How to work with or reset the Interlock Arduino

log in on the little white netbook called **minime** with:

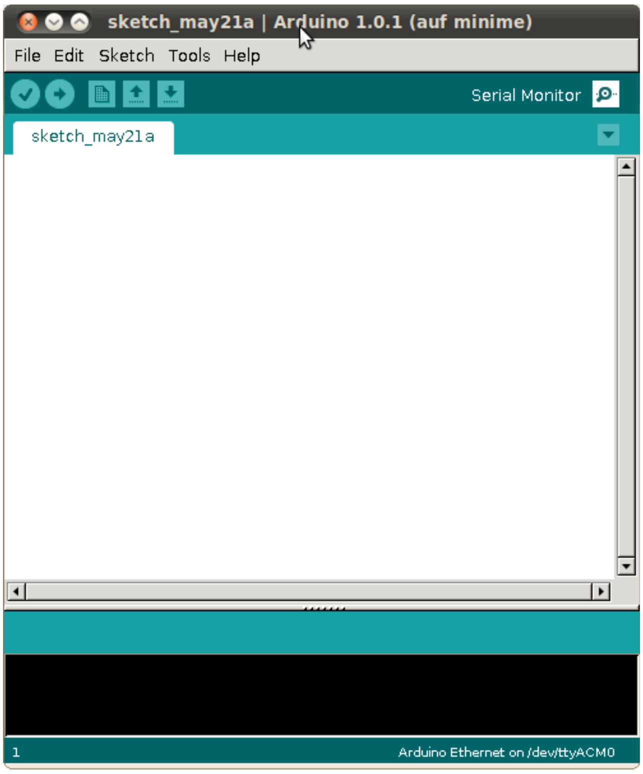
username **arduino**

password **arduino**

In order to work with the Arduino, like uploading new firmware or just listening to its output on its debugging console, you may use the graphical development suite. The software is installed in:

```
/home/arduino/Software/arduino-1.0.1
```

just call the binary called **arduino** to start the IDE. It will like roughly like this:


sketch_may21a | Arduino 1.0.1 (auf minime)
File Edit Sketch Tools Help
Serial Monitor
sketch_may21a
1
Arduino Ethernet on /dev/ttyACM0

Now you could load a source code file, compile it and upload it, but maybe you first want to listen to one of the **two arduino boards** which are currently connected to this netbook first.

Which Serial Port?

First you need to define, which arduino you want to work with, in order to do that, choose one of the **Serial Ports** available in the **Tools** menu.

Currently your choices are:

Tools → Serial Ports → /dev/ttyACM0 (**currently** the interlock arduino)

Tools → Serial Ports → /dev/ttyACM1 (**currently** the lid arduino)

Do not rely on this mapping. When this computer is rebooted or both arduinos get disconnected and then reconnected, the timing on the reconnection might define which one shows up on ttyACM0 and ttyACM1.

One can rely on the serial number of the FTDI chips, which you can get for example like this:

```
[arduino@minime ~]$ ll /dev/serial/by-id/
total 0
lrwxrwxrwx. 1 root root 13 May 17 07:47 usb-Arduino__www.arduino.cc__Arduino_USB-Serial_64936333137351315172-if00 -> ../../ttyACM0
lrwxrwxrwx. 1 root root 13 May 17 07:45 usb-Arduino__www.arduino.cc__Arduino_USB-Serial_6493633313735151B0D1-if00 -> ../../ttyACM1
```

As one can see from this listing the serial number of the FTDI chip in the interlock arduino is 64936333137351315172.

quick solution

However there is a quicker method, to find out which one is which. Simply choose one and listen to it. But be aware, when ever you connect to the arduino using the "Serial Monitor", the arduino software automatically and unavoidably resets the arduino. So in case you are currently testing which arduino is which, you are resetting both of them. This should be no problem for any of the two arduinos. However in case you are doing this while the lid arduino is currently moving the lids, then the lids will stop in mid movement. And I don't know how exactly the lid arduino behaves after reboot, it might automatically close both lids, but I do not think this is very likely.

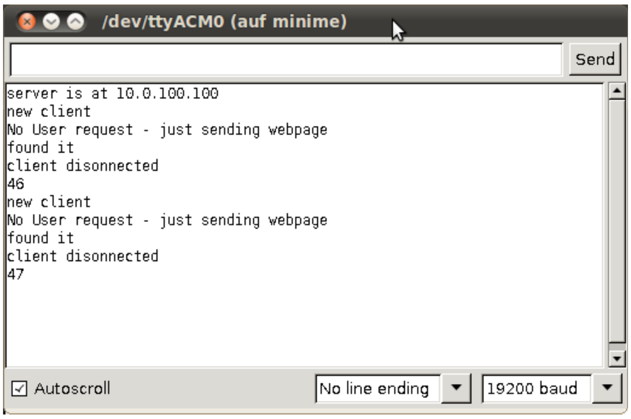
Listening to an arduinos serial port

Simply click on the Serial Monitor button in the arduino user interface and choose the right baud rate. These are currently:

interlock arduino: 19200 baud

lid arduino: 115200 baud

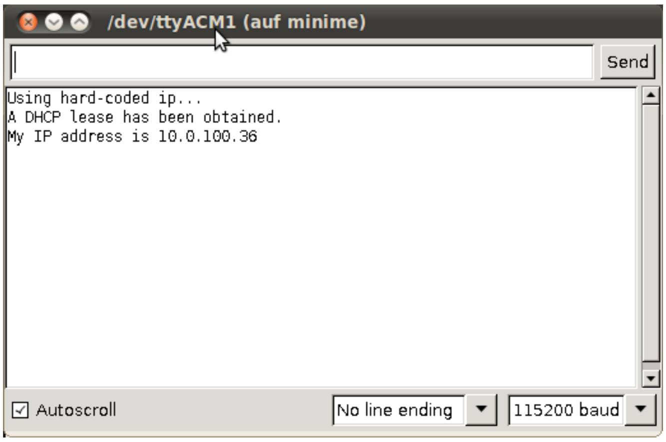
The output is visible in a newly opened window and might look like this:



```
server is at 10.0.100.100
new client
No User request - just sending webpage
found it
client disonnected
46
new client
No User request - just sending webpage
found it
client disonnected
47
```

☑ Autoscroll No line ending 19200 baud

Since this is only debugging output, the entire text might make no sense to the non-developing physicist, but the first line tells you, that the entire thing was just resetted and the server started up, using this IP.For the lid arduino this window looks like this:



```
Using hard-coded ip...
A DHCP lease has been obtained.
My IP address is 10.0.100.36
```

☑ Autoscroll No line ending 115200 baud

Resetting an arduino using the arduino software

Simply connect to the arduino you want to reset via the Serial Monitor and listen to it.

In the moment you click the **Serial Monitor** button on the upper right of the arduino software window. The software will send a reset signal to the arduino and then open the monitoring window

Resetting the USB controller of minime

In some cases one can't connect to the arduino via its serial console. Then it might help to reset the USB controller of the connected PC. In case of minime, which is running a Fedora linux, it can be done like this.

Be sure to execute this as root, e.g. by typing **sudo su**.

```
cd /sys/bus/pci/drivers/uhci_hcd
```

you should find symbolic links in this folder, named:

0000:00:1d.0

0000:00:1d.1

0000:00:1d.2

0000:00:1d.3

Now you write the names of these links into 'unbind' and then into 'bind'.

E.g. like this:

```
echo -n 0000:00:1d.0 > unbind
```

then

```
echo -n 0000:00:1d.0 > bind
```

repeat it for the other links: 0000:00:1d.1, 0000:00:1d.2, 0000:00:1d.3

It is perfectly fine, to prepair a bash script for this, but I currently have no time... sorry.