Olimexino-2560, SHIELD-EKG-EMG, and Arduino IDE Serial Plotter.

This project demonstrates how to get create a basic EKG measurement system using Olimex boards and Arduino IDE. One of the goals is to use Arduino's serial plotter tool.

1. Hardware setup

This project requires the following hardware parts (which can be seen in the picture 1 at the bottom of the page):

- **Olimexino-2560** x 1 – main Arduino-like board – LINK

- **SHIELD-EKG-EMG** x 1 – EKG shield attached to the Arduino-like board – LINK

- **SHIELD-EKG-EMG-PRO** x 1 – compatible with the shield EKG cable – LINK

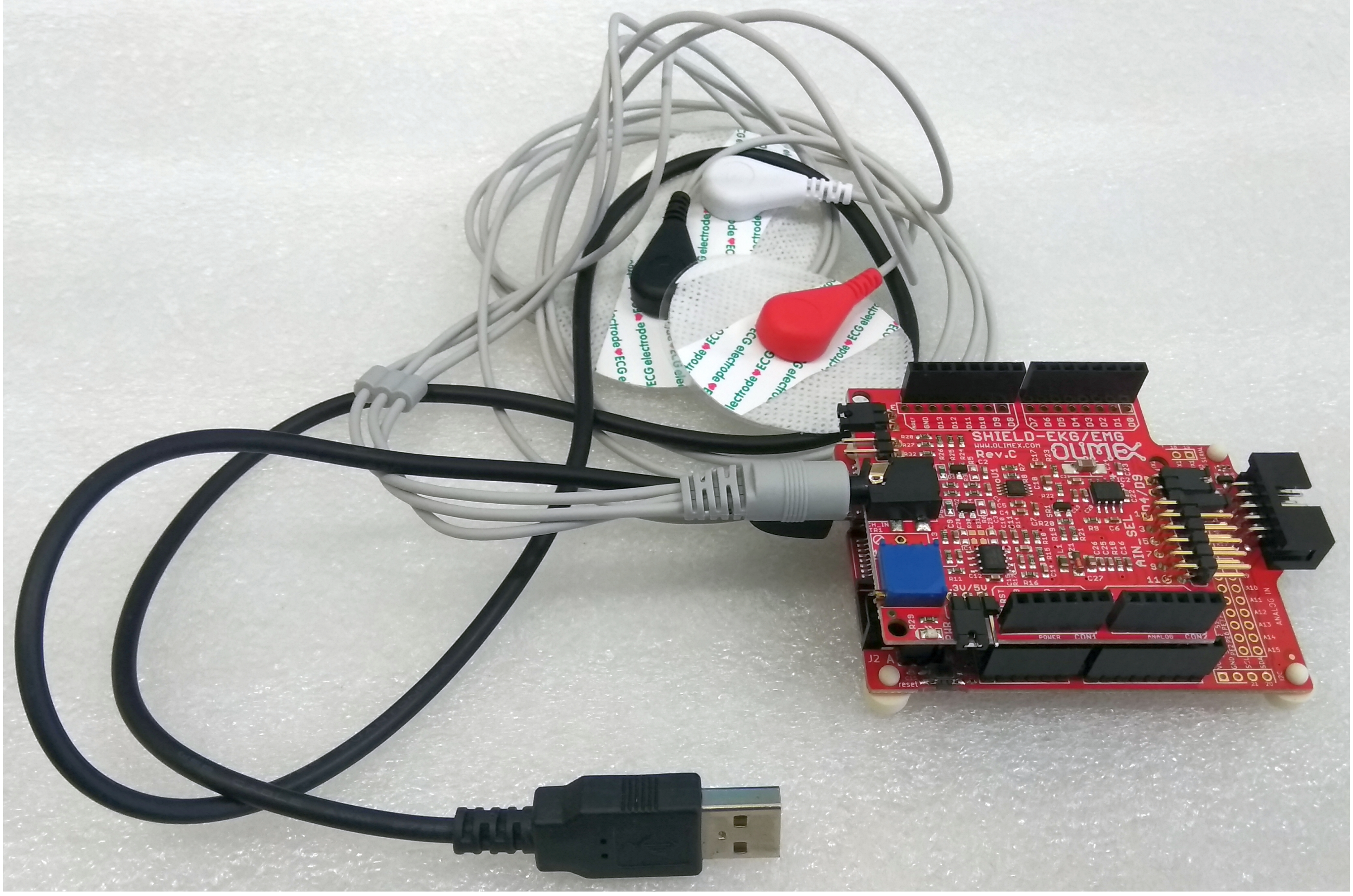
- **ECG-GEL-ELECTRODE** x 3 – electrodes for the EKG cable – LINK

- **USB MICRO CABLE** x 1 – for connecting the Arduino-like board to the PC – LINK

And of course, a computer running Arduino IDE and a free USB port.

Picture 1 Required hardware

Connect everything as shown below in picture 2, below:



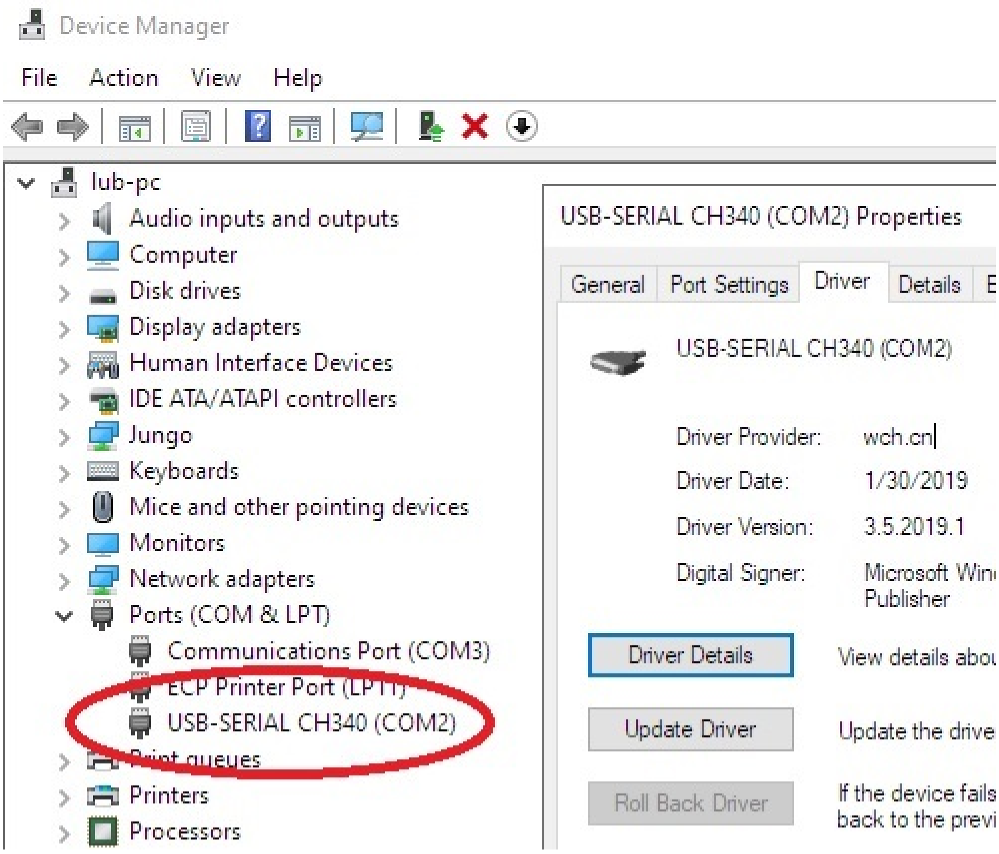
Picture 2 Hardware prepared to connect to PC and body

Make sure jumper REF_E is closed and leave the other jumpers as per default (3.3V/5V in position 3.3V, AIN_SEL set to 1, D4/D9 set to D9, and for Olimexino-2560 – 3.3V/5V set again to 3.3V, A/M set to A).

Then connect the board to the computer using the USB cable. Once the hardware setup is complete, and before we attach the electrodes to the body, we should complete the software setup.

2. Software setup

The board would be recognized as CH340 USB-SERIAL device, and give a COM number. Under Windows 10 it should look like this:



USB-SERIAL CH340 (COM2)

Driver Provider: wch.cn

Driver Date: 1/30/2019

Driver Version: 3.5.2019.1

Digital Signer: Microsoft Win
Publisher

View details abou

Update the drive

If the device fails
back to the previ

Picture 3 Olimexino-2560's USB-serial converter CH340

If it doesn't get recognized and installed automatically, make sure to install the drivers for CH340T.

For Windows you can find drivers here:

https://www.olimex.com/Products/Breadboarding/BB-CH340T/resources/CH341SER.zip

For Linux:

https://github.com/OLIMEX/ch340-dkms

For MAC:

https://www.olimex.com/Products/Breadboarding/BB-CH340T/resources/CH341SER_MAC.ZIP

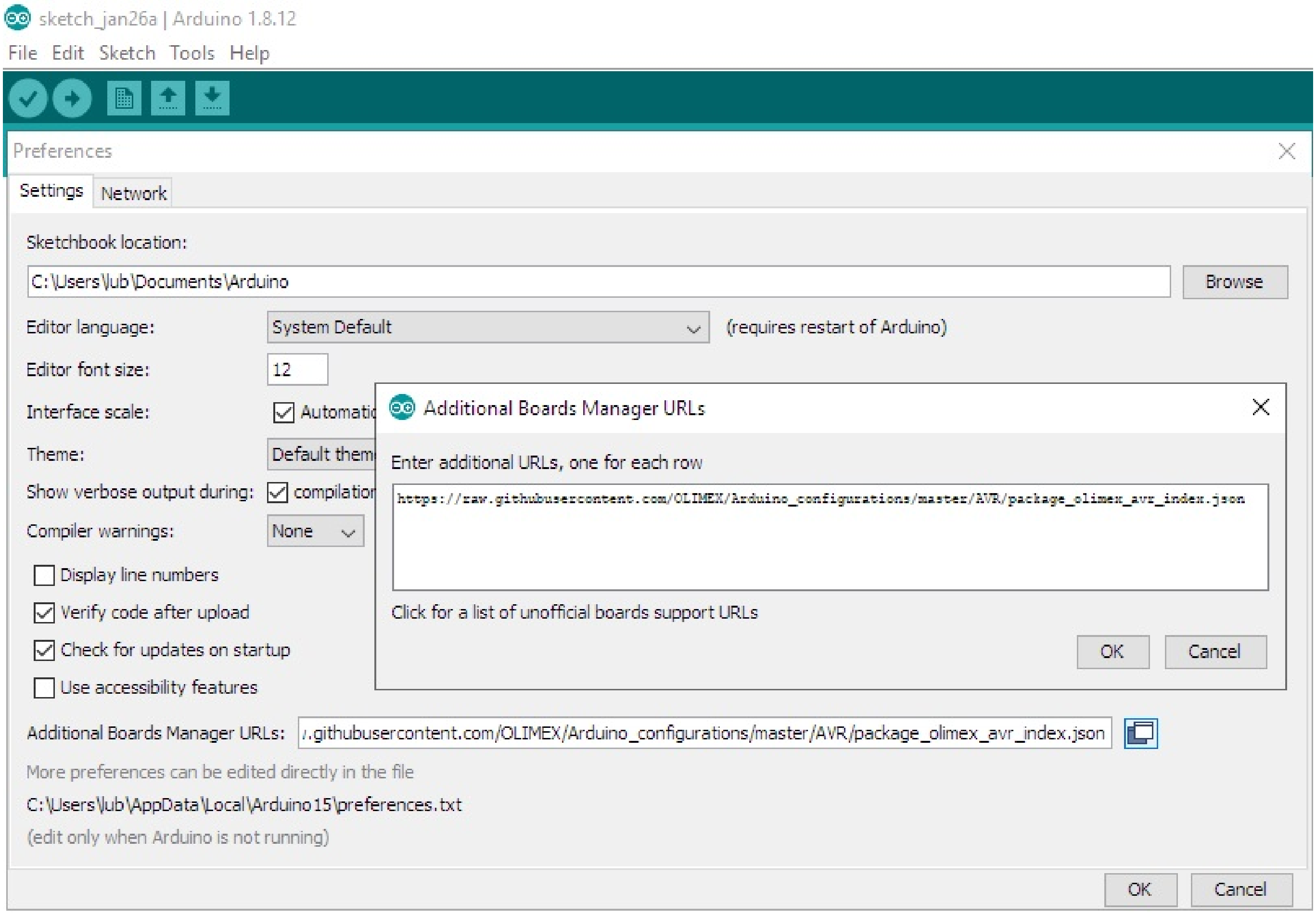
Once the Olimexino-2560 is properly recognized, install latest Arduino IDE if you don't have it already, you can get it from the official web-site

https://www.arduino.cc/

Once installed start Arduino IDE and add the Olimex AVR support package to it (to get OLIMEXINO-2560 in the board selection menu). Navigate to "File" → "Preferences". A new dialog would appear, search for "Additional Boards Manager URLs" field – you need to put a link or links to the Olimex AVR json file. Copy and paste the full https link in the "Additional Boards Manager URLs" field. You can have multiple json links, separated by semi-colon.

- json link for Olimex AVR boards (like OLIMEXINO-2560, OLIMEXINO-328, OLIMEXINO-32U4, OLIMEXINO-NANO,OLIMEXINO-85, among others):

https://raw.githubusercontent.com/OLIMEX/Arduino_configurations/master/AVR/package_olimex_avr_index.json


sketch_jan26a | Arduino 1.8.12
File Edit Sketch Tools Help
Preferences
Settings
Network
Browse
Additional Boards Manager URLs
OK
Cancel
OK
Cancel

Sketchbook location: C:\Users\lub\Documents\Arduino

Editor language: System Default (requires restart of Arduino)

Editor font size: 12

Interface scale: ☑ Automati

Theme: Default them

Show verbose output during: ☑ compilatior

Compiler warnings: None

☐ Display line numbers

☑ Verify code after upload

☑ Check for updates on startup

☐ Use accessibility features

Additional Boards Manager URLs: /.githubusercontent.com/OLIMEX/Arduino_configurations/master/AVR/package_olimex_avr_index.json

More preferences can be edited directly in the file

C:\Users\lub\AppData\Local\Arduino15\preferences.txt

(edit only when Arduino is not running)

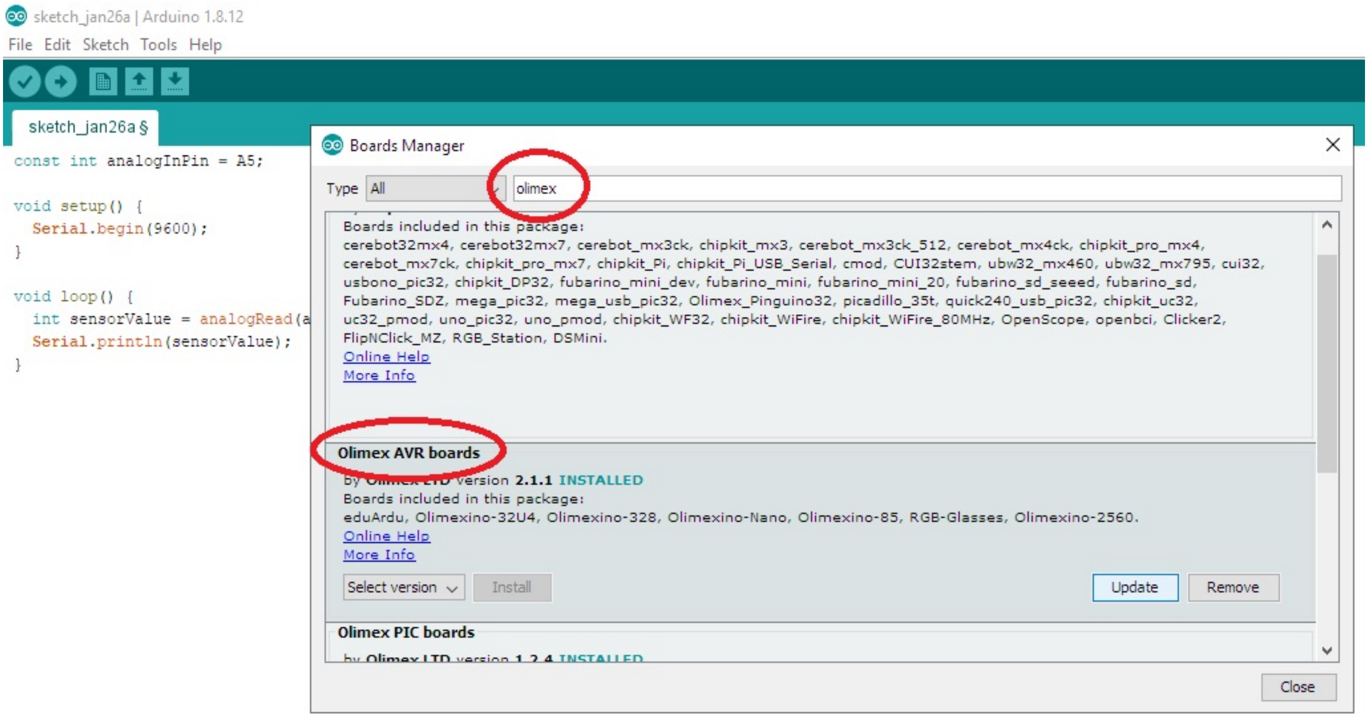
Enter additional URLs, one for each row

```
https://raw.githubusercontent.com/OLIMEX/Arduino_configurations/master/AVR/package_olimex_avr_index.json
```

Click for a list of unofficial boards support URLs

Picture 4: Adding Olimex AVR json to Arduino IDE

After the json link had been added, install the Olimex AVR package. Navigate to “Tool” → “Board” → “Boards Manager...”. A new window would appear, find the Olimex AVR package and click "Install" in the bottom right corner, or “Update” if you already have it. Note that the download and the installation might take a long time for certain boards since they might also come with different compiler.



```
const int analogInPin = A5;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int sensorValue = analogRead(a
  Serial.println(sensorValue);
}
```

olimex

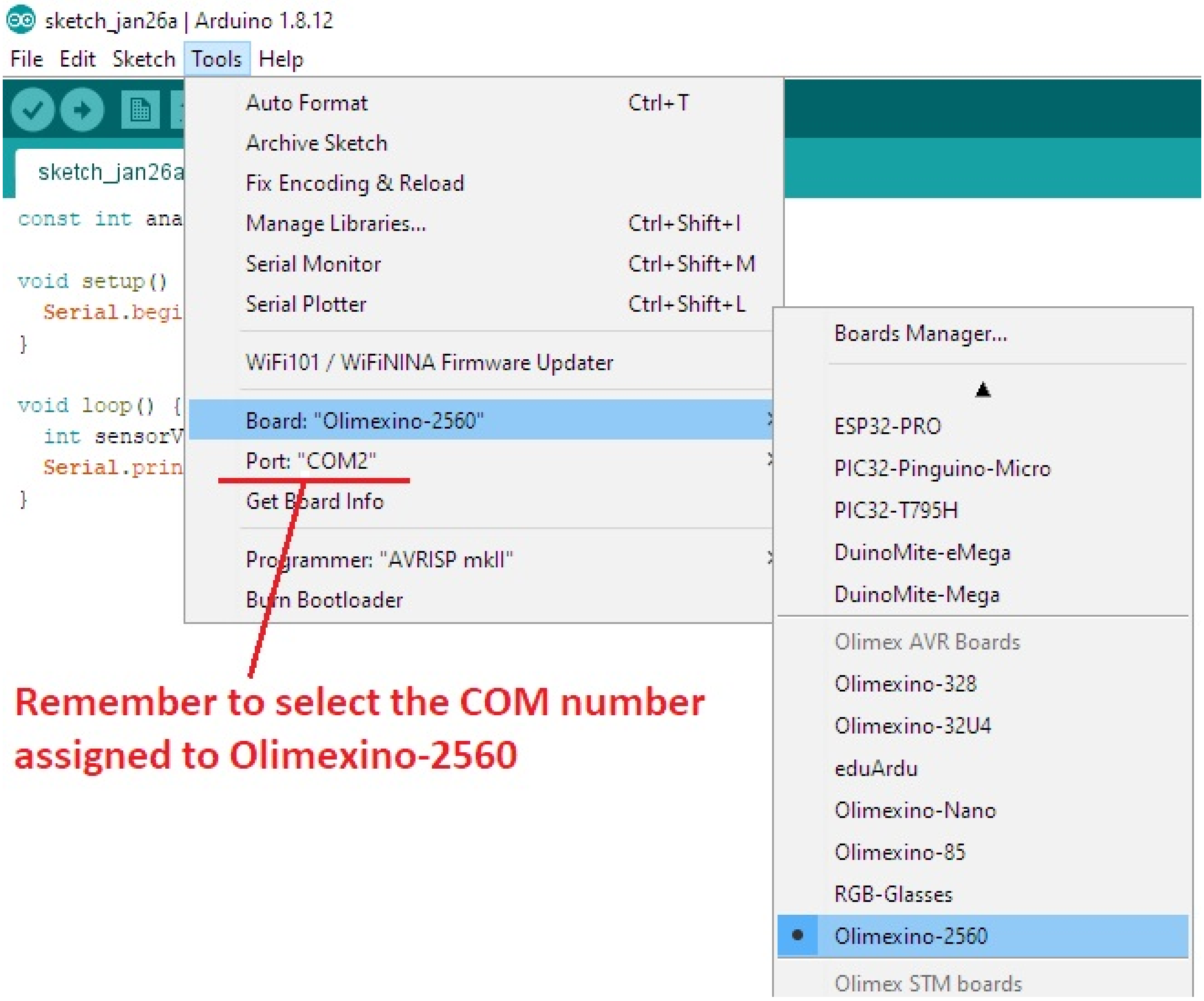
Boards included in this package:
cerebot32mx4, cerebot32mx7, cerebot_mx3ck, chipkit_mx3, cerebot_mx3ck_512, cerebot_mx4ck, chipkit_pro_mx4, cerebot_mx7ck, chipkit_pro_mx7, chipkit_Pi, chipkit_Pi_USB_Serial, cmod, CUI32stem, ubw32_mx460, ubw32_mx795, cui32, usbono_pic32, chipkit_DP32, fubarino_mini_dev, fubarino_mini, fubarino_mini_20, fubarino_sd_seeed, fubarino_sd, Fubarino_SDZ, mega_pic32, mega_usb_pic32, Olimex_Pinguino32, picadillo_35t, quick240_usb_pic32, chipkit_uc32, uc32_pmod, uno_pic32, uno_pmod, chipkit_WF32, chipkit_WiFire, chipkit_WiFire_80MHz, OpenScope, openbci, Clicker2, FlipNClick_MZ, RGB_Station, DSMini.

Olimex AVR boards
by Olimex LTD version **2.1.1** INSTALLED
Boards included in this package:
eduArdu, Olimexino-32U4, Olimexino-328, Olimexino-Nano, Olimexino-85, RGB-Glasses, Olimexino-2560.

Olimex PIC boards
by Olimex LTD version 1.2.4 INSTALLED

Picture 5: Finding Olimex AVR package in the “Boards Manager”

Once these installation are done, we have to select the Olimexino-2560 board from the board selection and the proper COM port that was assigned by the operating system. In my case the COM port was COM2.



```
const int ana

void setup()
  Serial.begi
}

void loop() {
  int sensorV
  Serial.prin
}
```

Board: "Olimexino-2560"

Port: "COM2"

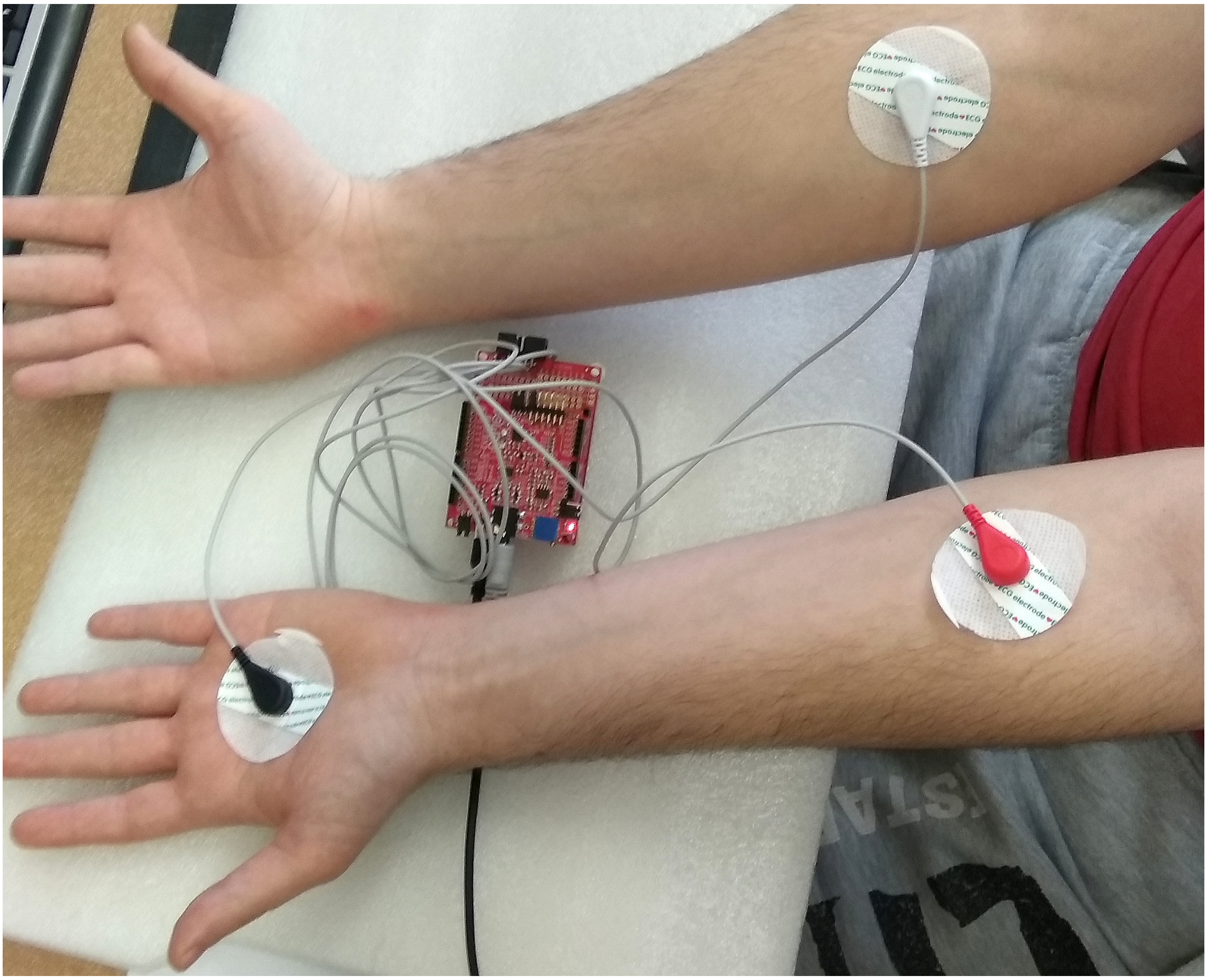
Remember to select the COM number assigned to Olimexino-2560

Picture 6: Selecting the Olimexino-2560 board and the assigned COM port

3. Attaching the electrodes to the body

The cable has three ends with different colors – red, white, and black one. The red one is marked with “L” and goes to the left arm, the white one with “R” and goes to the right arn. The black one is neutral or electrical earth and has to go to the right leg for best results, but you can also attach it to any place that is away from the right arm and the left arm.

For the purpose of testing and to save trouble I just attached the electrodes to my hands as shown in the picture below, notice that for best results the black electrode is recommended to be attached to a leg or other place away from the arms (I am not aiming for best results here):



Picture 7: Quick attachment of electrodes, suitable for R&D; for best results the black electrode should be connected to the right leg

4. Demo code and results

The demo code is the most basic one without any attempt to modify the data. The important part is to select the proper analog input. This is according to the AIN_SEL jumper (which by default goes to A0). Copy-paste the code below to a new Arduino project.

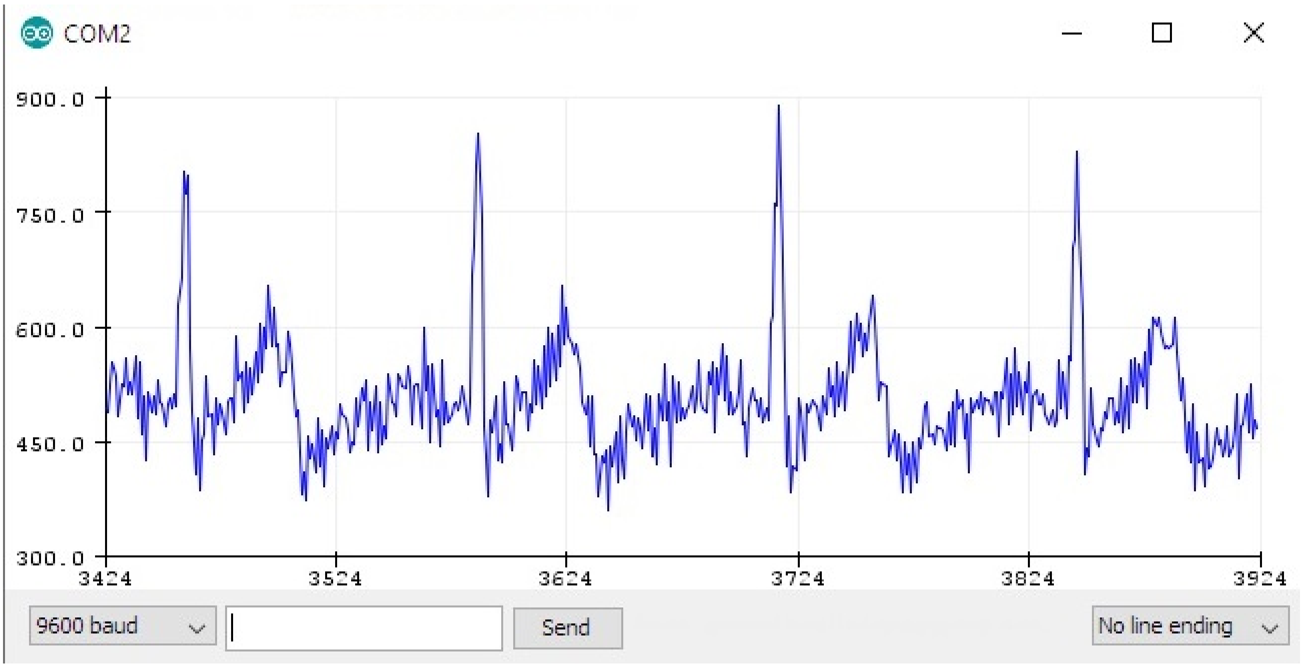
```
const int analogInPin = A0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int sensorValue = analogRead(analogInPin);
  Serial.println(sensorValue);
}
```

So in case you have changed AIN_SEL jumper to position 3, simply replace A0 with the corresponding analog input (A1). This jumper's position is used if you have more than one SHIELD-EKG-EMG connected together, so that each shield uses different analog pin.

Verify and upload the code to the Olimexino-2560. Now start the serial plotter from "Tools" → "Serial Plotter". If you've done everything as described, you should see something similar:



Picture 8: EKG data as seen in "Serial Plotter"

5. Further ideas

This project can serve as basis for bigger and more advanced projects, not just for EKG but also for EMG research. You can use it as basis for connecting more than one shield, SHIELD-EKG-EMG allows to have up to 6 shields connected, each using one of the analog pins selected by AIN_SEL jumper. Instead of using the Arduino IDE serial plotter you can always use Electric Guru other visualization software. Or you can write your own visualizator.