playground.arduino.cc

Arduino Playground - Python

Arduino and Python

Talking to Arduino over a serial interface is pretty trivial in Python. On Unix-like systems you can read and write to the serial device as if it were a file, but there is also a wrapper library called pySerial that works well across all operating systems.

After installing pySerial, reading data from Arduino is straightforward:

```
>>> import serial
>>> ser = serial.Serial('/dev/tty.usbserial', 9600)
>>> while True:
...     print ser.readline()
'1 Hello world!\r\n'
'2 Hello world!\r\n'
'3 Hello world!\r\n'
```

Writing data to Arduino is easy too (the following applies to Python 2.x):

```
>>> import serial # if you have not already done so
>>> ser = serial.Serial('/dev/tty.usbserial', 9600)
>>> ser.write('5')
```

In Python 3.x the strings are Unicode by default. When sending data to Arduino, they have to be converted to bytes. This can be done by prefixing the string with b:

```
>>> ser.write(b'5') # prefix b is required for Python 3.x, optional for Python 2.x
```

Note that you will need to connect to the same device that you connect to from within the Arduino development environment. I created a symlink between the longer-winded device name and `/dev/tty.usbserial` to cut down on keystrokes.

It is worth noting that the example above will not work on a Windows machine; the Arduino serial device takes some time to load, and when a serial connection is established it resets the Arduino.

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.

Python Arduino Prototyping API v2

An updated version of the Arduino Prototyping API library used to quickly prototype an Arduino based application in Python.

Instrumentino

Instrumentino is an open-source modular graphical user interface framework for controlling Arduino based experimental instruments. It expands the control capability of Arduino by allowing instruments builders to easily create a custom user interface program running on an attached personal computer. It enables the definition of operation sequences and their automated running without user intervention. Acquired experimental data and a usage log are automatically saved on the computer for further processing. The use of the programming language Python also allows easy extension. Complex devices, which are difficult to control using an Arduino, may be integrated as well by incorporating third party application programming interfaces into the Instrumentino framework.

Links:
* Official page
* Release article
* Package in PyPi
* Code in GitHub

Python Firmata

A Python API for the Firmata protocol. It's based on the Processing Library for Arduino and supports Firmata v2.0

---

pyFirmata

Python interface for the Firmata protocol. It is compliant with Firmata 2.1.

- Documentation
- Examples (by Fabian)

(gui can be build with gtk, not included)

Py2B

Implements a simple 2 ASCII Chars protocol.
To start playing with Python and Arduino go to Py2B

---

arduino_serial.py is a library and command line that allows you to communicate with an Arduino board without installing any additional Python modules.

Turn your Arduino into an IRC bot with python.

Control Arduino using HTML forms via Python.

Arduino-Python 4-Axis Servo Control: with a good general explanation of how to coordinate Arduino and Python code.

Here is an Arduino serial utility for BSDish systems that is written in python.

Control Arduino using this Python-SimpleMessageSystem library.

Realtime graphing of data over serial

Graphing realtime data from an Arduino using Python and Matplotlib:
http://www.blendedtechnologies.com/realtime-plot-of-arduino-serial-data-using-python

A threaded GUI example with Python and Qt/PyQt

A simple example/tutorial of a threaded GUI displaying information from the Arduino.

http://blog.wickeddevice.com/?p=191

A simple UI with Python and GTK

Arduino LED commander is a simple GUI for controlling an output pin of the Arduino. This project was made as a little tutorial/example to show the possibility of controlling an Arduino board with Python and a GTK interface.

- Details: http://fabian-affolter.ch/blog/index.php/arduino-controlled-by-a-gtk
- Source: http://www.gitorious.org/arduino-led
- Video: http://youtu.be/WZ9FKHfVKRY

PyCmdMessenger

PyCmdMessenger https://github.com/harmsm/PyCmdMessenger is a simple, but powerful, interface for reliably sending commands and data to and from an Arduino over a serial connection using Python3. The user should use CmdMessenger http://playground.arduino.cc/Code/CmdMessenger define commands and callbacks on the Arduino. These can then be accessed by PyCmdMessenger using a clean API.