**Requirements**

Software:

* Arduino ide <https://www.arduino.cc/download_handler.php?f=/arduino-1.8.2-windows.exe>
* Miniconda Python2.7 <https://conda.io/miniconda.html>
* Custom arduino sketches and python software. <https://drive.google.com/a/sheffield.ac.uk/file/d/0ByETnnBqkEIXTlNMZHVUYlZGSWM/view?usp=drivesdk>

Hardware

* Arduino uno/mega <http://uk.rs-online.com/web/p/processor-microcontroller-development-kits/7154081/> (2 for monitor+controller configuration)
* AD5171 digital potentiometer <http://uk.rs-online.com/web/p/digital-potentiometers/8099301/>
* Surface Mount (SMT) Board SOT Epoxy Glass Double-Sided 23.5 x 13.5 x 1.5mm FR4 <https://uk.rs-online.com/web/p/surface-mount-smt-to-through-hole-adapter-boards/7288838/?searchTerm=re906&relevancy-data=636F3D3126696E3D4931384E53656172636847656E65726963266C753D656E266D6D3D6D61746368616C6C7061727469616C26706D3D5E2E2A2426706F3D31333326736E3D592673743D43415443485F414C4C5F44454641554C542673633D592677633D4E4F4E45267573743D7265393036267374613D726539303626&dym=re90>
* RCD24 LED driver <http://uk.rs-online.com/web/p/led-drivers/6689882/>
* 2X 4.7 KOhm resistors
* More material: see table at the end.

Arduino Sketches:

* *Incubator firmware*: sketch that doesn’t require a PC running or the pyRhodopsin program (the Arduino can be connected to a 5V power supply for smartphone for power). The parameters have to be hard coded in the sketch before uploading it to the Arduino (see below)
* *BasalSerial4-DigitalPotramps*: this program is used in conjunction with the pyRhodopsin python program which is used to set the parameters from the pc.
* *BasalSerial5-DigitalPotramps*: same as the previous one but with random spiking.
* Counter (optional): this has to be installed on a second Arduino that is connected to a PC. It works together with the pyMonitor python program and records in a file the timing of each spiking.

Python Software:

* pyRhodopsin.py: program used to set the parameters (duration/ power) Used together with the BasalSerial4 arduino sketch
* pyMonitor.py (optional): program used to interface with the second Arduino for monitoring the spike timing. The Counter sketch has to be installed on the second Arduino.

**Software installation**

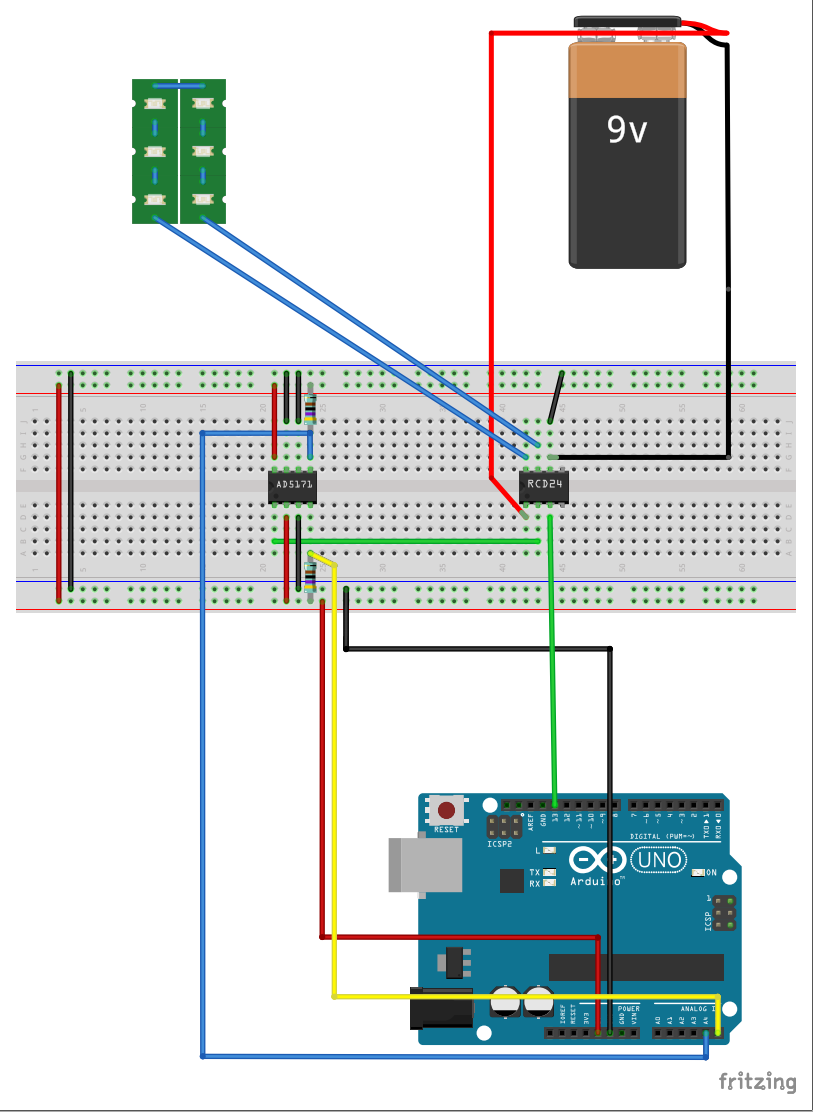
Arduino

1. Install Arduino ide.
2. Connect the Arduino uno to the pc using a USB cable.
3. Open Arduino ide and load one of the Sketches (Incubator firmware, BasalSerial4 or 5).
4. For the incubator firmware (which doesn’t require a computer for setting the parameters), modify these values if required:
   1. Isi: interspike interval in ms
   2. Duration: duration of the main light step in ms
   3. potValue: amplitude of the main Step (values from 0 to 63. 0 means LED fully on, 63 fully off)
   4. potValuePretrigger: amplitude of the pretrigger Step (values from 0 to 63. 0 means LED fully on, 63 fully off)
   5. pretriggerDuration: duration of the pretrigger light step in ms
5. In Tools, select the right Board (Arduino uno) and Port (COMxx)
6. Select Sketch>Upload. The sketch is now loaded on the Arduino.
7. If using the second Arduino for counting the spikes, connect the second Arduino to the PC, select it from the Tools menu and install the Counter sketch using the same steps.

Python:

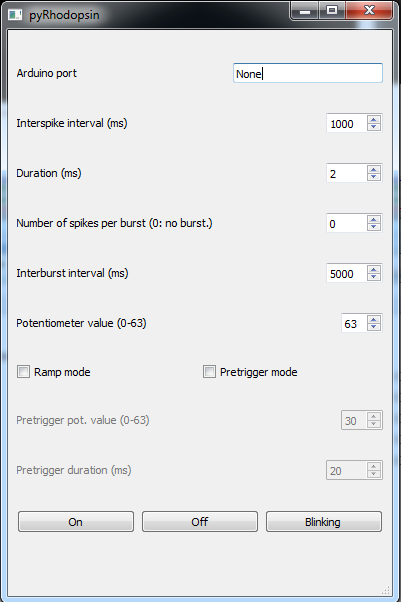
1. Install miniconda
2. Open a terminal (Start> write “cmd.exe”)
3. Create a conda environment named “imaging” *conda create –n imaging numpy*
4. Activate the environment *activate imaging*
5. Install the required packages. *conda install numpy scipy pyqt=4 matplotlib pyserial ipython*
6. Other packages may be required. Install with *conda install nameofthepackage.*

Schematics:



* The 9V battery is the 24V power supply…
* Pin number 13 and 3 on the Arduino are the ones that switch the LEDs on and off (13 is used here, but either can be used).
* (optional) If a second Arduino is used for monitoring spikes, connect pin number 2 of the second Arduino to pin number 13 (or 3) of the main Arduino. Also connect one of the ground GND pins of the second Arduino with the ground of the first.
* NOTE: the blue and yellow wires attached to the A4 and A5 input of the Arduino are for the SDA/SCL I2C protocol, which is the serial protocol that is used to change the output power to the LEDs. These ports are different in other version of the Arduino board (for example they are associated with pins 20 and 21 in the Arduino MEGA). Look for the *Wire* reference for your Arduino board. https://www.arduino.cc/en/reference/wire

**pyRhodopsin program**

* Install BasalSerial4-DigitalPotramps or 5 in the Arduino using the above steps. Connect the Arduino to the PC
* Using the Arduino IDE check which port (COMxx) the arduino is attached to(Tools/port)
* Navigate to the pyRhodopsin folder (where the pyRhodopsin.py file is)
* Shift + right click inside the folder > Open command window here (alternative: open a cmd.exe window, navigate to the folder using the *cd* command).
* Type *activate imaging* . You should see the line starting with [imaging]
* Type *python pyRhodopsin.py*
* 
* Arduino port: write the name of the port (COMxx). The LEDs (and the LED on the Arduino) should blink when the connection is successfull
* All the other variables are self-explanatory. Potentiometer value (i.e. led power), goes from 0, led off, to 63 full power (note that this is the opposite of the Incubator firmware). Ramp mode increases the power gradually up to the pretrigger pot value, “Pretrigger mode” increases the power in a single step before the main step. Number of spikes per burst: set to 0 for continuous spiking

**pyMonitor software (optional)**

* Connect the second Arduino to the pc and install the Counter sketch using the Arduino IDE (see above)
* Write down the port of this Arduino (COMxx)
* Open a terminal in the folder containing the pyMonitor.py file (shift+right click in the folder, open command window here)
* Open pymonitor.py with a text editor and change the name of the serial port (COMxx) if required. Save and close.
* Type *activate imaging* . You should see the line starting with [imaging]
* Type *python pyMonitor.py*
* Leave the command window open. The program records the timing of each spike and write the isi in a text file called file.txt. This file is overwritten every time the pyMonitor program is restarted





**Shopping List**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| item | Quantity | link | Unitary price | price |
| Royal blue LED | 6 for each heatsink | <http://www.luxeonstar.com/royal-blue-20mm-star-led-910mw> | 11.43$ |  |
| Blue LED (470, more suitable for CHr) | 6 for each heatsink | <http://www.luxeonstar.com/blue-470nm-20mm-star-led-70lm> | 9.84$ |  |
| Thermal Adhesive Tape for 20 mm Star LED Assemblies | 2 | <http://www.luxeonstar.com/pre-cut-thermal-tape-for-20mm-hex-bases-12pcs> | 7.49$ |  |
| Carclo 20 mm Black Hex Optic Holder | 30 | <http://www.luxeonstar.com/carclo-20mm-hex-optic-holder-black> | 0.5$ |  |
| Standard extruded heatsinks | 5-6 | http://uk.rs-online.com/web/p/heatsinks/1694252/ | 11.37£ |  |
| LED Driver | 5 | <http://uk.rs-online.com/web/p/led-drivers/7063238/> | 14.10£ |  |
| Binder 2 Pole Cable Mount Connector Socket, male Contacts | 5 | <http://uk.rs-online.com/web/p/industrial-automation-circular-connectors/1296460/> | 4.91£ |  |
| Binder 2 Pole Cable Mount Connector Socket, Female Contacts | 5 | <http://uk.rs-online.com/web/p/industrial-automation-circular-connectors/0469125/> | 5.38£ |  |
| Led Lenses  (any CARCLO lens can be used depending on the application) | 7 packs of 3 | <http://uk.rs-online.com/web/p/led-lenses/6974272/> | 4.38£ per pack |  |
| 24V power supply | 1 | <http://www.mascot.no/catalog/8921/c-24/p-120> | ? |  |
| Junction boxes | 2 packs of three | <http://uk.rs-online.com/web/p/junction-boxes/7000811/> | 22.179 per pack |  |