DiY biosignals

let's get to work

Digital-to-Analog Converter (DAC)

LED
video camera sync

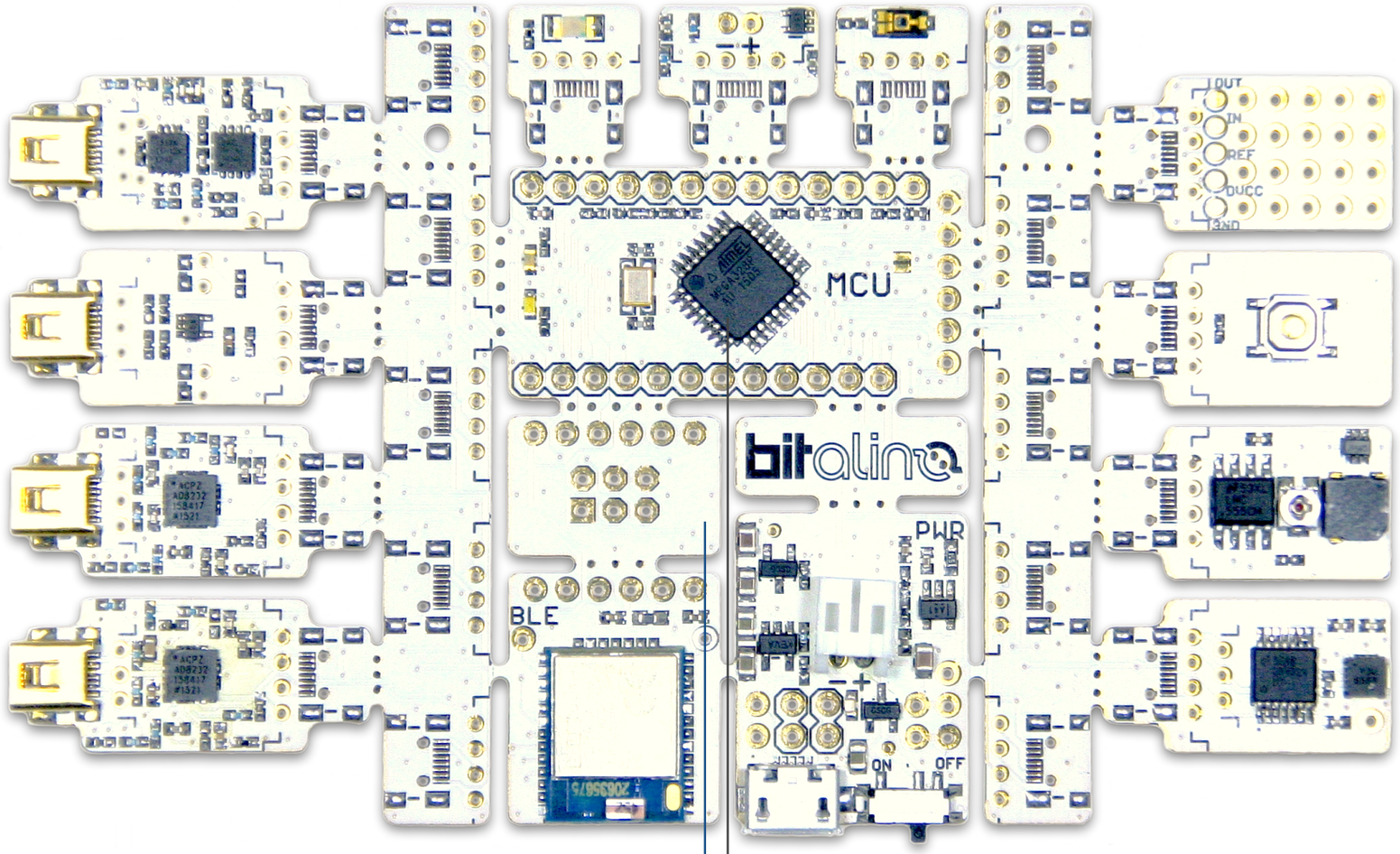
Light Sensor (LUX)
computer screen sync

Electroencephalography (EEG)
brain activity

Electrodermal Activity (EDA)
sympathetic nervous system

Electrocardiography (ECG)
cardiac activity

Electromyography (EMG)
muscle activity



Prototyping Board (PRT)
interface for custom accessories

Pushbutton (BTN)
manual event annotation

Buzzer (BUZ)
audio feedback

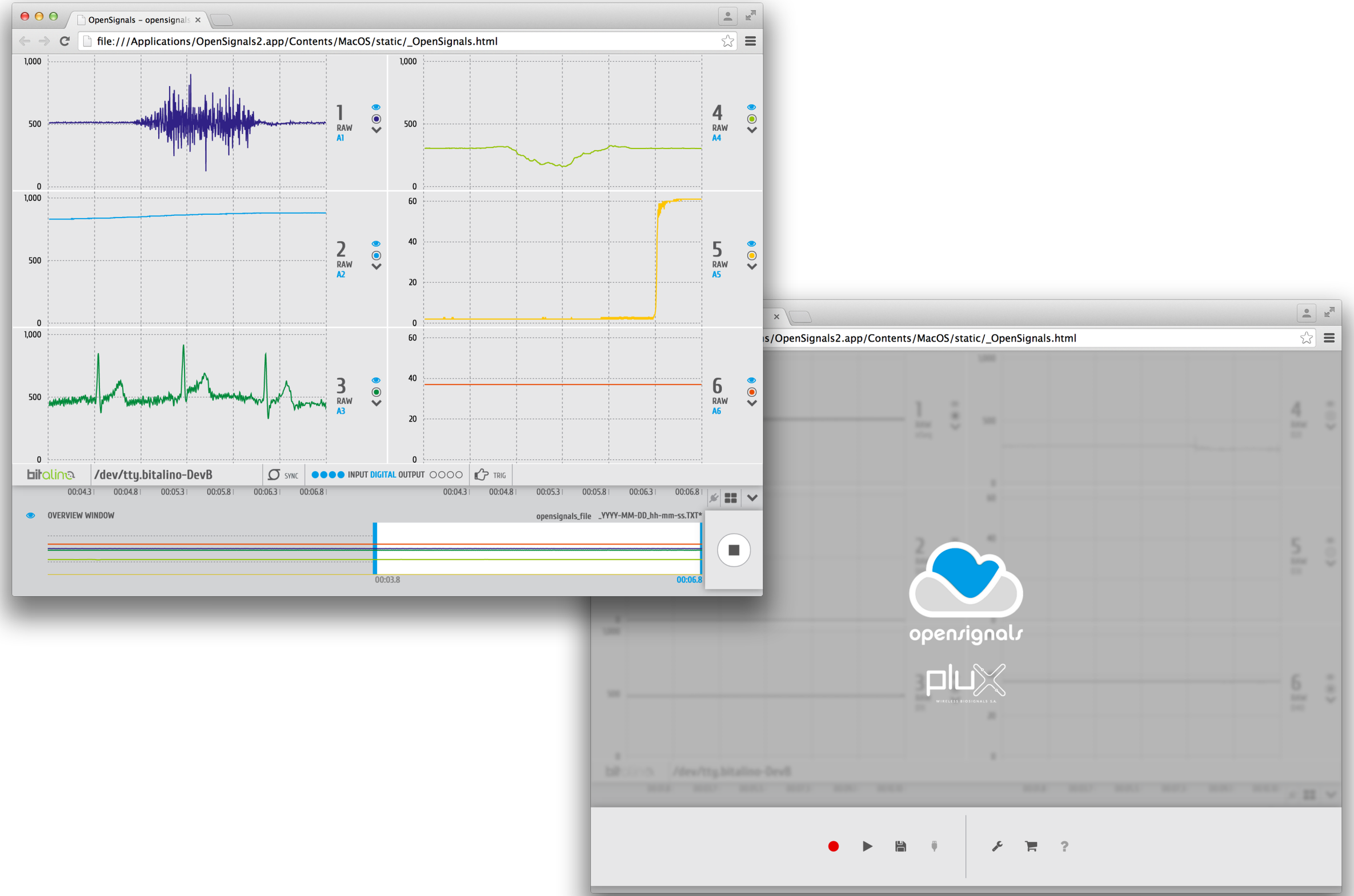
Accelerometer (ACC)
motion

BLE
Bluetooth also available

Power
3.7V Li-Po w/ charger

AVR ISP 6-pin header

Micro-controller unit (MCU)


OpenSignals – opensignals
file:///Applications/OpenSignals2.app/Contents/MacOS/static/_OpenSignals.html
1,000
500
0
1
RAW
A1
2
RAW
A2
3
RAW
A3
4
RAW
A4
60
40
20
5
RAW
A5
6
RAW
A6
bitalino
/dev/tty.bitalino-DevB
SYNC
INPUT DIGITAL OUTPUT
TRIG
00:04.3
00:04.8
00:05.3
00:05.8
00:06.3
00:06.8
OVERVIEW WINDOW
opensignals_file _YYYY-MM-DD_hh-mm-ss.TXT*
00:03.8
00:06.8
/OpenSignals2.app/Contents/MacOS/static/_OpenSignals.html
opensignals
plux
WIRELESS BIOSIGNALS S.A.

LoadFile.py

LoadFile.py 〉 No Selection

```
1  # -*- coding: utf-8 -*-
2  from pylab import *
3
4  data = loadtxt("SampleEMG.txt")
5
6  plot(data[:,5])
7  show()
```

BITalinoWorld/revolution-python-api: Python API for BITalino (r)evolution

GitHub, Inc. github.com/BITalinoWorld/revolution-python-api

Reader

Apple iCloud Facebook Twitter Wikipedia Yahoo News Popular

This repository Search Pull requests Issues Gist

BITalinoWorld / **revolution-python-api**

Unwatch 4 Star 1 Fork 2

Code Issues 0 Pull requests 0 Projects 0 Wiki Pulse Graphs Settings

Python API for BITalino (r)evolution

Edit

New Add topics

9 commits **1** branch **0** releases **3** contributors GPL-3.0

Branch: **master** New pull request Create new file Upload files Find file Clone or download

rfreixo8 committed on **GitHub** Update setup.py Latest commit ca24b9c on Dec 9, 2016

LICENSE.md	setup.py added	3 months ago
README.md	Update README.md	3 months ago
bitalino.py	read data for 5sec	3 months ago
setup.py	Update setup.py	3 months ago

README.md

BITalino (r)evolution Python API

The BITalino (r)evolution Python API provides the needed tools to interact with BITalino (r)evolution using Python.

Dependencies

- Python >2.7 or Anaconda

bitalino.py

bitalino.py 〉 No Selection

```
504
505 if __name__ == '__main__':
506     macAddress = "00:00:00:00:00:00"
507
508     batteryThreshold = 30
509     acqChannels = [0,3]
510     samplingRate = 1000
511     nSamples = 10
512     digitalOutput = [0,0,1,1]
513
514     # Connect to BITalino
515     device = BITalino(macAddress)
516
517     # Set battery threshold
518     print device.battery(batteryThreshold)
519
520     # Read BITalino version
521     device.version()
522
523     # Start Acquisition
524     device.start(samplingRate, acqChannels)
525
526     # Read samples
527     print device.read(nSamples)
528
529     # Turn BITalino led on
530     device.trigger(digitalOutput)
531
532     # Stop acquisition
533     device.stop()
534
535     # Close connection
536     device.close()
537
538
539
540
541
542
543
544
545
546
```

Spyder (Python 2.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

Preferences Ctrl+Alt+Shift+P
PYTHONPATH manager
Current user environment variables...
Reset Spyder to factory defaults
Update module names list
Open command prompt

C:\Users\Hugo Silva

Editor - C:\Users\Hugo Silva\.spyder\temp.py

temp.py

```
1 # -*- coding: utf-8 -*-
2 """
3 Spyder Editor
4
5 This is a temporary script file.
6 """
7
8
```

Help

Source Console Object

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

Variable explorer File explorer Help

IPython console

Console 1/A

```
Python 2.7.13 |Anaconda 4.3.0 (64-bit)| (default, Dec 19 2016, 13:29:36) [MSC v.1500 64 bit
(AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 5.1.0 -- An enhanced Interactive Python.
?         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.

In [1]:
```

Python console History log IPython console

Python Path Manager

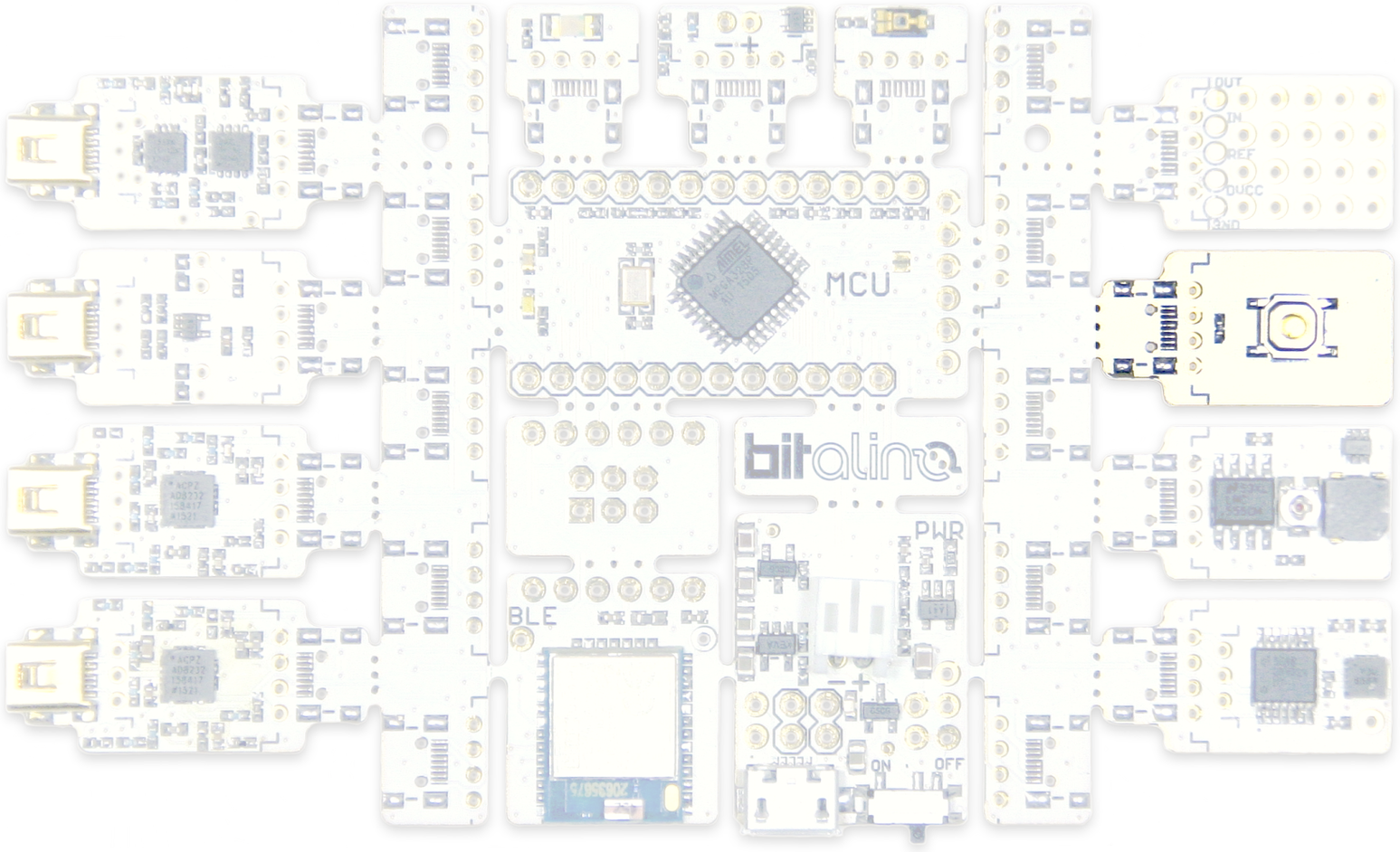
Permissions: **RW** End-of-lines: **CRLF** Encoding: **UTF-8** Line: **1** Column: **1** Memory: **61 %**

POR 23:14 04/03/2017

LightsBIT.py

LightsBIT.py 〉 No Selection

```python
# -*- coding: utf-8 -*-
import bitalino

import numpy

macAddress = "/dev/tty.BITalino-60-88-DevB"

device = bitalino.BITalino(macAddress)

state = device.state()

toggle = 1-state['digitalChannels'][2]

device.trigger([toggle, 0])

device.battery(0 if toggle else 63)

print "LIGHTS ON" if toggle else "LIGHTS OFF"

device.close()
```



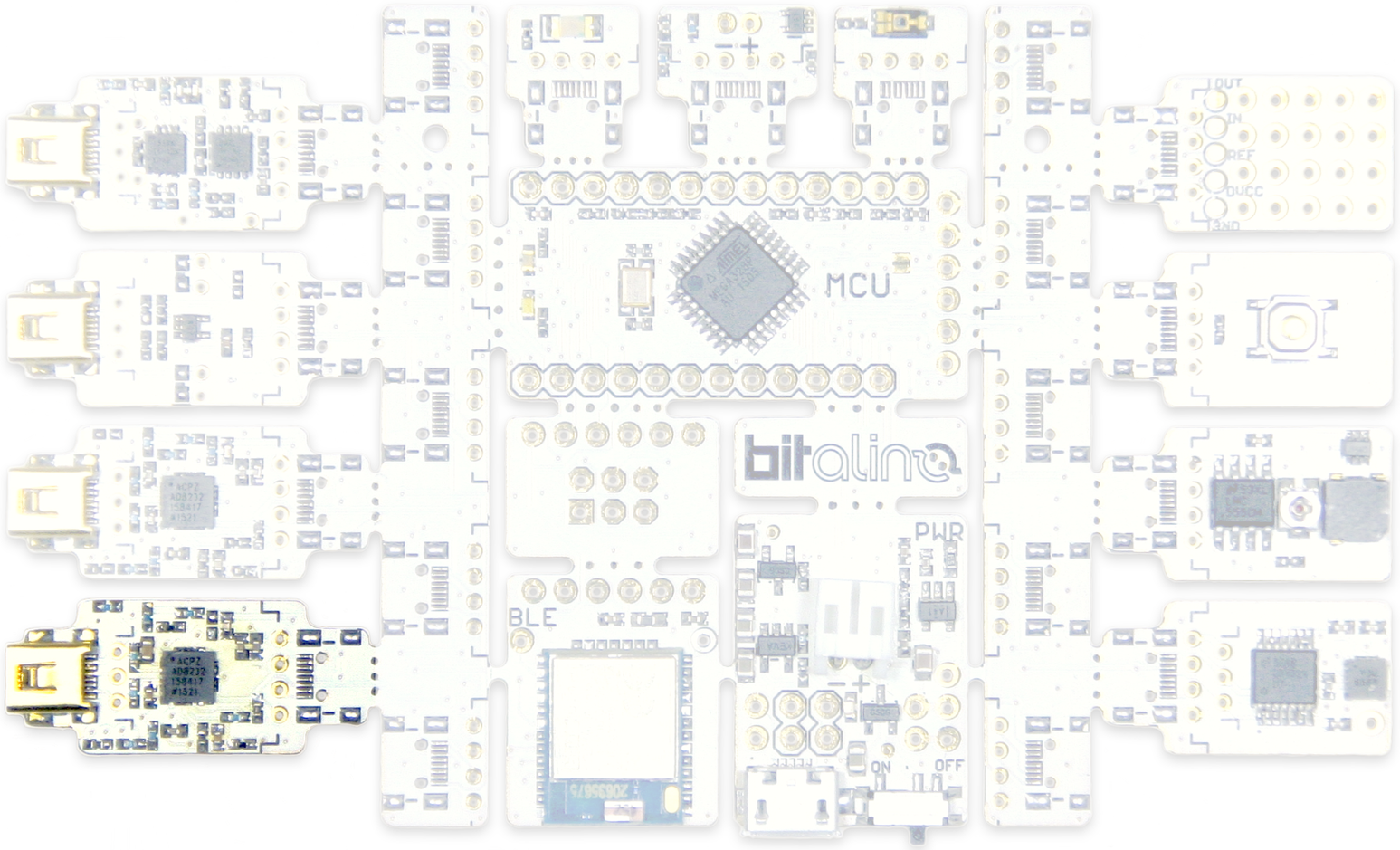
Pushbutton (BTN)

manual event annotation

ButtonBIT.py

ButtonBIT.py 〉 No Selection

```python
# -*- coding: utf-8 -*-
import bitalino

import numpy

macAddress = "/dev/tty.BITalino-60-88-DevB"

device = bitalino.BITalino(macAddress)

while True:
    state = device.state()

    toggle = state['digitalChannels'][0]

    device.trigger([toggle, 0])

    device.battery(0 if toggle else 63)

    print "LIGHTS ON" if toggle else "LIGHTS OFF"

device.close()
```



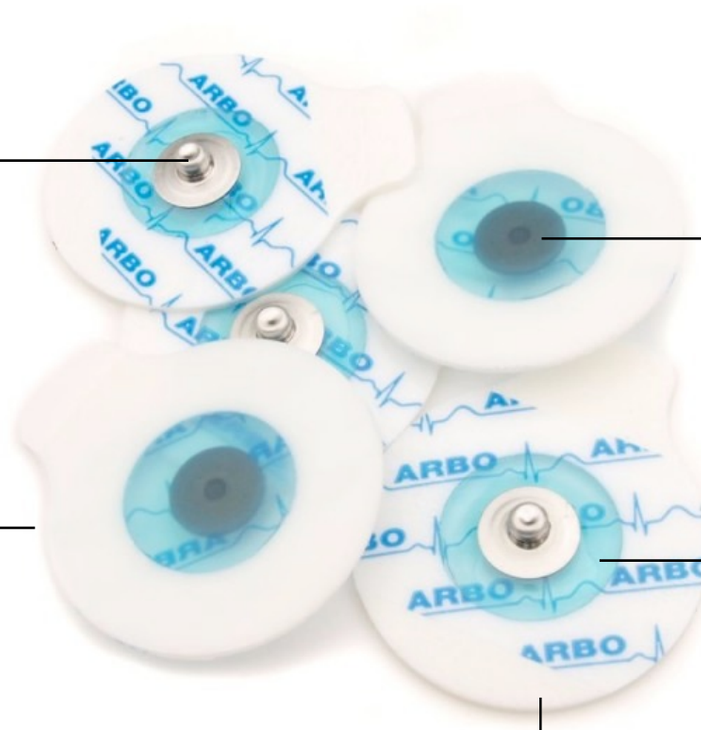
Electromyography (EMG)

muscle activity

“anatomy” of an electrode

metal stud

snap-on sensor connection



detection surface

iron, platinum or Ag/AgCl*

*the most frequently used

protective substrate (optional)

thin transparent protection film

(optional) conductive gel

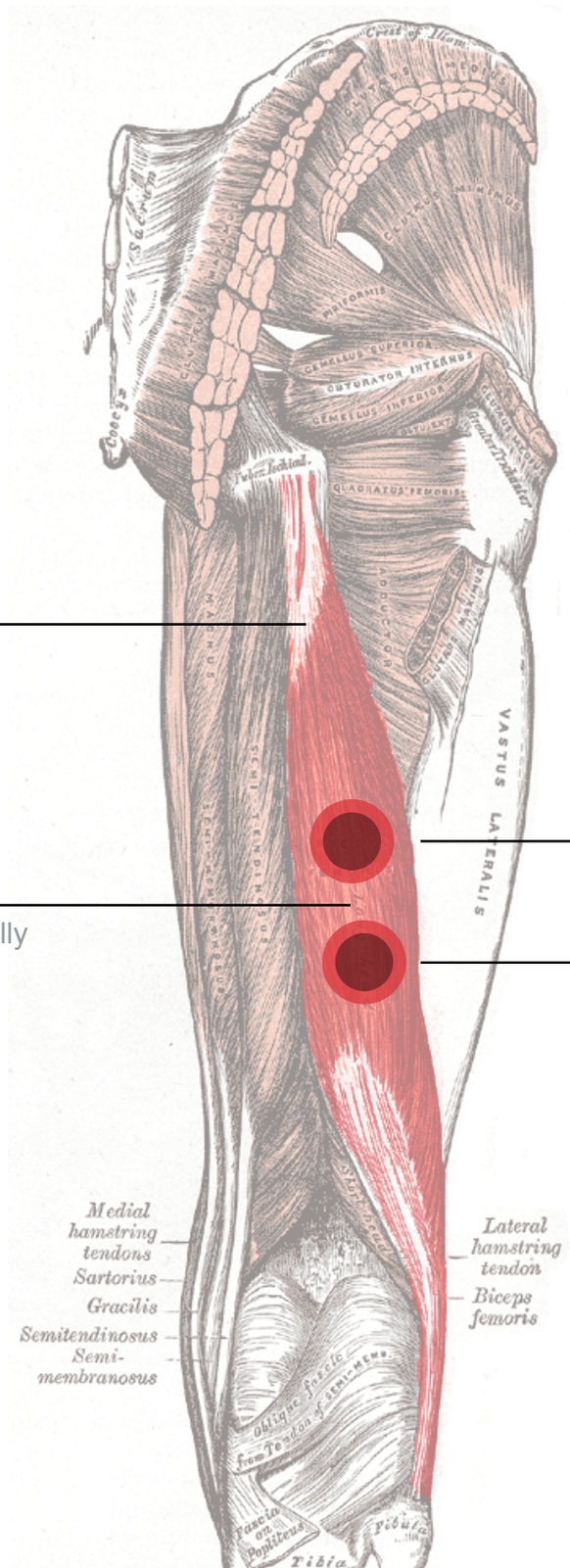
highly conductive electrolyte

(often with with self-adhesive properties)

support substrate (optional)

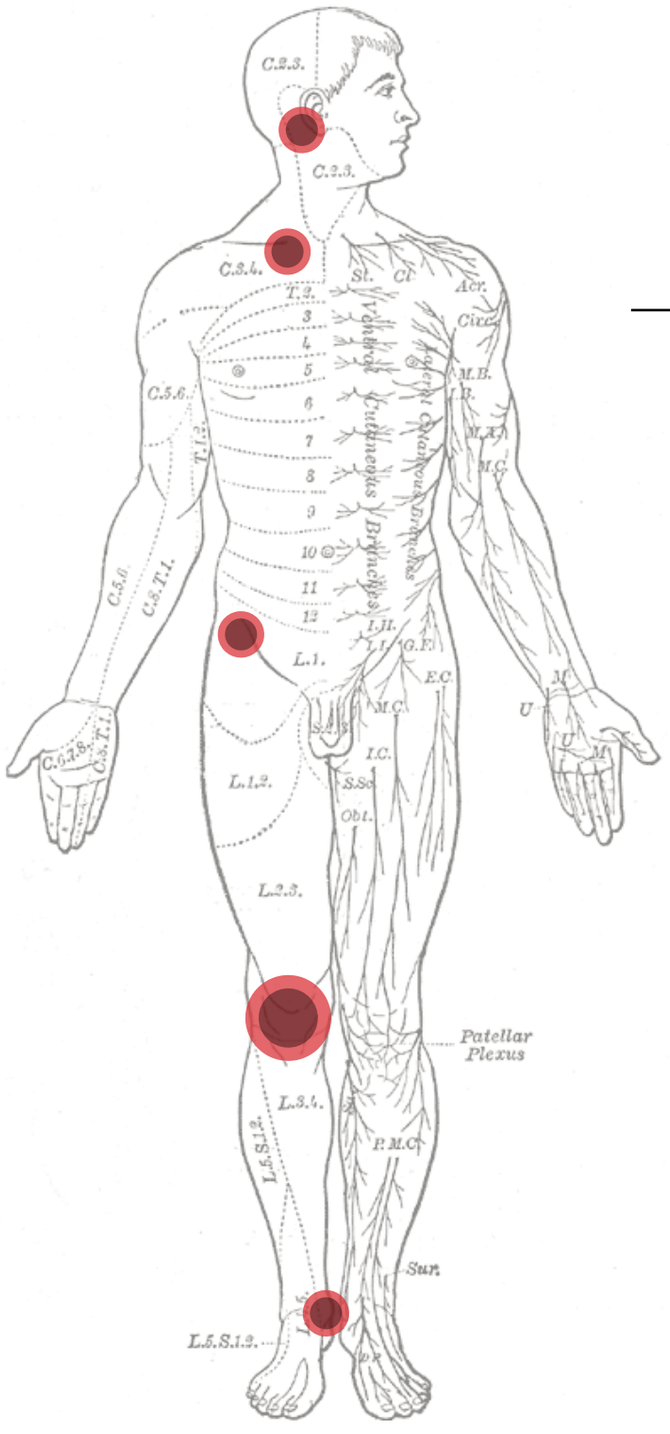
layer to hold electrode in place

electrode **placement**



http://www.bartleby.com/107/illus434.html

reference electrode



ground

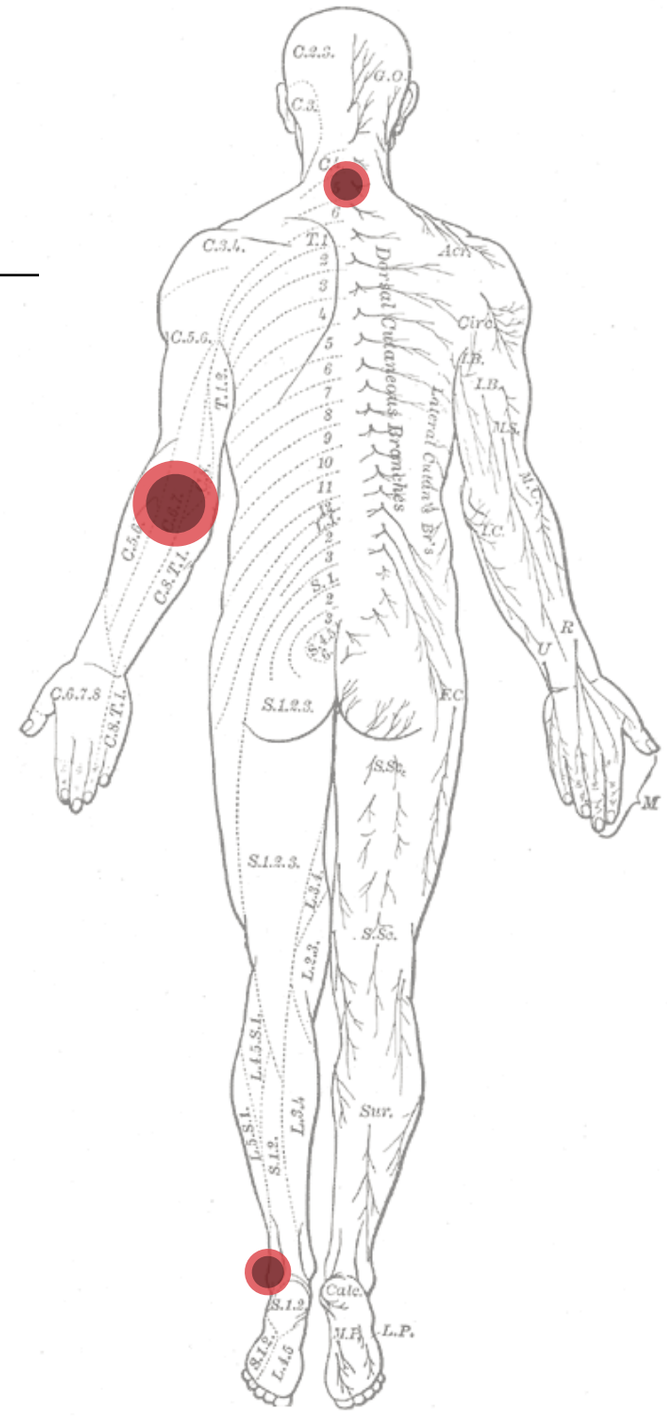
---

needs to be highly conductive

placed over neutral tissue

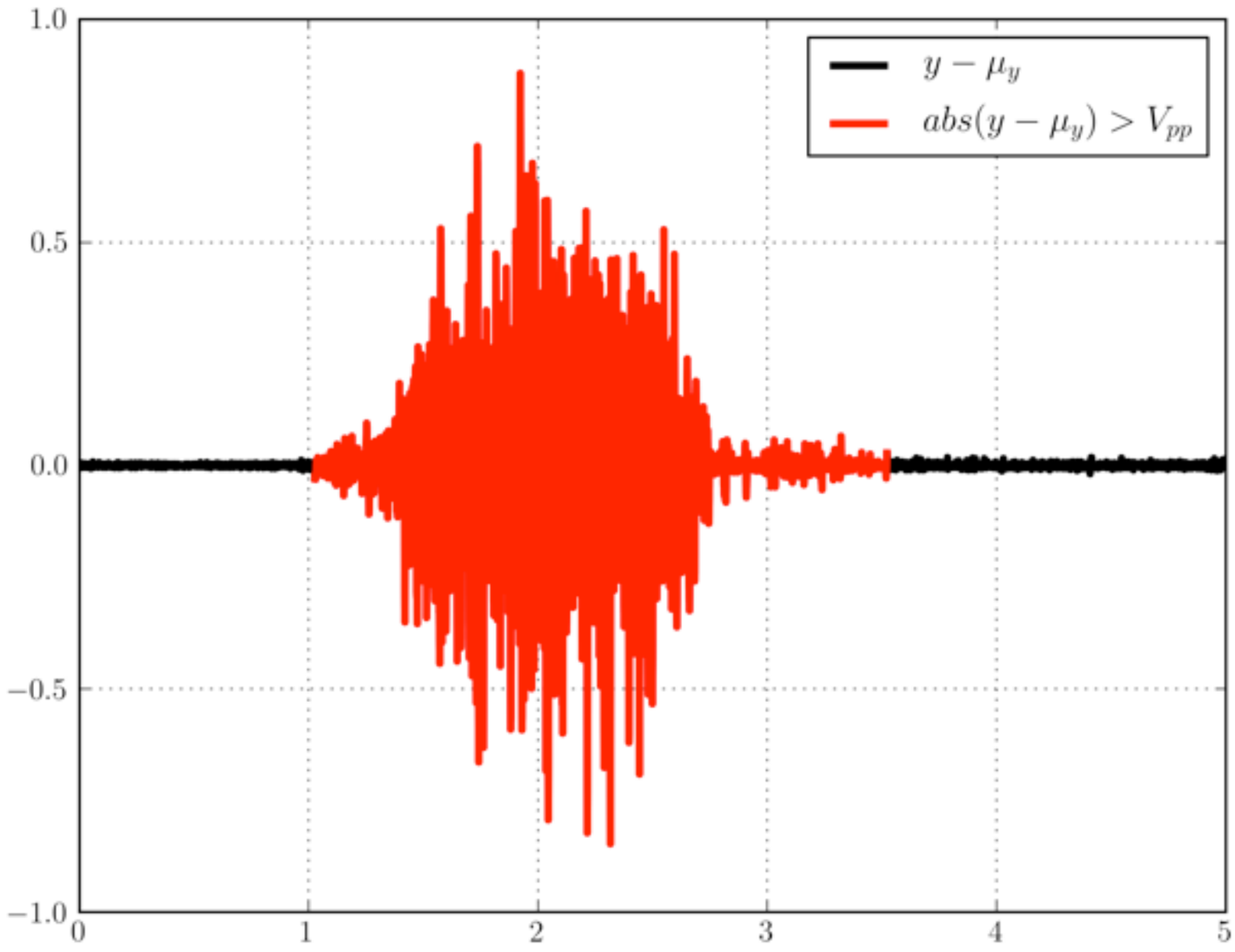
bone structures (preferred)

large area (indicative)

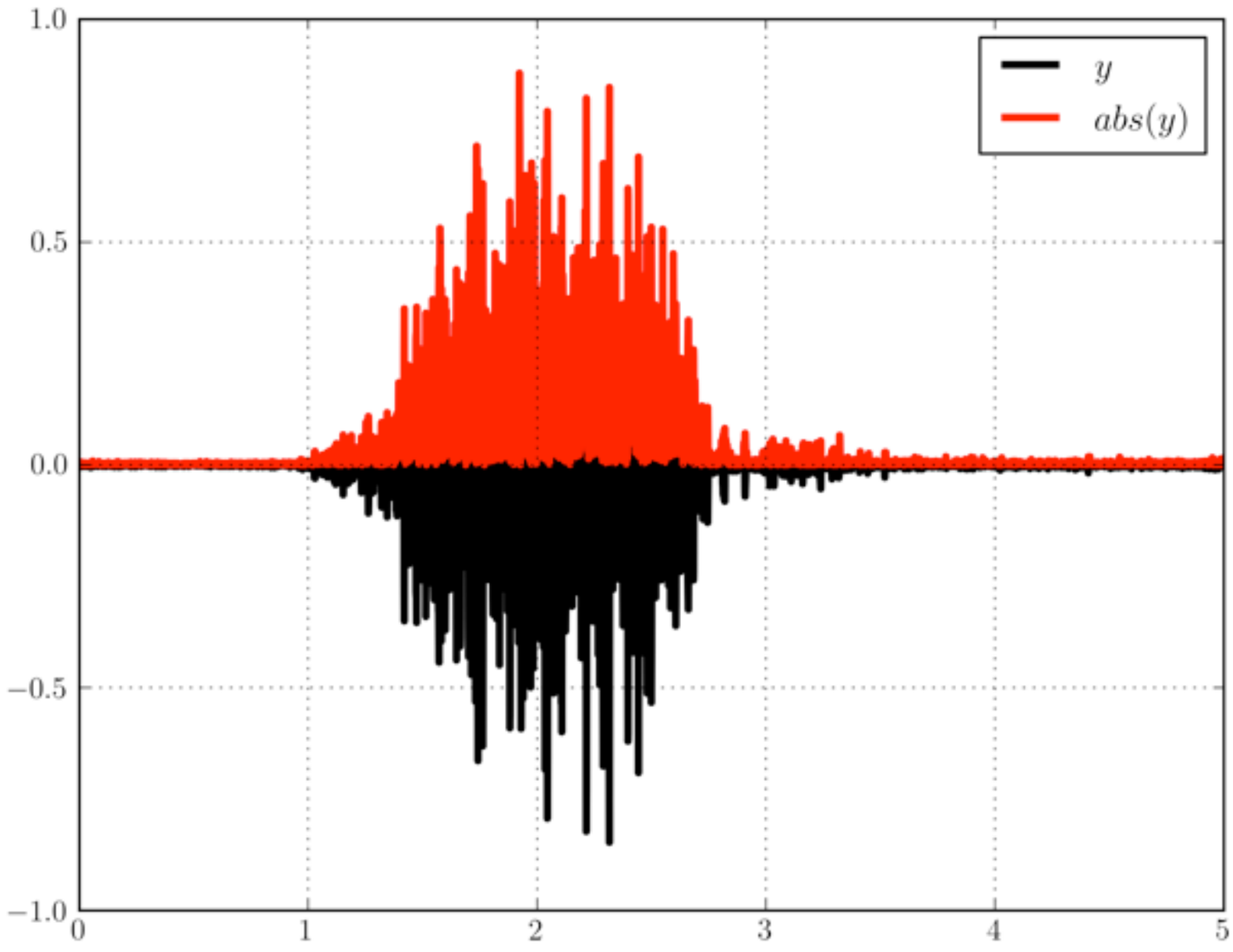


http://www.bartleby.com/107/illus797.html
http://www.bartleby.com/107/illus798.html

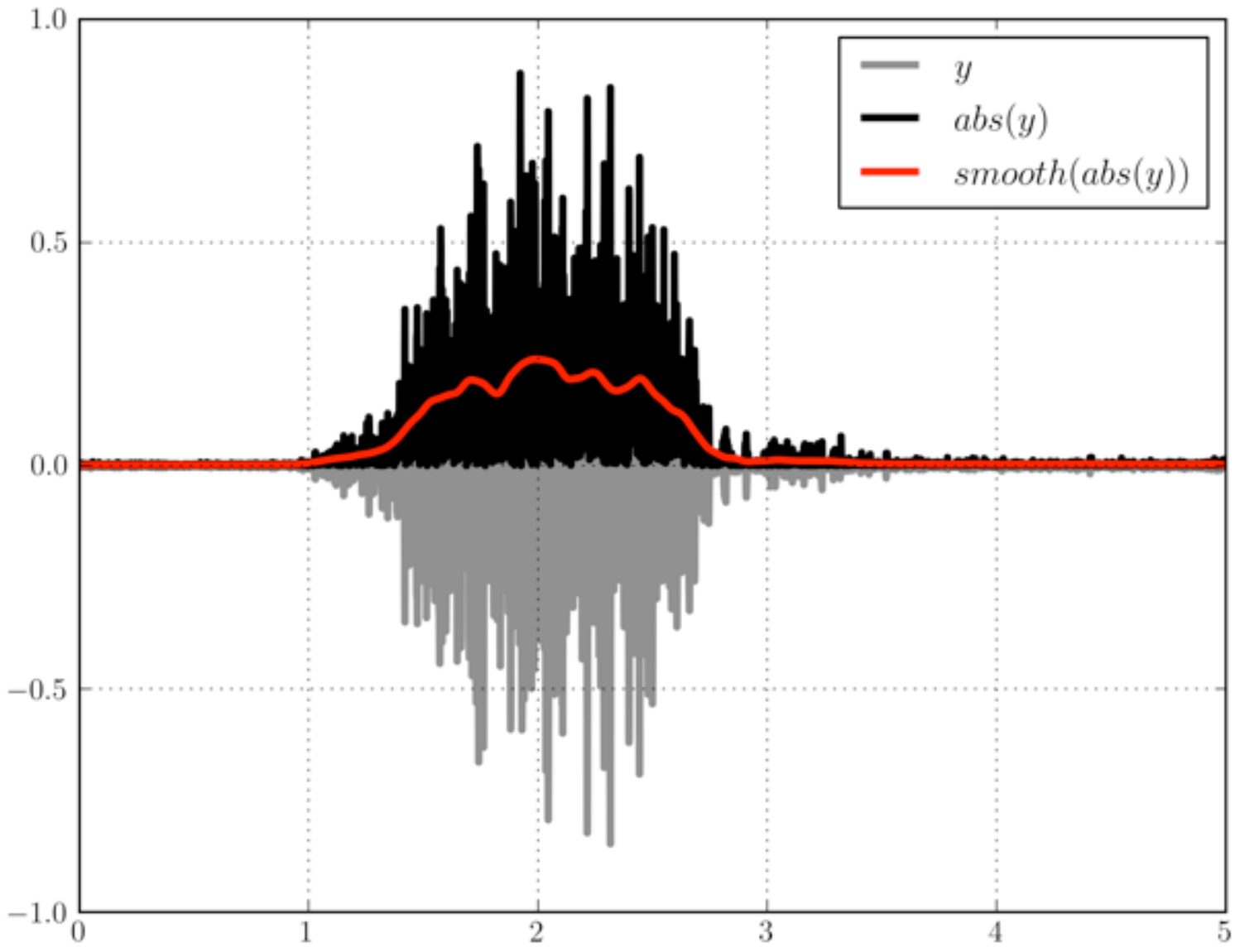
basic **onset** detection



envelope computation



envelope computation



MuscleBIT.py

MuscleBIT.py 〉 No Selection

```python
# -*- coding: utf-8 -*-
import bitalino

import numpy
import time

macAddress = "/dev/tty.BITalino-60-88-DevB"

device = bitalino.BITalino(macAddress)
time.sleep(1)

srate = 1000
nframes = 100
threshold = 5

device.start(srate, [0])
print "START"

try:
    while True:

        data = device.read(nframes)

        if numpy.mean(data[:, 1]) < 1: break

        EMG = data[:, -1]

        envelope = numpy.mean(abs(numpy.diff(EMG)))

        if envelope > threshold:
            device.trigger([0, 1])
        else:
            device.trigger([0, 0])

finally:
    print "STOP"
    device.stop()
    device.close()
```