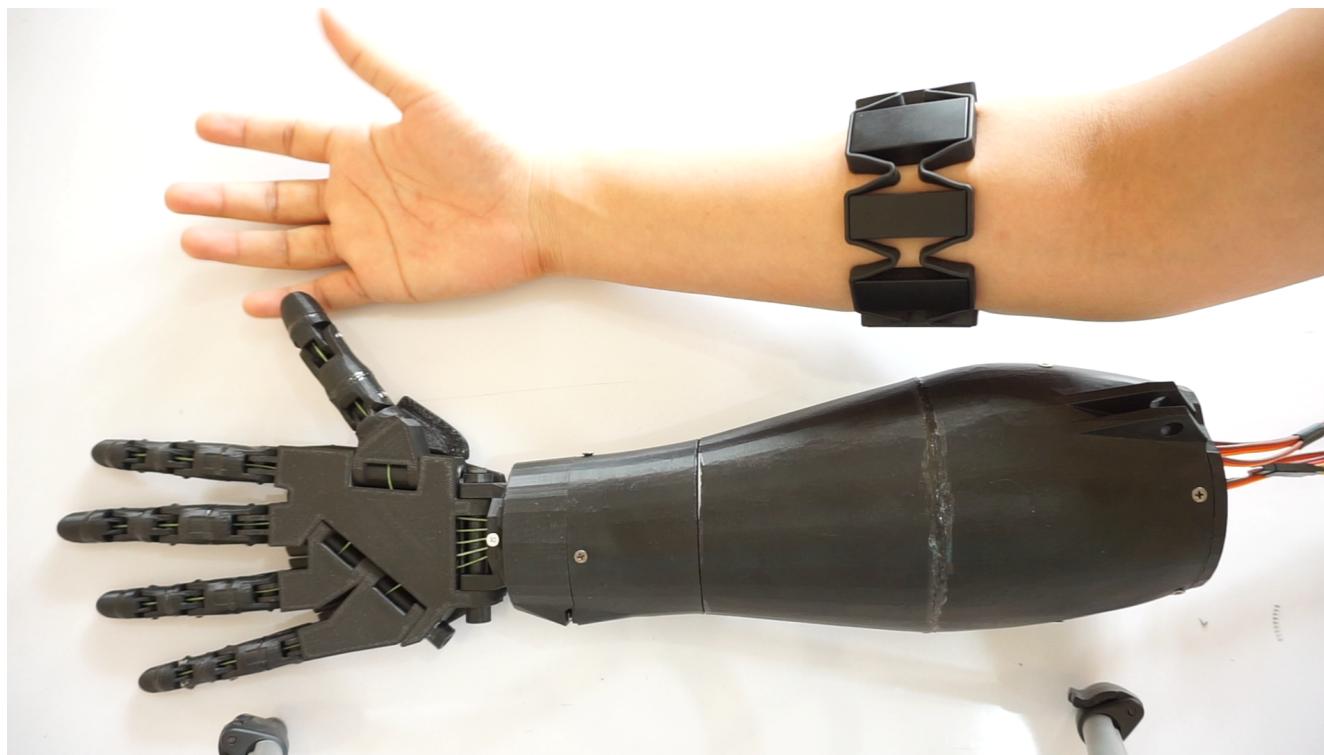


Machine, I and feedback loops

Wifi:

- **TECHNARIUM**, password: **user23422**
- <https://github.com/Technariumas/MyoElectrics>

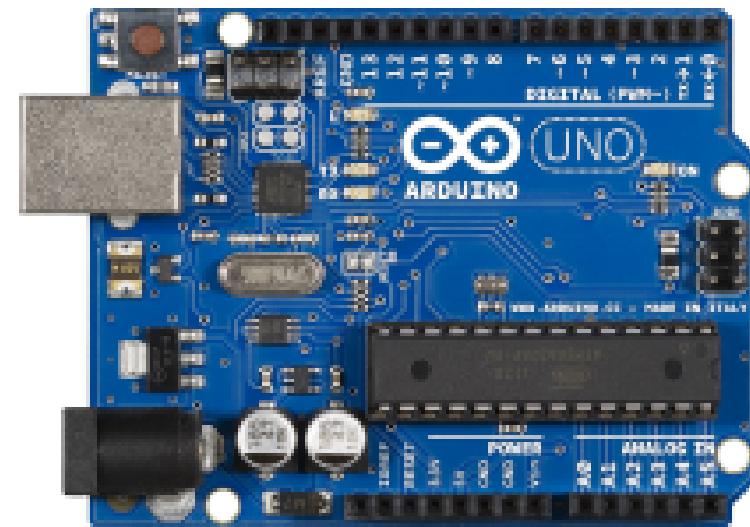
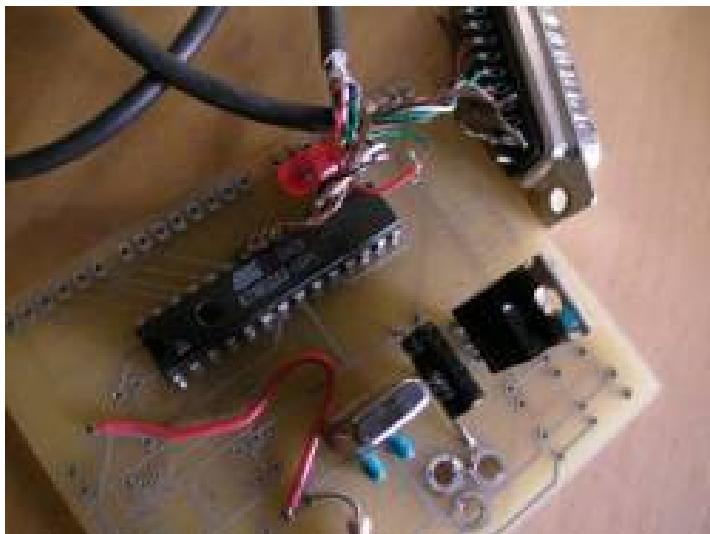


Apimtis

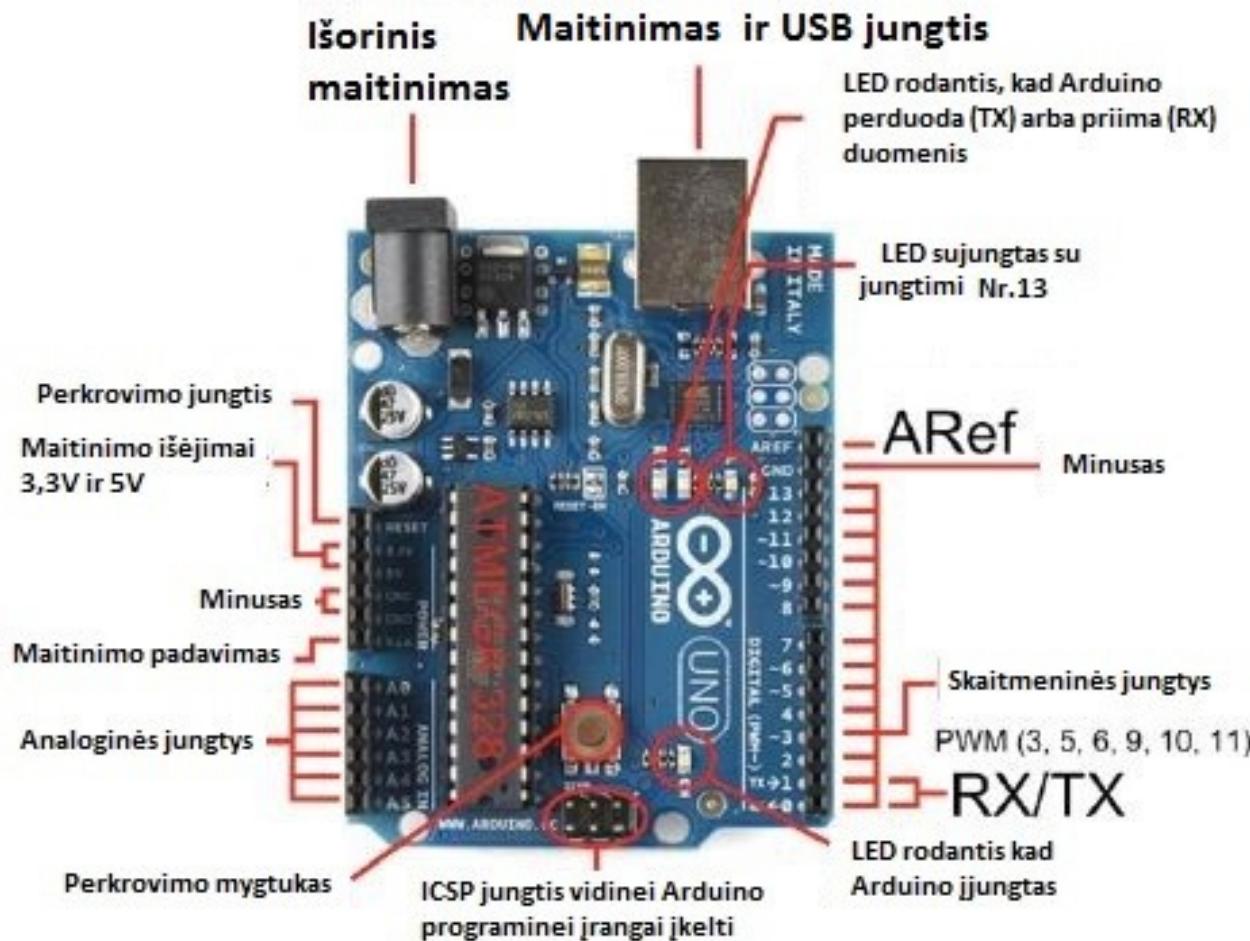
- Susipažinsime su Arduino mikrokontroleriu, jo programavimo aplinka
- Susipažinsime su Mstim plokštėmis, skirtomis raumenų elektrinei stimuliacijai
- Surinksime bandomają elektrinę schemą
- Užprogramuosime Arduino, mokysimės
- Sujungsime kitus sensorius: EMG, EKG
- Kursime grįžtamojo ryšio tarp mūsų kūnų ir mikrokontrolerio sistemas

Arduino

- Atviro kodo platforma
- Atmel mikrokontroleris
- Prieinamas ir lengvai programuojamas
- Plečiamas – turi “shield’ų” visiems gyvenimo atvejams

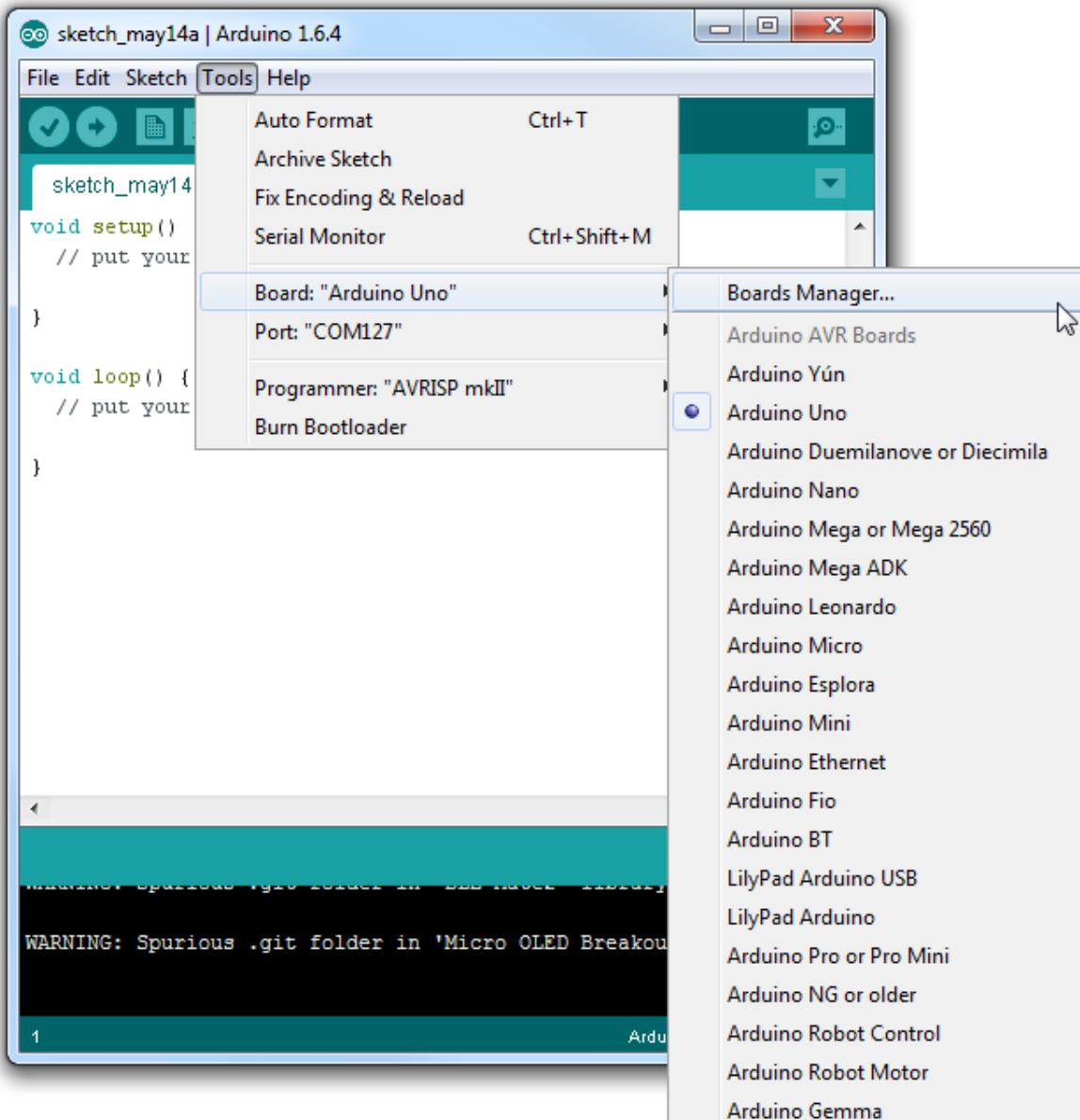


Arduino anatomija



(Robotikos mokyklos iliustracija)

Arduino IDE konfigūravimas



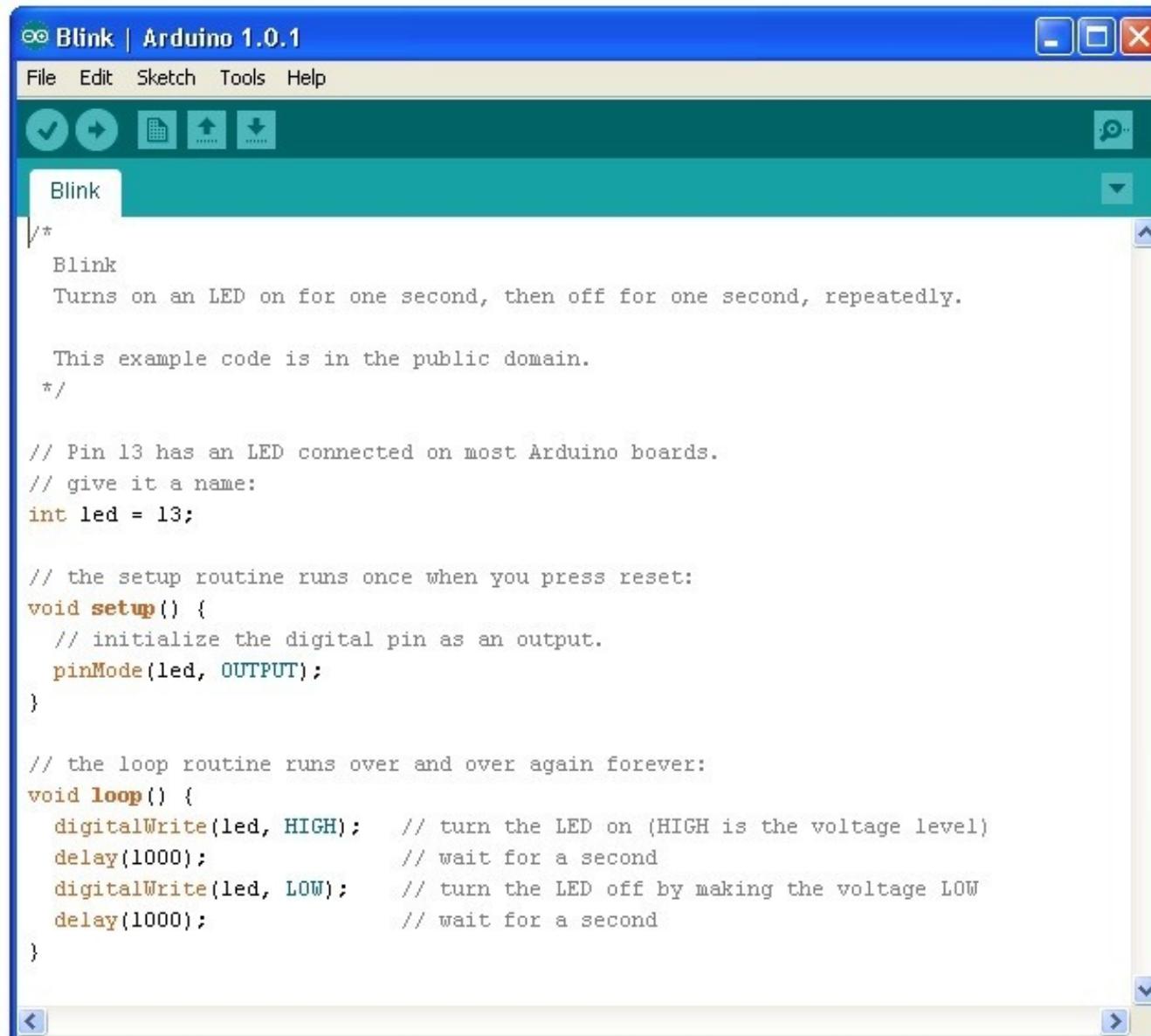
Arduino programa

- Funkcijos
- Kintamieji
- Komentarai

```
void setup() {  
    Serial.begin(9600);  
    Serial.println("Hello, world!");  
}  
  
void loop() {  
}
```

Arduino programu struktūra

- File > Examples > Basic > Blink



The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 1.0.1". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for new, open, save, and upload. The main area displays the "Blink" sketch code. The code is a standard Arduino sketch for controlling an LED on pin 13. It includes setup and loop functions with digitalWrite and delay commands. A red arrow points to the "Upload" button in the toolbar.

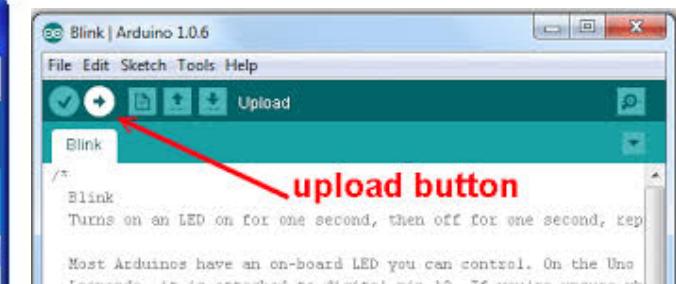
```
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.

This example code is in the public domain.
*/

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);      // turn the LED on (HIGH is the voltage level)
  delay(1000);                // wait for a second
  digitalWrite(led, LOW);       // turn the LED off by making the voltage LOW
  delay(1000);                // wait for a second
}
```



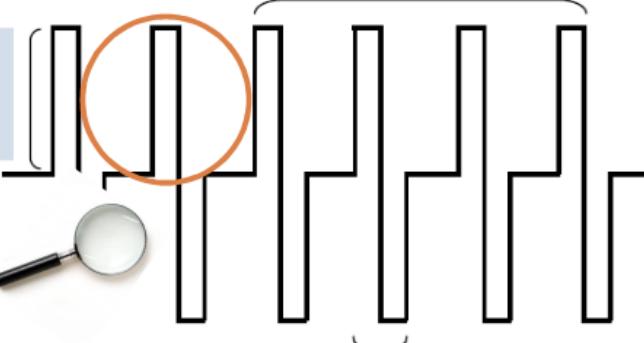
Raumenų elektrostimuliacija



Amplitude (mA)
intensity of electrical stimulation,
amount of energy flowing/ unit time



Frequency (Hz)
the rate pulses are delivered,
no of pulses/ sec



Pulse width (μsec)
duration of the electrical pulse

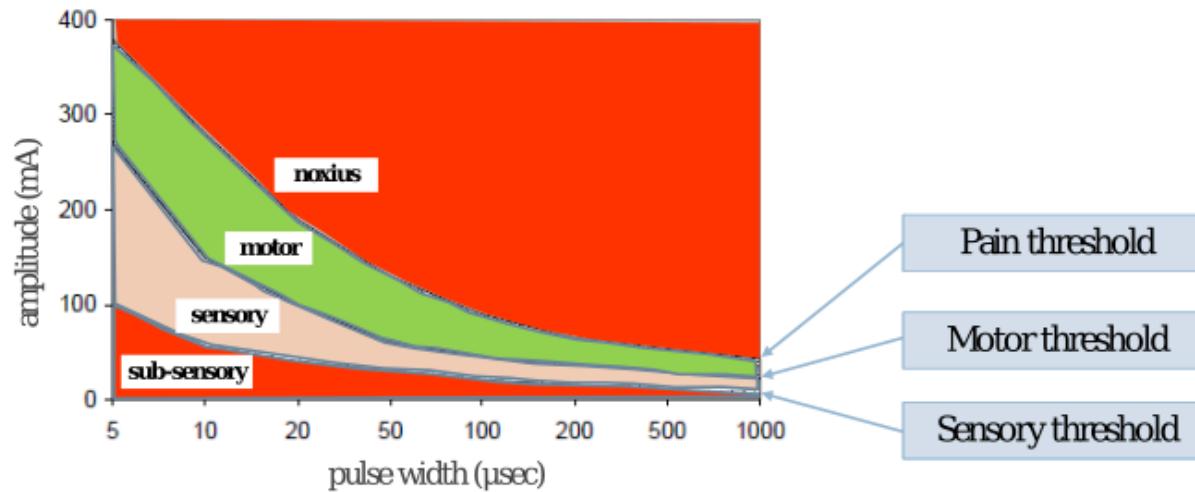
Ramp up (ramp down) (sec)
the time current intensity increases to
(decreases from) a preset intensity

On - off time (sec)
length of time the pulse is delivered
vs. no stimulation

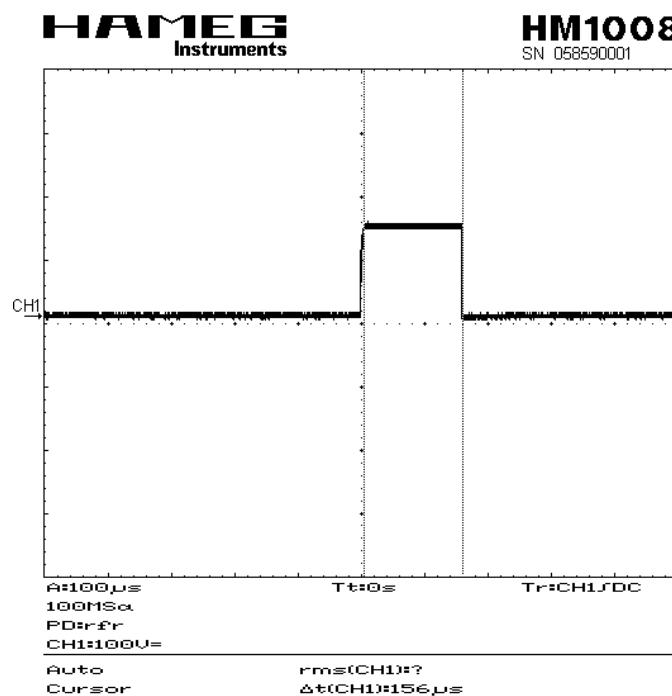
Mstим плокштес

- Sukurtos J. Urbono operai "Honey, Moon", plokštės autorius: D. Snegin:
<https://vimeo.com/277541049>
Panašūs darbai: *Clair/Obscur* (2014), <https://www.dailymotion.com/video/x28hzhn>
- Daito Manabe performansas (<https://www.youtube.com/watch?v=YxdIYFCp5lc>)





Rdinson A, in *Clinical Electrophysiology* 2008, 71-105; Cipriano et al., *J Appl Physiol*, 2014, 117, 633-8





Darbu sauga

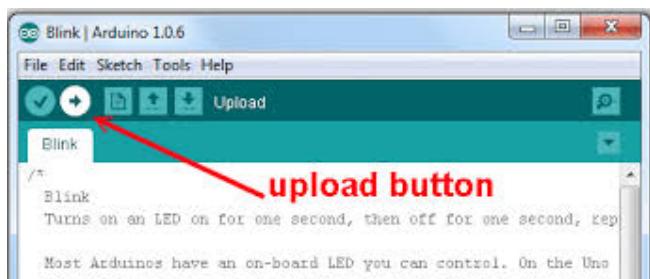
- Visada išjungti plokštės maitinimą prieš ką nors perjungiant
- Du kartus pasitikrinti +/- maitinimo laidus
- Nejungti stimuliavimo elektrodų arti širdies, prie galvos
- Neužtrumpinti laidų ir gnybtų tarpusavyje
- **Atsargiai** reguliuoti mStim plokštės įtampa
- Prasminga laidų spalva: + šiltos, - šaltos

Mstim plokščių jungimas

- Maitinimo laidai, iš kairės: -, +
- Signalo valdymas (ta pati jungtis, dešinėje: Arduino 5 pin'as (viršuje), Arduino Gnd pin'as (apačioje)
- Išėjimo elektrodai (gnybtais) (2*2 išėjimai kiekvienoje plokštėje)
- Du kartus patikriname
- Nusivalome odą, užsidedame elektrodus bent 5 cm atstumu

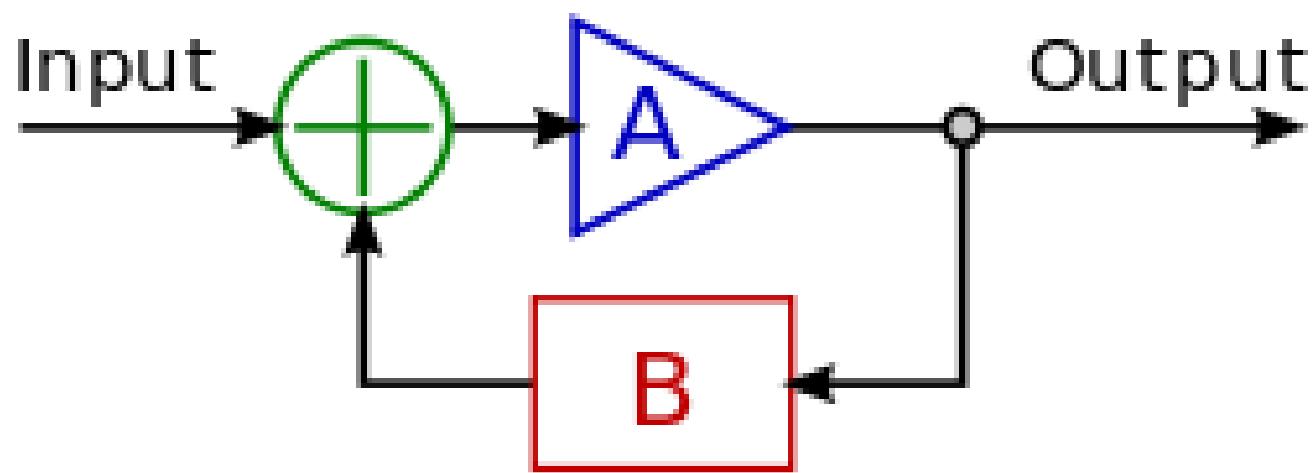
Arduino užprogramavimas

- <https://github.com/Technariumas/MyoElectrics/archive/master.zip>
- Atsidarome test/test.ino su Arduino IDE, pažiūrime į kodą
- Ctrl+U arba Upload mygtukas



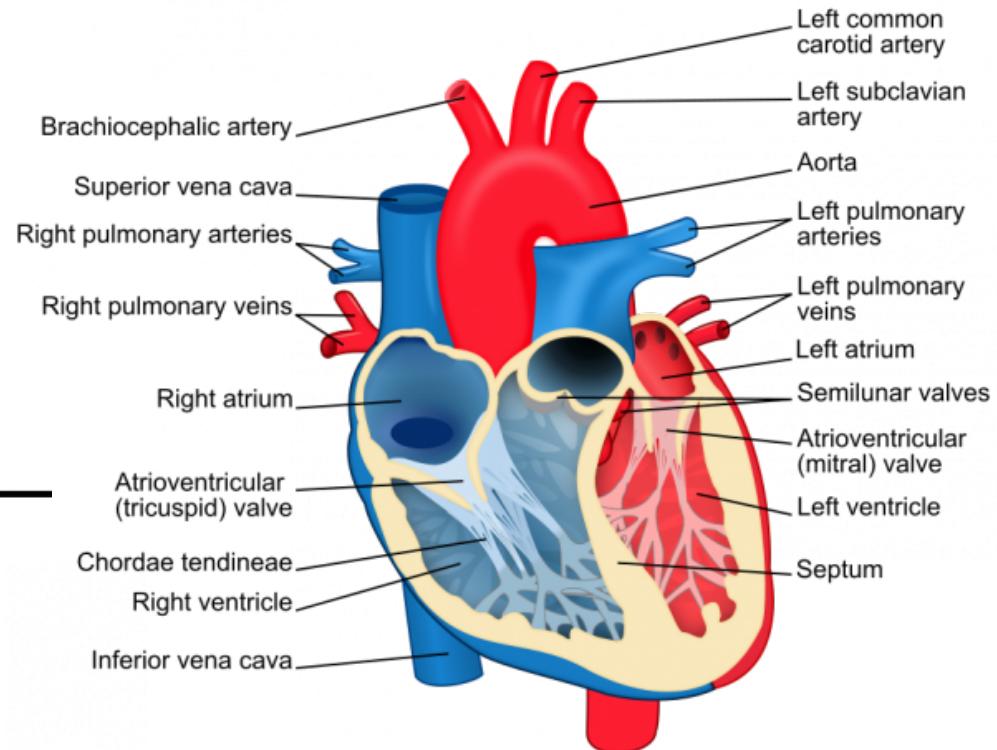
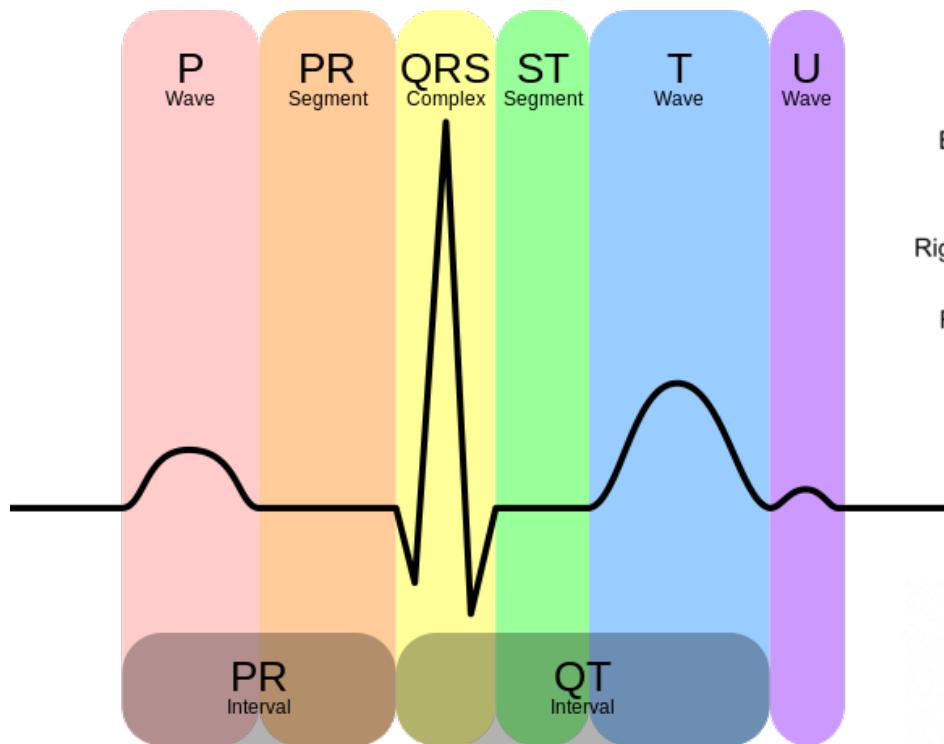
- Ar mirksi LED'as?
- Ijungiamo Mstym maitinimo gnybtus
- Reguliuojame dešinį potenciometrą (S 501 00619) prieš laikrodžio rodyklę, kol signalas tampa jaučiamas.

Gr̄ztamasis ryšys



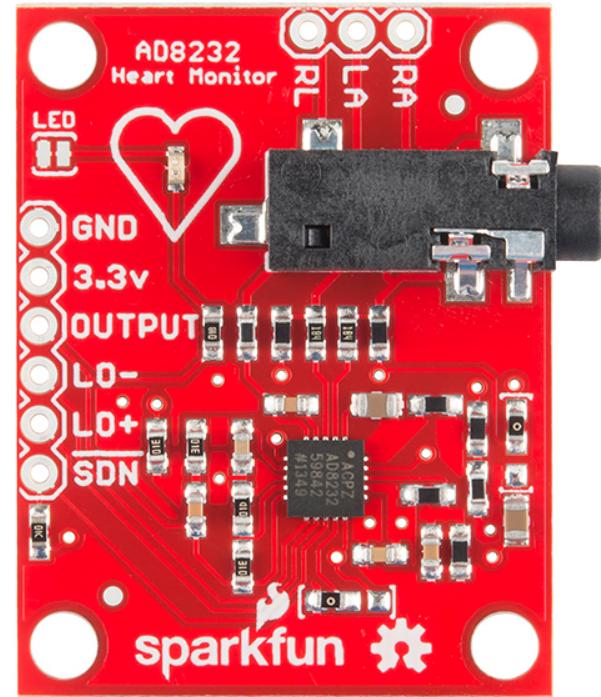
EKG jutiklis

- EKG (ECG, electrocardiography)
- Sparkfun AD8232 jutiklis,
<https://learn.sparkfun.com/tutorials/ad8232-heart-rate-monitor-hookup-guide>

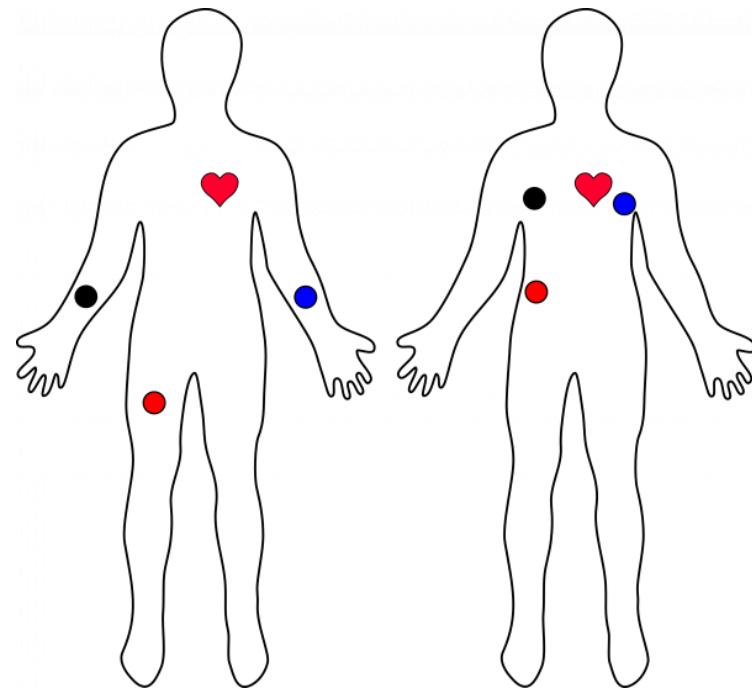


Jungimas

- Gnd – Gnd
- 3.3V – 3.3V
- LO- – 11 Arduino pin'as
- LO+ – 12 Arduino pin'as
- Output – A0



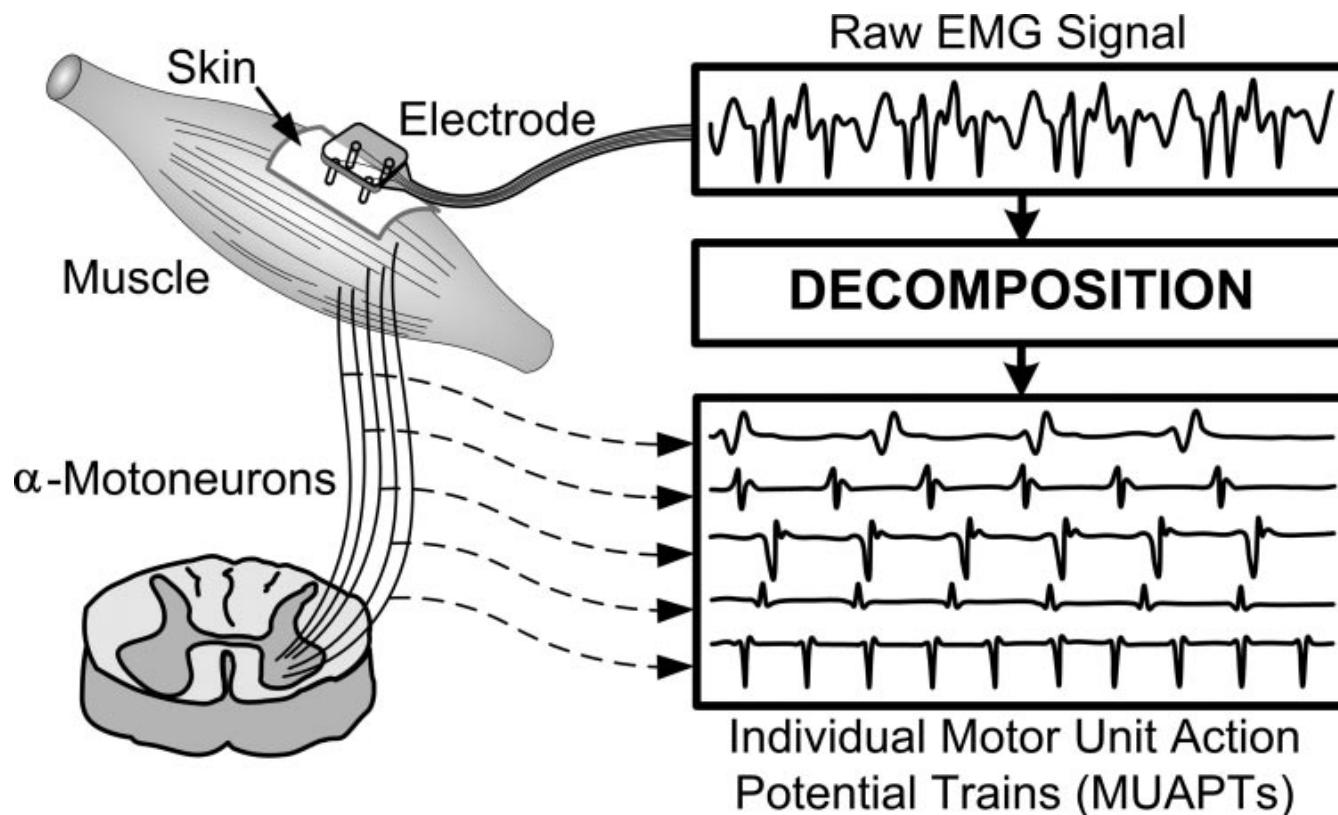
EKG sensoriai ir Arduino kodas



- File > Open > heart/heart.ino
- Įkeliame programą..
- Ctrl+Shift+L

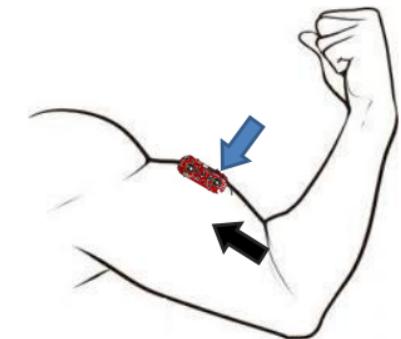
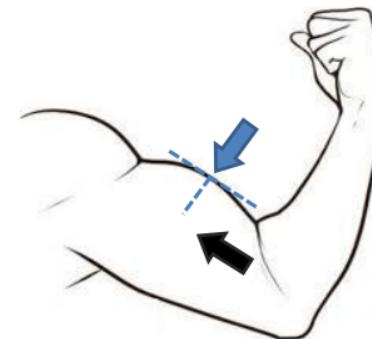
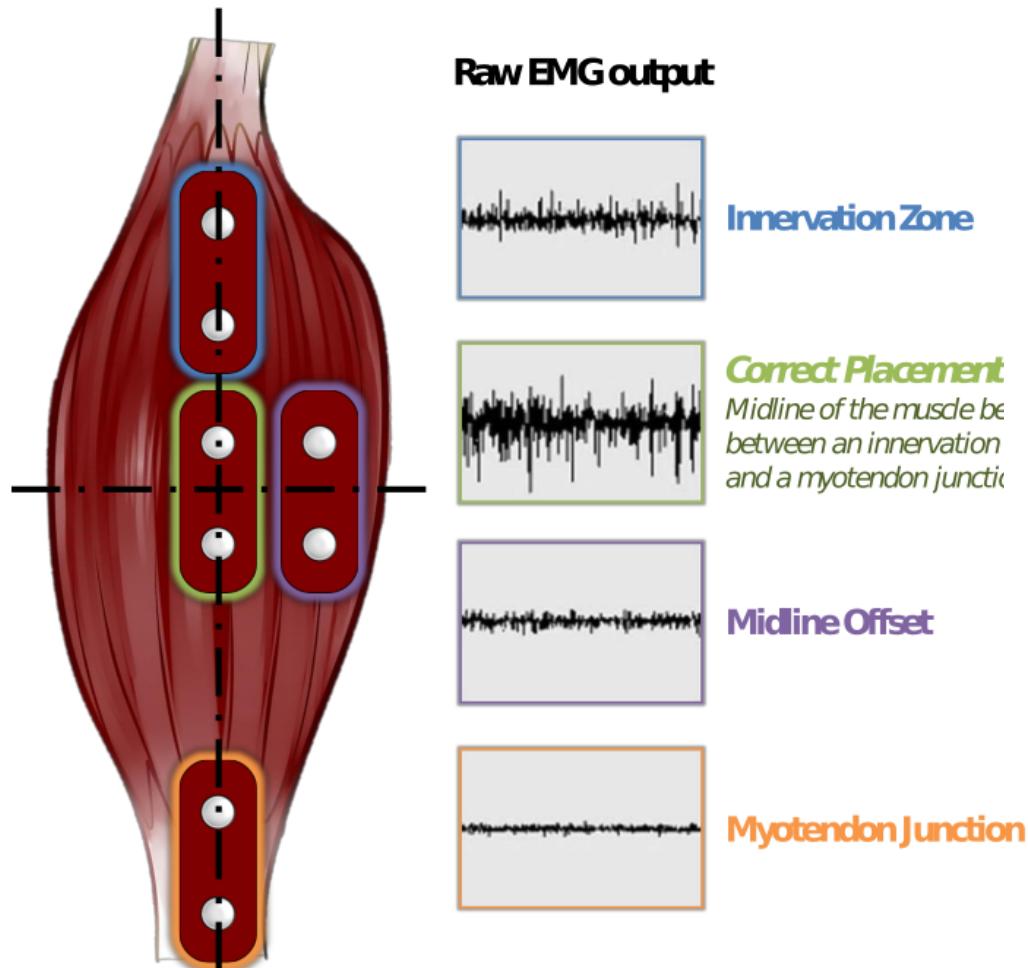
MyoWare EMG sensorius

- EMG: elektroneuromiografija
- Raumenų aktyvumo stebėjimas matuojant labai silpnus potencialo skirtumus raumenyje



Elektrodų vietas parinkimas

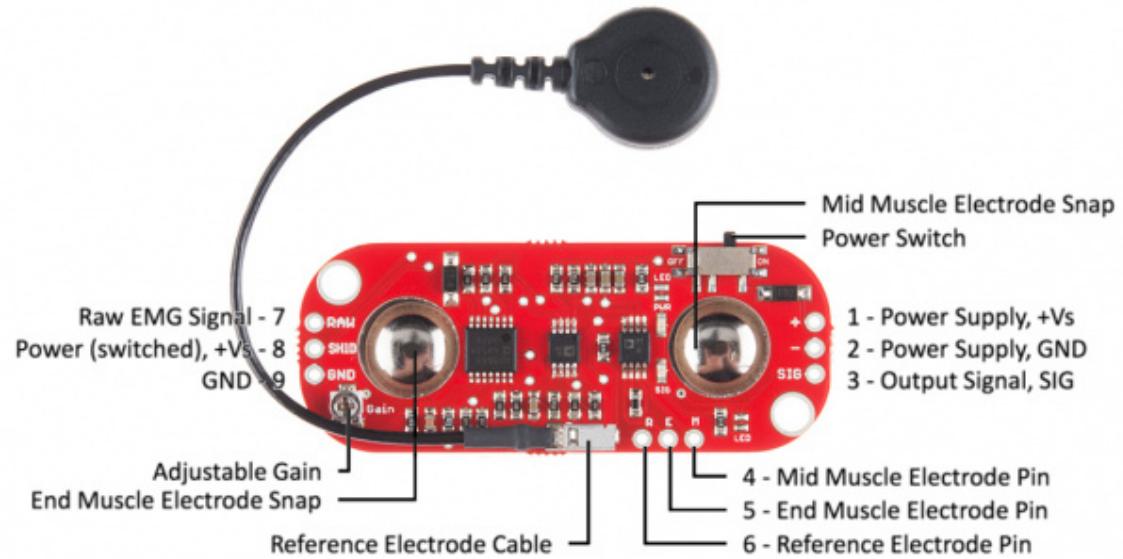
Why is electrode placement important?



Example Sensor Location for Bicep

Jungimas

- + → 5V
- - → Gnd
- SIG → A0



- File > Open: *emg/emg.ino*

Arduino kodas

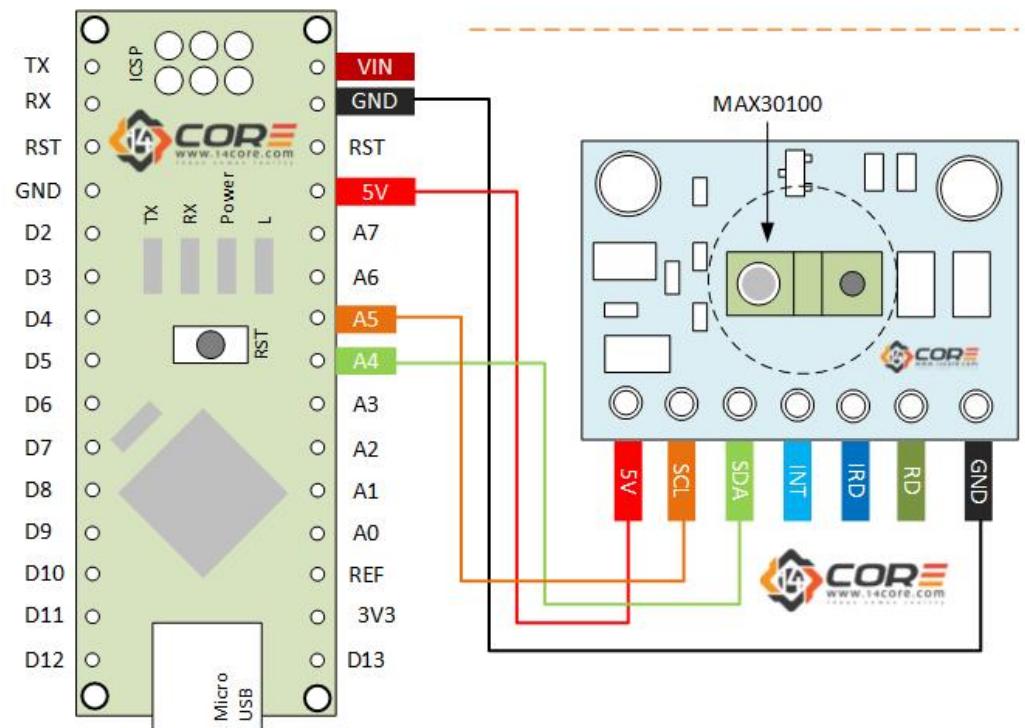
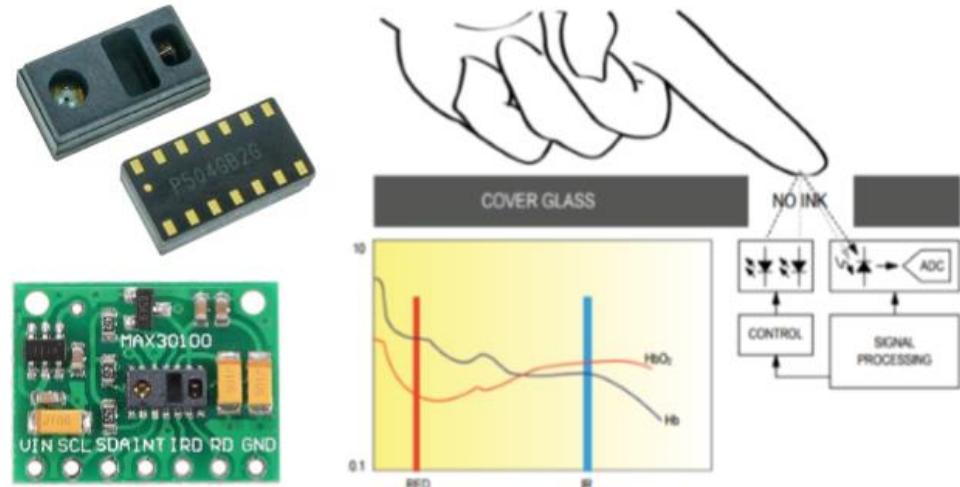
- Tools > Serial Plotter (arba Ctrl+Shift+L)
- Abi programos: susirandame kintamajį *threshold*, koreguojame jo vertę:
int threshold = 600;
- Grj̄tamasis ryšys: atkomentuojame eilutę
//digitalWrite(volt, HIGH); //switch Mstim board on
- Sukeliame pakoreguotą programą, įjungiamo maitinimą

MAX 30100

- Širdies pulso/deguonies kiekio kraujyje sensorius
- IR ir raudonos šviesos absorbcija
- Laikomas, pvz., ant piršto
- Kodas: *pulse/pulse.ino*

Jungimas

- VIN → 5V
- GND → GND
- SCL → A5
- SDA → A4



Keletas idėjų

- Grįztamasis ryšys iš kito žmogaus
- Visa kita, ką galima prijungti prie Arduino
- Elektrostimuliacijos stiprumo reguliavimas (PWM)
- Valdymas potenciometu
- Sonifikavimas (Pure Data)

Daugiau resursų

- Elektronikos pirmadieniai Technariume
- <http://arduino.cc> – dokumentacija
- Adafruit Arduino pamokos
- <http://stackoverflow.com>

