

Serial port Com with Python



- Amongst the many possibilities of Python, communication with the Serial (COM) port of the computer where it is running ranks very high.
- The key library is pySerial, see for details:
 - <https://pyserial.readthedocs.org/en/latest/index.html>
- An important introductory example is to detect the available serial ports, on Python prompt enter: (pySerial needs to be installed)
 - `>> from serial.tools import list_ports`
 - `>> list_ports.comports()`
 - Python OUT `>>`
 - `[['/dev/cu.Nokia206-COM1', 'n/a', 'n/a'],`
 - `[ '/dev/cu.Bluetooth-Modem', 'n/a', 'n/a'],`
 - `[ '/dev/cu.usbmodem411', 'Arduino Due Prog. Port', 'USB VID:PID=2341:3d`
`SNR=95238343334351C02112' ]]`

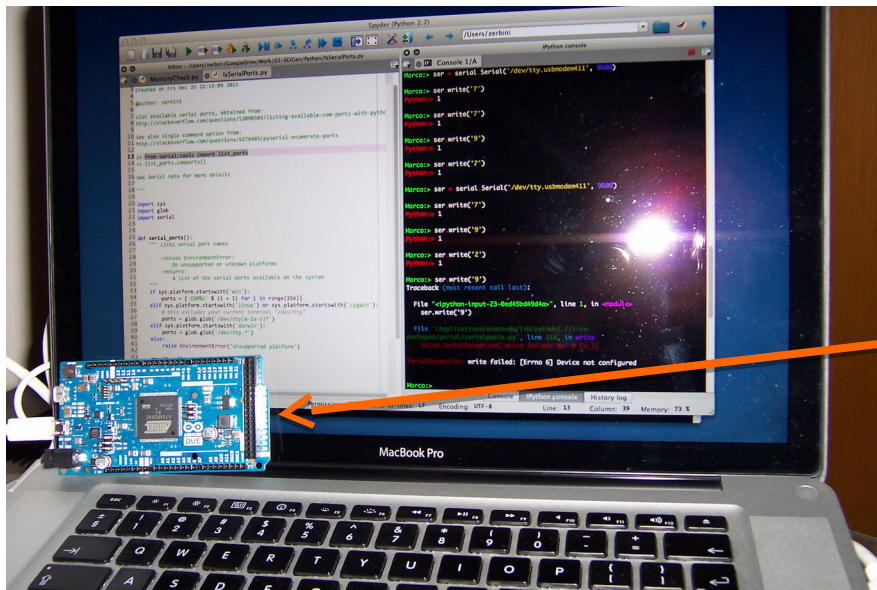
In the example an Arduino DUE is connected to port 411. The example is described in:

<http://stackoverflow.com/questions/12254378/how-to-find-the-serial-port-number-on-mac-os-x>
<http://stackoverflow.com/questions/12254378/how-to-find-the-serial-port-number-on-mac-os-x>



Python Serial communication, cont.

- Next step is to use Python to control a device, see: www.akeric.com/blog/?p=1140
- Create and upload the Arduino Module SerialPythonRX, as in the webpage.
- Connect Arduino TM to the computer via USB and Detect the name of serial port
- Open a Python session and enter:
 - `>>> import serial`
 - `>>> ser = serial.Serial('/dev/tty.usbmodem411', 9600)`
 - `>>> ser.write('5')`
- the LED connected to pin 13 will blink 5 times. Make sure to use correct Serial port name.



Arduino is a programmable prototyping board based on a ATmega processor, connected via USB to the control computer.