Reply

lakshay nagpal says:

on March 26, 2014 at 20:45

hey!

Please can you help me out to write a c++ code of the given arduino code

please reply me on my given e-mail and i will send you the required condition with the arduino file included.

Reply

salilkapur says:

on March 26, 2014 at 21:03

Hi,

I am assuming you are looking for the C++ version of the C code mentioned in the blog post. You can use the given C code directly in your C++ file. Just copy the code in any .cpp file and compile it using g++ and it should work.

I hope I have answered your query. Tell me if you were looking for something else.

Salil

Reply

lakshay nagpal says:

on March 27, 2014 at 11:14

no sir, the arduino code is different and i am asking for the c++ code of the arduino that i m having which is different from the code that you have provided above.

i understood your's arduino's code but i'm stucked in a different one

so please can you help me out ?

Reply

Zahir says:

on May 15, 2014 at 18:23

Good day,

I have been dabbling with the Arduino, but not having a good grasp of C++, it is an uphill battle.

I'm trying to read value from an MS excel spreadsheet, to the Arduino, and compare it with values read into an analogue port of the Arduino. If higher, I want to light a LED – if lower, a second LED and if equal, a third LED.

These inputs read at the analog ports, must also be written into the same excel spreadsheet (different column) so that a graph can be generated by live if necessary on excel.

It is purely for experimental purposes.

So, basically, I want to read and write to and from excel – draw a graph and light three LED's dependent on the comparative values.

Any great ideas??

Regards

Z

Reply

salilkapur says:

on May 16, 2014 at 18:30

Hi Zahir,

All you need is a library that can help you interface with Excel. Now there are many libraries available for Java and Python. I am not sure how easy it is to use C/C++ for such a task. I would suggest you to use Python because it will make it easier for you on both fronts Excel interfacing as well as communicating with the Arduino. You can use <http://www.python-excel.org/> for the excel part and for the arduino all you have to do is again open the device file and read/write values to/from it. It is exactly how I have done using C/C++.

I hope this helps! Leave a comment otherwise.

Reply

Zahir says:

on May 19, 2014 at 11:10

Thanks... back to the drawing board for me. My son has been urging me to use Python as well. The reason for me going to C++, was mainly because I thought it was a good way to ease writing commands to PIC devices – I have in the past used assembler to program these and I have recently bought a book that starts with the words “A good understanding of C++ is required”.

At my age, a few years from 60, I still pride myself knowing valve theory!!

I have downloaded PLX over the weekend, this enabled me to write to an excel file – storing analog values read from the Arduino to excel drawing a graph live. Great to use as a logging device, but, this does half the job reading from Arduino – as stated, I need a two way comms live.

Maybe I must overcome my fear of snakes and tackle the PYTHON!

Cheers

Z

salilkapur says:

on May 19, 2014 at 11:50

Hi Zahir,

After looking up over the internet, I found this library “libxl” <http://www.libxl.com/>

This might be the thing for you. The library has support for both C/C++ just in case you want to shift to C. I guess this can help in all your tasks. PIC, Excel, Arduino.

PS:- This PYTHON isn't that deadly, give it a try 😊

PPS:- It is inspiring, how you are still so enthusiastic 😊

Cheers

Salil

Reply

[Blog at WordPress.com.](#) / [The Truly Minimal Theme.](#)

Follow

Follow “Fascinating”

Powered by [WordPress.com](#)