Progetto Arduino mano robotica

TUTORIAL ARDUINO #12 – ROBOTIC HAND

[http://www.marcopucci.it/tutorial-arduino-12-robotic-hand/Arduino](http://www.marcopucci.it/category/arduino/)

[26 giugno 2014](http://www.marcopucci.it/tutorial-arduino-12-robotic-hand/) [0](http://www.marcopucci.it/tutorial-arduino-12-robotic-hand/#comments)

Share it !

**Tutorial #12 – Robotic Hand**

Materiale utilizzato – Arduino UNO – 5 servo motori – 5 sensori di flessione – 5 resistenze – materiale per costruire la mano robotica (tubi, carta, colla, filo, etc…) – guanto.

**Per vedere tutti i tutorial:**[**www.marcopucci.it/arduino/**](http://www.marcopucci.it/arduino/)

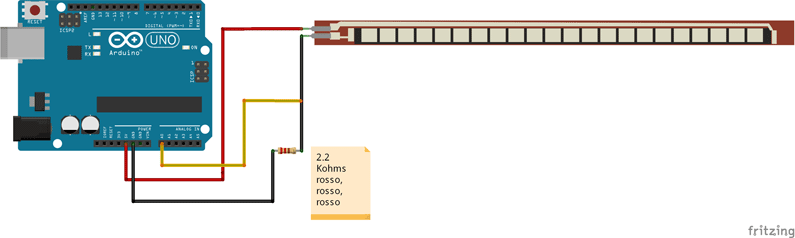
[](https://www.facebook.com/pages/Tutorial-Arduino/1377902315765530?ref=hl)  
Segui la pagina Tutorial Arduino su Facebook

**N.B:**[**Clicca qui**](http://www.marcopucci.it/tutorial/Robotic_Hand.zip)**per scaricare il file .zip con tutti gli sketch utilizzati per realizzare la mano robotica.**

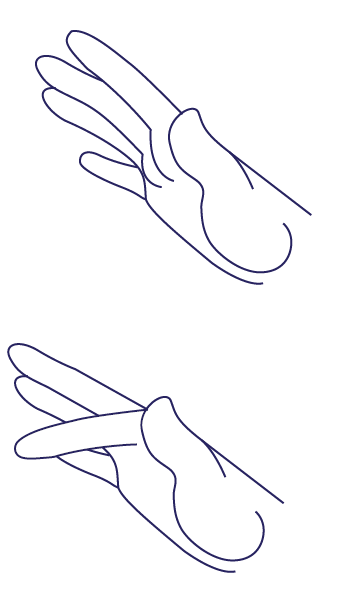
In questo tutorial realizziamo una mano robotica (low cost) in grado di replicare i movimenti della nostra mano.  
Inseriamo dei sensori di flessione sulle dita di un guanto, analizziamo il valore di questi sensori e muoviamo dei servo motori 180° che tirano un filo collegato a ciascun dito della mano robotica.  
La parte elettronica consiste quindi nella creazione di un guanto con sensori e la mano robotica con i cinque motorini.  
Se non avete mai utilizzato i servo motori vi consiglio di leggere il tutorial n. 8 – clicca qui.  
Procediamo con la realizzazione del guanto.

[](http://www.marcopucci.it/wp-content/uploads/2014/06/SENSORE_DI_FLESSIONE.jpg)

I sensori di flessione (immagine sopra) sono delle piccole barrette flessibili che indicano la quantità di flessione a cui sono sottoposti. Attaccandoli alle dita del guanto sappiamo esattamente il movimento di ciascun dito.  
I sensori di flessione devono essere collegati ad Arduino con delle resistenze. Possono essere acquistati online sul sito di Robot-Italy.  
Sensore: [www.robot-italy.com/it/flex-sensor-2-2.html](http://www.robot-italy.com/it/flex-sensor-2-2.html) – 8 euro  
Resistenze: [www.robot-italy.com/it/resk-kit-500-resistenze.html](http://www.robot-italy.com/it/resk-kit-500-resistenze.html) – 5 euro (kit da 500)  
Altrimenti potete acquistare in qualsiasi negozio di elettronica delle resistenze da 2.2 Kohms  
**Questo tutorial è stato realizzato nel 2014, prima di procedere all’acquisto online verificate se altri siti più convenienti vendono i sensori di flessione. Attualmente questo segnalato è il più economico.**

[](http://www.marcopucci.it/wp-content/uploads/2014/06/flex_sensor_1_bb.png)

Il sensore di flessione deve essere collegato in questo modo: la barra laterale più stretta deve essere alimentata con i 5V di Arduino. La barra più grande deve essere collegata al GND di Arduino passando per una resistenza. Il cavo che arriva al pin analogico di Arduino (serve per leggere il valore del sensore) deve essere saldato prima della resistenza (guardare lo schema qui sopra). A cosa serve la resistenza?  
In questo caso la resistenza serve per diminuire e rendere leggibile il valore del sensore.  
Inserendo una resistenza di 2.2 Kohms avremo come lettura analogica un valore uguale a 50 se il sensore non è piegato, uguale a 90 se il sensore è flesso in una direzione e un valore uguale a 10 se il sensore viene piegato nel senso inverso.  
Qui sotto un piccolo schema per vedere i valori del sensore di flessione.  
**N.B: se utilizzate un altro tipo di resistenza questi valori saranno diversi.**

[](http://www.marcopucci.it/wp-content/uploads/2014/06/hands.png)

VALORE SENSORE = 50

VALORE SENSORE = 10

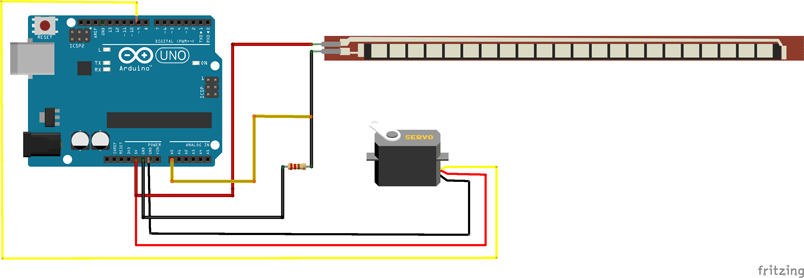
Iniziamo con un solo sensore. Realizzate il circuito elettrico della pagina precedente e aprite il software di Arduino per iniziare a programmare. Per il momento non utilizziamo né il guanto né la mano robotica.  
L’assemblaggio verrà realizzato alla fine, quando il codice e il circuito di un singolo dito saranno completati.  
**INFORMATICA**

Per testare il funzionamento del sensore apriamo il file presente all’interno del programma di Arduino per la lettura analogica del sensore (FILE – ESEMPI – BASIC – ANALOGREADSERIAL).  
Questo sketch permette di leggere qualsiasi sensore posizionato nel pin 0 di Arduino.  
Se non avete mai visto questo codice fermatevi qui e iniziate tutto il corso di Arduino dal tutorial n1 – [clicca qui](http://www.marcopucci.it/tutorial-arduino-1-cose-arduino/)

Uplodate il codice sulla scheda e aprite il Serial Monitor.  
Tenendo il sensore fermo il numero visualizzato dovrebbe essere un 50.  
Piegandolo in avanti raggiungete il numero 10, piegandolo nel senso inverso il numero 90.  
**Se avete utilizzato una resistenza diversa il valore cambia.**

Segnatevi i valori del sensore fermo e piegato in avanti. Dovrete sostituirli a quelli che utilizzerò io nel tutorial.

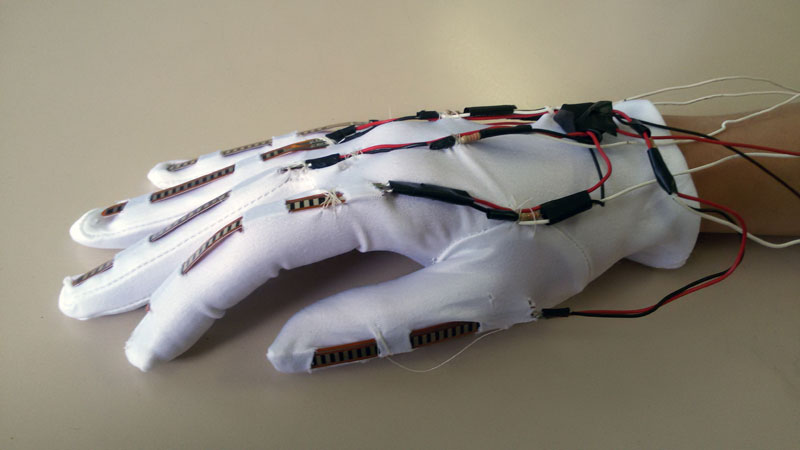
**COMANDO MAP (Aprite lo sketc \_2\_1\_sensore\_1\_motore\_val)**  
Passiamo ora ai motori servo.  
Colleghiamo un servo ad Arduino seguendo lo schema qui sotto.

[](http://www.marcopucci.it/wp-content/uploads/2014/06/flex_sensor_1_motor_1.png)

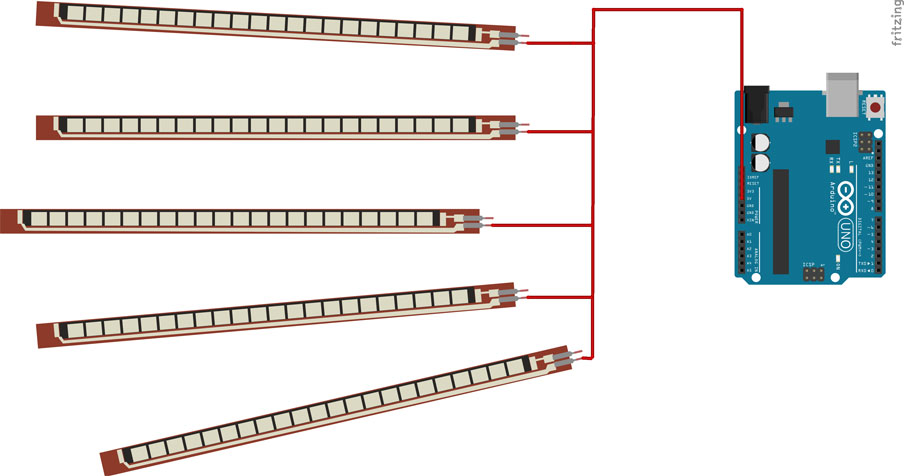
Quando il sensore di flessione verrà piegato, il motorino si muoverà in maniera proporzionale.  
Abbiamo già visto che con il comando myservo.write(90); possiamo muovere il nostro servo in un angolo qualsiasi compreso tra 0 e 180°. Esiste un comando nel programma di Arduino che ci permette di passare un valore al motorino in proporzione a quello che avviene nel sensore.  
Il sensore ci invia un dato che è compreso tra 90 e 10. Il motorino ha bisogno di un valore tra 0 e 180.  
Il comando **map** esegue il rapporto tra i due valori. Es: **valore = map(valore, 90, 10, 0, 179);**  
I primi due numeri si riferiscono al flex sensor mentre gli altri due al servo. In base al numero in entrata abbiamo un valore adatto al motorino.  
Provate a caricare il codice qui sotto per capire meglio il funzionamento del comando MAP (oppure apri il file**\_2\_1\_sensore\_1\_motore\_val**)

#include <Servo.h>  
Servo myservo;  
int indice;  
void setup() {  
Serial.begin(9600);  
myservo.attach(9);  
//Attacchiamo il servo al pin 9  
}  
void loop() {  
indice = analogRead(A0);  
Serial.print(“valore originale:”);  
Serial.println(indice);  
indice = map(indice, 90, 10, 0, 179);  
myservo.write(indice);  
Serial.print(“valore scalato:”);  
Serial.println(indice);  
//Stampiamo il valore originale del sensore e poi quello modificato dal comando map  
delay(150);  
}

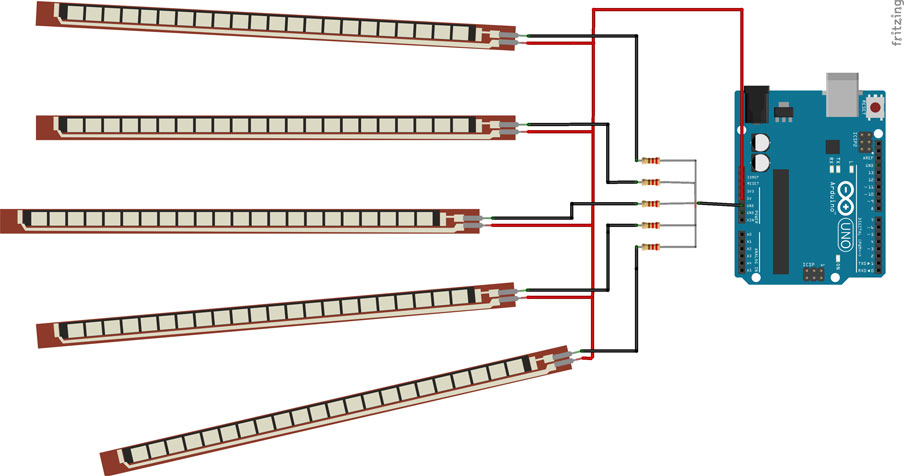
Se pieghiamo il sensore il motorino si muove e tira il filo a cui è collegato il dito della mano robotica.  
Iniziamo a realizzare il guanto e la mano robotica.  
Prendiamo un guanto e attacchiamo il sensore sul dito indice. Il modo migliore per fissarlo è cucirlo o fissarlo con del nastro isolante.  
Se possibile saldiamo dei lunghi cavi elettrici al sensore e colleghiamoli con Arduino.  
Stiamo ancora realizzando un prototipo molto elementare, una volta capito il funzionamento e la quantità di cavi che dobbiamo utilizzare penseremo a montare il guanto definitivo.

[](http://www.marcopucci.it/wp-content/uploads/2014/06/robot_hand_1.jpg)

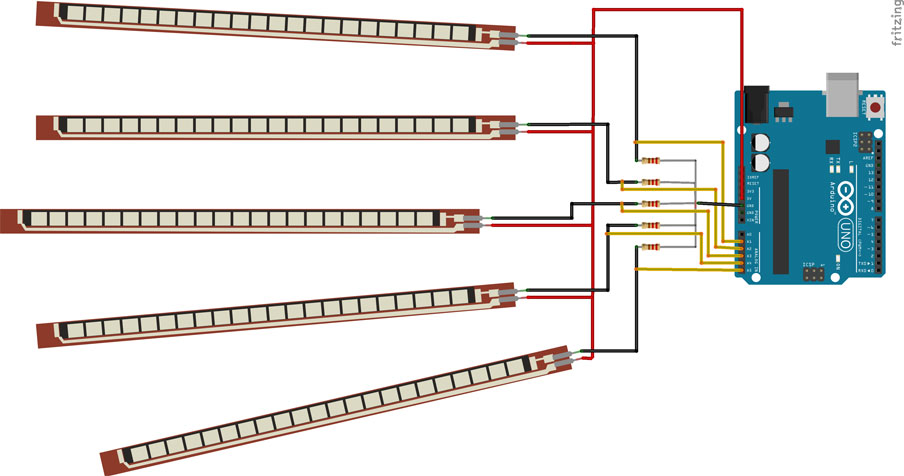
Seguendo lo schema elettrico saldiamo cinque fili rossi ai pin positivi del sensore di flessione. Le estremità dei cavi rossi devono essere saldate insieme sul dorso della mano. Ai 5 fili ne aggiungiamo uno più lungo che colleghiamo al pin 5V di Arduino.

[](http://www.marcopucci.it/wp-content/uploads/2014/06/flex_sensor_5_dita_nero.jpg)

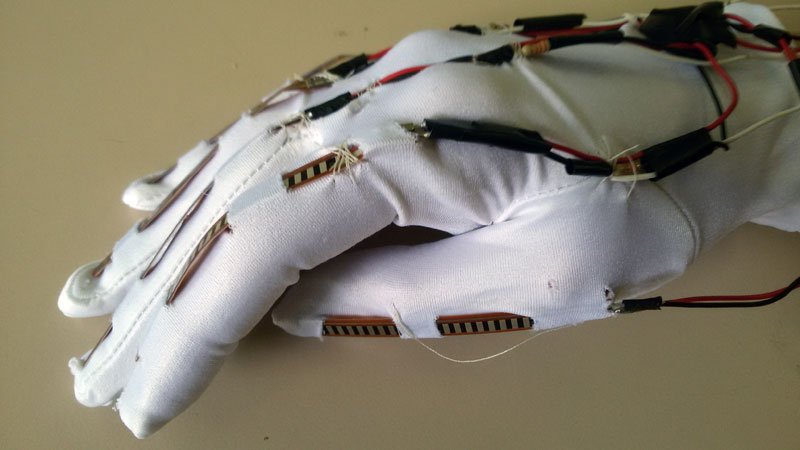
Prendiamo 5 piccoli fili neri e li saldiamo all’altra estremità del sensore di flessione. Per ogni cavo nero dobbiamo saldare una resistenza. All’altra estremità della resistenza saldiamo altri 5 fili neri che si uniscono al centro della mano. Da questi ultimi parte un ultimo filo nero più lungo che inseriamo nel Pin GND di Arduino.

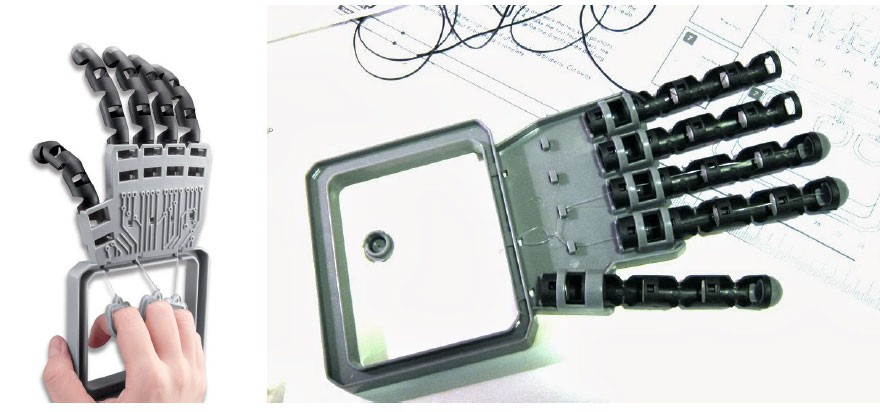
[](http://www.marcopucci.it/wp-content/uploads/2014/06/flex_sensor_5_dita_giallo.jpg)

A questo punto dobbiamo collegare 5 fili lunghi gialli. Ogni filo giallo deve essere saldato nella prima estremità della resistenza e deve essere collegato al pin analogico di Arduino. Ciascuno dei fili gialli finirà dunque nei pin A0, A1, A2, A3 e A4. Questi fili invieranno ad Arduino i dati di ciascun sensore di flessione.

[](http://www.marcopucci.it/wp-content/uploads/2014/06/flex_sensor_5_dita_bb.jpg)

[](http://www.marcopucci.it/wp-content/uploads/2014/06/robot_hand_arduino.jpg)

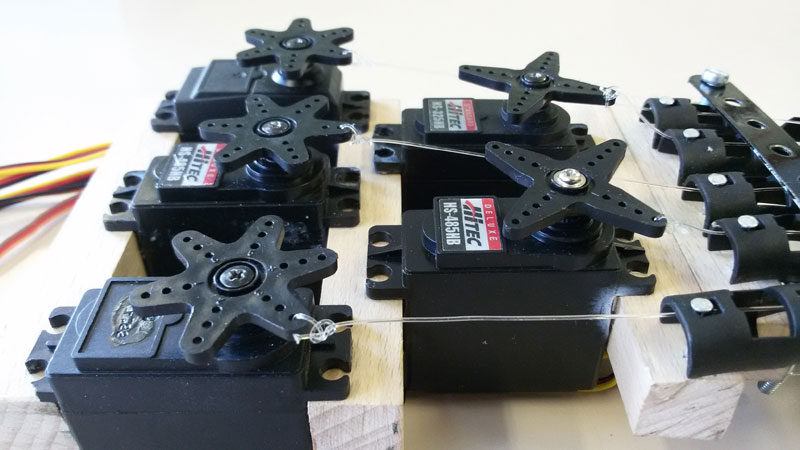
[](http://www.marcopucci.it/wp-content/uploads/2014/06/guanto-interattivo.jpg)

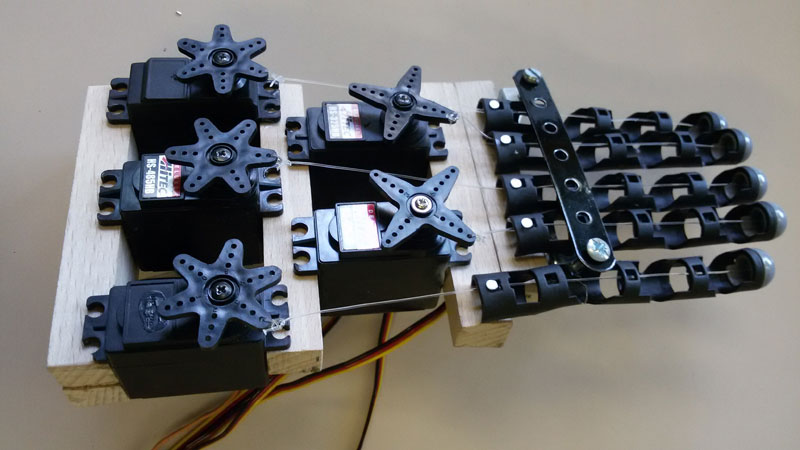
Passiamo ora alla mano robotica.  
Realizzare la mano è la parte più difficile dell’intero tutorial.  
Segnalo questo tutorial (clicca qui) realizzato molto bene. Lo scopo è quello di creare una mano con le dita collegate a dei fili. Quando il motorino tira il filo, il dito si piega.  
Se non volete costruire la mano robotica da soli potete acquistare questo giocattolo della 4M:  
Robotic Hand Kit – [www.amazon.com/4M-3774-Robotic-Hand-Kit/dp/B005MK0OPO](http://www.amazon.com/4M-3774-Robotic-Hand-Kit/dp/B005MK0OPO)  
[](http://www.marcopucci.it/wp-content/uploads/2014/06/robot_hand_4m.jpg)

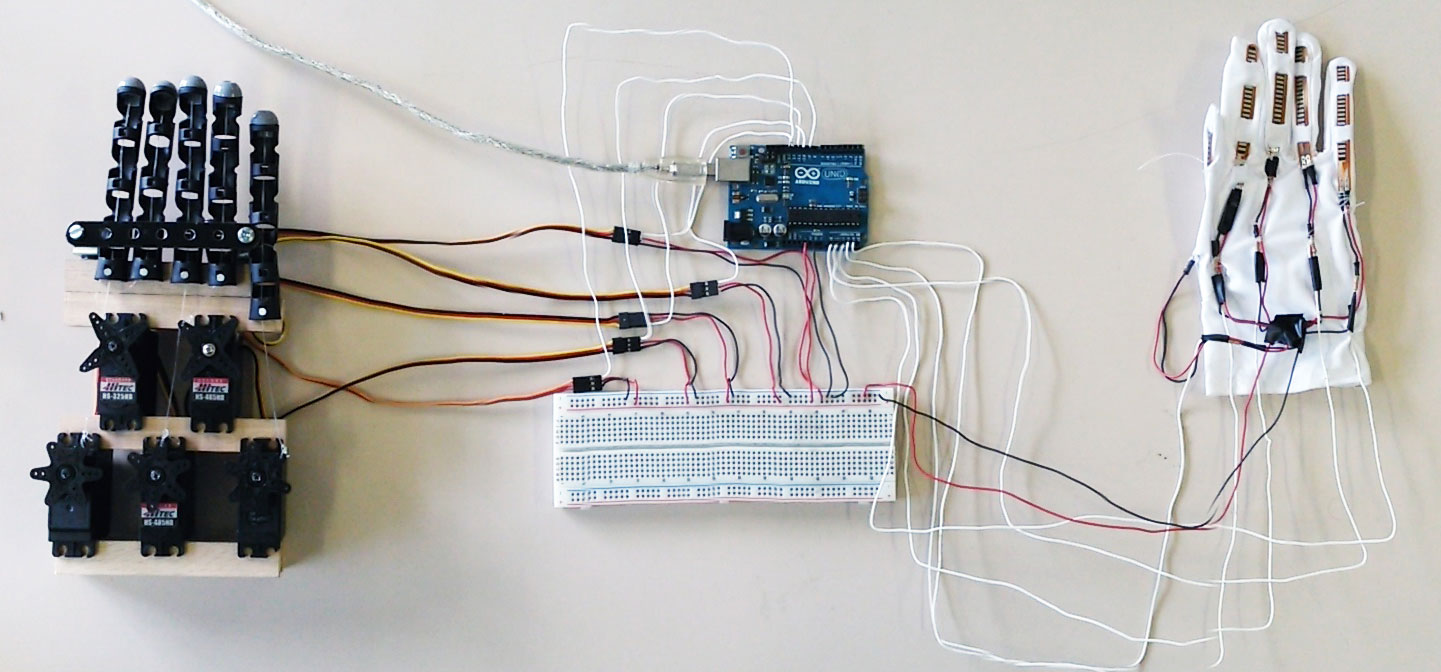
Una volta montata basta collegare i vari fili ai 5 motorini.  
La realizzazione della mano la lascio alla vostra fantasia. Inviatemi via email ([puccimarco76@yahoo.it](mailto:puccimarco76@yahoo.it)) le vostre creazioni, le pubblicherò sul sito e sulla pagina  
Facebook dei Tutorial di Arduino:  
<https://www.facebook.com/pages/Tutorial-Arduino/1377902315765530?fref=ts>.

Qui sotto alcune immagini della mano robotica realizzata con una struttura di legno e le dita del giocattolo presentato qui sopra.  
La struttura è la parte fondamentale della mano. I cinque motorini non devono muoversi (incollateli a dei tasselli di legno o fissateli con delle viti a delle barre di metallo).  
Con del filo da pesca colleghiamo ogni motorino al dito costruito e assembliamo la parte elettronica dei 5 motorini.

[](http://www.marcopucci.it/wp-content/uploads/2014/06/mano_robotica.jpg)

[](http://www.marcopucci.it/wp-content/uploads/2014/06/mano_robotica_arduino.jpg)

[](http://www.marcopucci.it/wp-content/uploads/2014/06/mano_robotica2.jpg)

[](http://www.marcopucci.it/wp-content/uploads/2014/06/mano_robotica_guanto-arduino.jpg)

**(Aprite lo sketc \_4\_tutti\_sensori\_tutti\_motori\_val)**

Di seguito il codice completo per gestire i 5 sensori e i 5 motorini.  
I valori da inserire nel comando MAP, es. map(pollice, 20, 40, 0, 179);  
possono variare in base al vostro sensore, alla vostra resistenza, al tipo di motorini, ecc…  
Vi consiglio di fare delle prove modificando i numeri fino a trovare il giusto compromesso tra movimento del guanto e risposta della mano robotica.

#include <Servo.h>  
Servo myservo\_pollice;  
Servo myservo\_indice;  
Servo myservo\_medio;  
Servo myservo\_anulare;  
Servo myservo\_mignolo;  
int pollice;  
int indice;  
int medio;  
int anulare;  
int mignolo;

void setup() {  
Serial.begin(9600);  
myservo\_pollice.attach(9);  
myservo\_indice.attach(10);  
myservo\_medio.attach(11);  
myservo\_anulare.attach(12);  
myservo\_mignolo.attach(13);  
}

void loop() {  
pollice = analogRead(A0);  
indice = analogRead(A1);  
medio = analogRead(A2);  
anulare = analogRead(A3);  
mignolo = analogRead(A4);  
Serial.println(mignolo);

pollice = map(pollice, 20, 40, 0, 179);  
indice = map(indice, 20, 40, 0, 179);  
medio = map(medio, 20, 40, 0, 179);  
anulare = map(anulare, 20, 40, 0, 179);  
mignolo = map(mignolo, 20, 40, 0, 179);

myservo\_pollice.write(pollice);  
myservo\_indice.write(indice);  
myservo\_medio.write(medio);  
myservo\_anulare.write(anulare);  
myservo\_mignolo.write(mignolo);

delay(150);  
}

**Ti è piaciuto il tutorial? Se ti va offrimi un caffè…**

Inizio modulo

 https://www.paypal.com/en_US/i/scr/pixel.gif

Fine modulo

I tutorial sono liberamente scaricabili e condivisibili sotto l’etichetta Creative Commons (non è possibile commercializzarli e modificarli)

[icona_pdf](http://www.marcopucci.it/wp-content/uploads/2014/07/12_robotic_hand.pdf)  
Scarica il Pdf del tutorial di Arduino  “Robotic Hand”

<http://www.maffucci.it/2014/11/15/edurobot-uno-come-costruire-il-vostro-primo-arduino-robot-lezione-2/>

<http://www.maffucci.it/2014/11/15/edurobot-uno-come-costruire-il-vostro-primo-arduino-robot-lezione-3/>

Vorrei provare a realizzare qualcosa con Arduino.  
Stavo pensando di provare a muovere un braccio robotico  
da una postazione desktop via usb (e poi wireless, ma cominciamo con il desktop usb).  
  
Come braccio robotico pensavo a qualcosa del genere:  
**http://www.robot-italy.com/index.php?cPath=11\_20**  
  
Dovrei poi procurarmi Arduino Uno.  
  
Ed ecco le domande:  
è molto difficile collegare Arduino al braccio meccanico?  
Ho qualche speranza di riuscire nel progetto?  
  
ciao a tutti e grazie in anticipo per le risposte!

**Quali strumenti avete utilizzato?**  
«Trapano, flex, lima, saldatrice. E tanta pazienza, costanza, passione e determinazione. Non ho mai smesso di crederci».  
  
**Quali riscontri ha avuto?**  
«Ho inviato il mio progetto a un sito sul "fai da te" americano e non solo ho avuto moltissime visualizzazioni, ma anche ricevuto molte domande di carattere tecnico. Su questo stesso sito ho partecipato a un contest, **dove sono arrivato terzo, vincendo una fotocamera**. Da qui la notizia si è diffusa sul web e sono stati pubblicati articoli e notizie su diversi siti in tutto il mondo. Ho trovato persino un articolo in russo. Questo mi ha dato molta visibilità tanto che **sono stato contattato da Open Biomedical Foundation**, associazione no profit attiva nel settore biomedicale, che vuole sviluppare prodotti a basso costo e open source soprattutto in favore di paesi svantaggiati dal punto di vista economico, **rendendo dunque la tecnologia accessibile a tutti e prodotti ad alto contenuto tecnologico** facilmente riproducili. Sul mio progetto sono anche usciti degli articoli **sul blog di Arduino e sul blog di Mauro Alfieri,** che cura un blog dedicato al mondo open source e ad Arduino al quale mi sono rivolto spesso in corso d'opera per avere suggerimenti e consigli. Anche il sito wevolver dedicato sempre alle tecnologie open source ha mostrato interesse

### INTRODUCTION

The Cyborg Beast is a remix of the snap together Robohand (<http://www.thingiverse.com/thing:92937>) and the original Robohand (<http://www.thingiverse.com/thing:44150>).

The “Cyborg Beast” hand has been created as a collaboration between Jorge Zuniga, Ivan Owen and Peter Binkley. All files for this design can be found on [Thingiverse](http://www.thingiverse.com/thing:261462" \t "_blank).

Most voluntary closing devices including this design need sufficient wrist movement and strength for proper function. This design doesn't require Orthoplastic. The fingers are designed for better gripping and to avoid over extension of the proximal phalange and distal finger segment. It fits a 1 to 2 mm cabling and elastic bands.

For more information on where to connect with people who can help someone in need with sizing, printing and/or assembly of a mechanical hand please join the e-NABLE Google+ Community entitled, e-NABLE:  
<https://plus.google.com/communities/102497715636887179986>  
For more information and stories:  
[http://enablingthefuture.org](http://enablingthefuture.org/)  
<https://www.facebook.com/enableorganization>

## Description

Here is "InMoov", the robot hand you can print and animate. You have a 3D printer, some building skills, This project is for you!!  
This is a hand built for a job (still picture commercial), it was supposed to be mobile but not animated and mostly be able to take human hand positions. Cables or fishing rods might be added in order to control it, but it was not the purpose of this work at first. I am now working on another model(for the fun), trying to animate it, I really would like to do it with computer control but I have never worked in that field. Never used servo's before, so any help is welcomed.  
Blog:  
[**http://inmoov.blogspot.com**](http://inmoov.blogspot.com/)  
Site:  
[**http://www.inmoov.fr**](http://www.inmoov.fr/)  
A first movement video. I have to say that this video was not easy to do. I was holding the camera with one hand and trying to trigger the rods with the other.  
I definitly need three hands!  
[**http://www.youtube.com/watch?v=r\_rQ2WJ0dq4**](http://www.youtube.com/watch?v=r_rQ2WJ0dq4)  
[**http://www.youtube.com/watch?v=JztSQpwZaRU**](http://www.youtube.com/watch?v=JztSQpwZaRU)  
[**http://www.youtube.com/watch?v=zvbNcuUPHyQ**](http://www.youtube.com/watch?v=zvbNcuUPHyQ)  
[**http://www.youtube.com/watch?v=BAs2F4sFVdA**](http://www.youtube.com/watch?v=BAs2F4sFVdA)  
Last of all:  
[**http://www.youtube.com/watch?v=tojIdfywYVI**](http://www.youtube.com/watch?v=tojIdfywYVI)  
A demo using Leap motion to control the hand:  
[**http://youtu.be/GOTRXM4O9L0**](http://youtu.be/GOTRXM4O9L0)  
There is also the left hand model for the one that plan to make a complete robot, Arf, still quite a bite of work though.  
[**http://www.thingiverse.com/thing:18939**](http://www.thingiverse.com/thing:18939)

<http://www.instructables.com/id/Assembly-of-e-NABLEs-Raptor-Reloaded-Hand/>

## Description

The Raptor Reloaded is an updated release of the original that incorporates feedback from the e-NABLE community and has CAD design source files available in STEP, IGES, IPT, and F3D formats.

Updates in the 12-17-14 release include:  
-- Improved print-ability due to custom modeled supports  
-- Cyborg Beast compatible sizing  
-- Improved dovetail geometry  
-- New tensioner retention clip  
-- Easier to use tensioner pins  
-- Low profile elastics with two tie-off options  
-- More access to elastic and flexsor routing channels  
-- Debossed versioning information on palm and gauntlet  
-- Slimmer, more anthropomorhpic fingers  
-- Narrower knuckle block  
-- More intuitive assembly (new one-way proximals)  
-- Thorough documentation and modular design  
-- Source files in native Fusion 360 format, also STEP and IGES  
-- Additional velcro-mounting options as well as traditional velcro loop and leather options  
-- Re-oriented snap-pin head recesses to improve printability  
-- Knuckle pins are now removable, making the hand easy to repair and upgrade  
-- Strengthened tie-bars on finger tips

Native CAD source files available here:  
[**Full Assembly**](http://a360.co/1uV2X0G)  
[**Left Palm**](http://a360.co/1wKN4Ra)  
[**Right Palm**](http://a360.co/1GOauFn)  
[**Gauntlet**](http://a360.co/1wKNbMt)  
[**Proximal Phalange**](http://a360.co/1wKNd7f)  
[**Fingertip**](http://a360.co/1wKNhUz)  
[**Finger Snap Pin**](http://a360.co/1wYg08r)

[**Detailed assembly instructions available here.**](http://bit.ly/RaptorReloaded)

Changelog:  
-- 1/4/15 -- updated finger snap pins (raptor\_2.0\_finger\_pin.stl) to fix fit problems. Updated full hand plates with new pin as well. AB  
-- 2/1/15 -- updated cover image to reflect recommended elastic stringing, added fingertip with finger nail that can be painted, added link to detailed print and assembly instructions on Instructables. AB

**Control any gadgets with your Brain! (Neurofeedback with Arduino Microcontroller)**

[[](http://cdn.instructables.com/FTA/05TO/GTZK8E4F/FTA05TOGTZK8E4F.LARGE.jpg)](http://cdn.instructables.com/FTA/05TO/GTZK8E4F/FTA05TOGTZK8E4F.LARGE.jpg)

[[](http://cdn.instructables.com/FF0/9ULF/GTXKH9EJ/FF09ULFGTXKH9EJ.LARGE.jpg)](http://cdn.instructables.com/FF0/9ULF/GTXKH9EJ/FF09ULFGTXKH9EJ.LARGE.jpg)

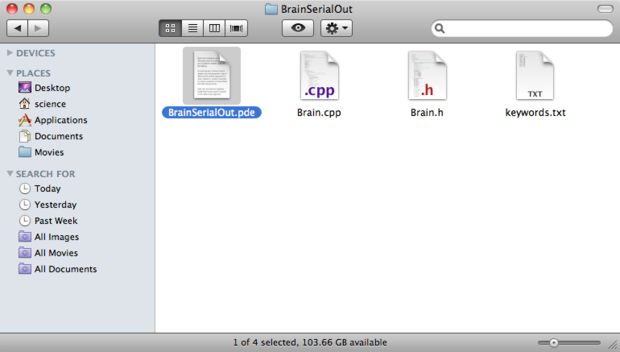
**You've Seen many cool gadgets on news like, controlling robot.  
I've been pretty shaken up with that one. So after some 6 months of research, I made my own!  
All the parts cost me less then 100$! Thats cool isn't it?**  
  
**Materials:  
  
- 1 MindFlex - 50$ <= Got mine from**[**EBay**](http://www.ebay.com/sch/i.html?rt=nc&LH_BIN=1&_nkw=mindflex&_trksid=p3286.c0.m301)**!  
- 1 Arduino Microcontroller <= Visit the official Aruino Site**[**Here**](http://arduino.cc/)**!  
- Some Wires <= Duh! I got it from my old mashed up computer...  
- A Personal Computer <= Everyone has it. Right?  
- Some time  
  
Software & Files:  
  
- Arduino GUI <= Download Right**[**Here**](http://arduino.cc/en/Main/Software)**!  
- Arduino Brain Example Code <= Download at the bottom!  
  
Tools:  
  
- Soldering Iron <= You'll need it.  
- Hot Glue gun <= Not REALLY needed, but we have to secure wires right?  
- Drill <= Not really needed, its used to make a hole so wire can go through the case  
- Other tools (Pliers, Wire Cutters, Wire Strippers, etc.) <= This will help ya!  
  
Well, Thats about it! Lets move on!**

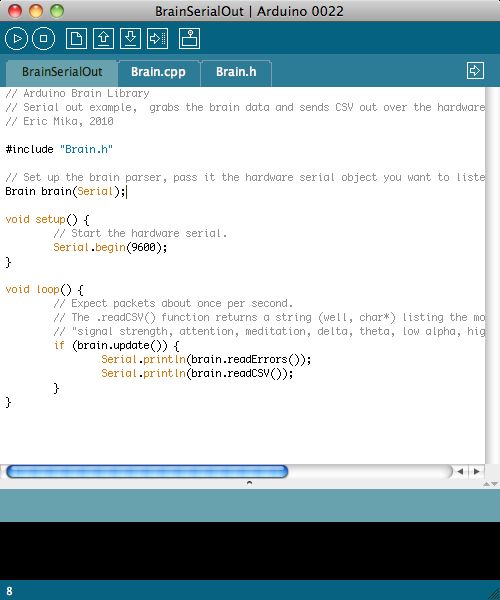
[[http://www.instructables.com/static/defaultIMG/file/ZIP.gif](http://www.instructables.com/files/orig/FFC/JM9C/GTZK8EJI/FFCJM9CGTZK8EJI.zip)**Brain.zip**](http://www.instructables.com/files/orig/FFC/JM9C/GTZK8EJI/FFCJM9CGTZK8EJI.zip)16 KB

|  |  |  |
| --- | --- | --- |
|  |  |  |

Remove these adsRemove these ads by [**Signing Up**](http://www.instructables.com/account/gopro?sourcea=removeads&nxtPgName=Control+any+gadgets+with+your+Brain%21+%28Neurofeedback+with+Arduino+Microcontroller%29&nxtPg=/id/Control-any-gadgets-with-your-Brain-Nero-Feedbac/)

## Step 1: Prepare your Arduino Software!

[[](http://cdn.instructables.com/F2J/VUZ7/GTXKH9I1/F2JVUZ7GTXKH9I1.LARGE.jpg)](http://cdn.instructables.com/F2J/VUZ7/GTXKH9I1/F2JVUZ7GTXKH9I1.LARGE.jpg)

[[](http://cdn.instructables.com/F2H/G48R/GTXK8AQZ/F2HG48RGTXK8AQZ.LARGE.jpg)](http://cdn.instructables.com/F2H/G48R/GTXK8AQZ/F2HG48RGTXK8AQZ.LARGE.jpg)

**First, Make sure your arduino is connected to your computer with USB connector and  
the Arduino GUI is installed.  
I will assume that you already had experiance with uploading code into your arduino. If not, Check**[**This**](http://arduino.cc/en/Guide/HomePage)**guide!  
  
When thats done, Open up the downloaded 'Brain' folder and click'Examples'. Then you will see 2 folders  
For now, We will use the program 'BrainSerialOut'. So open the folder and Double-Click the  
'BrainSerialOut.pde' file and upload it into your Arduino.  
  
Thats all the software you will need!! :D  
  
Helpful Notes: The 'Brain.cpp' and 'Brain.h' is a library for this program. Make sure the'BrainSerialOut.pde' file is in the same folder with 'Brain.cpp' and 'Brain.h' or else it wont work.**

## Step 2: Time for hardware hacking!! - Opening your Headset

[[](http://cdn.instructables.com/F53/BFSY/GTYSUZ77/F53BFSYGTYSUZ77.LARGE.jpg)](http://cdn.instructables.com/F53/BFSY/GTYSUZ77/F53BFSYGTYSUZ77.LARGE.jpg)

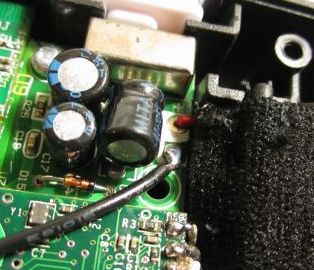
**OK! Time for Hardware Hacking!!!! >:D**

**To hack our victim (aka Mindflex Headset), We need to open it. Well, Go get your screwdriver and start opening**

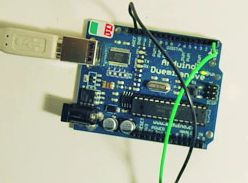
## Fase 3: saldare alcuni fili

[[](http://cdn.instructables.com/FDO/VIZQ/GTYSUZ7O/FDOVIZQGTYSUZ7O.LARGE.jpg)](http://cdn.instructables.com/FDO/VIZQ/GTYSUZ7O/FDOVIZQGTYSUZ7O.LARGE.jpg)

[[](http://cdn.instructables.com/FZ3/YWNM/GTZK8EAH/FZ3YWNMGTZK8EAH.LARGE.jpg)](http://cdn.instructables.com/FZ3/YWNM/GTZK8EAH/FZ3YWNMGTZK8EAH.LARGE.jpg)

[[](http://cdn.instructables.com/FH0/I9LY/GTXKH9JQ/FH0I9LYGTXKH9JQ.LARGE.jpg)](http://cdn.instructables.com/FH0/I9LY/GTXKH9JQ/FH0I9LYGTXKH9JQ.LARGE.jpg)

**Ora il nostro auricolare è aperto, Controllare il circuitboard e trovare l'etichetta denominata 'T'.   
La 'T' sta per trasmissione. Qual è il perno che ci invia le informazioni.   
Ci sarà un po 'glob saldatura sotto l'etichetta di' T '. Saldare un filo al glob ( guardare in alto per le foto !) Una volta che il fatto, saldare un altro filo al glob saldatura terra ( Vedi foto per i dettagli ). Avrete totale 2 fili totale saldato! Andiamo avanti!**

[[](http://cdn.instructables.com/FJN/0DIC/GTXK8AXO/FJN0DICGTXK8AXO.LARGE.jpg)](http://cdn.instructables.com/FJN/0DIC/GTXK8AXO/FJN0DICGTXK8AXO.LARGE.jpg)

**Per arrivare acess al nostro auricolare qui, quello che dovrete fare è quello di collegare i fili che arrivano   
dalla cuffia al *corretto* perno della nostra Arduino. Allora cosa pin?Collegare il cavo che abbiamo collegato al blob 'T' in precedenza al pin RX Arduino, e per l'altro filo, collegarlo al Arduino terra (lo dirò di nuovo, controllare le immagini di cui sopra) Punte: Il perno di Arduino 'RX' sta per il ricevitore, che riceve i dati dalla cuffia. Questo è tutto! : D**

[[](http://cdn.instructables.com/FTA/05TO/GTZK8E4F/FTA05TOGTZK8E4F.LARGE.jpg)](http://cdn.instructables.com/FTA/05TO/GTZK8E4F/FTA05TOGTZK8E4F.LARGE.jpg)

[[](http://cdn.instructables.com/FF0/9ULF/GTXKH9EJ/FF09ULFGTXKH9EJ.LARGE.jpg)](http://cdn.instructables.com/FF0/9ULF/GTXKH9EJ/FF09ULFGTXKH9EJ.LARGE.jpg)

[MindFlex HeadSet](http://cdn.instructables.com/FF0/9ULF/GTXKH9EJ/FF09ULFGTXKH9EJ.LARGE.jpg)

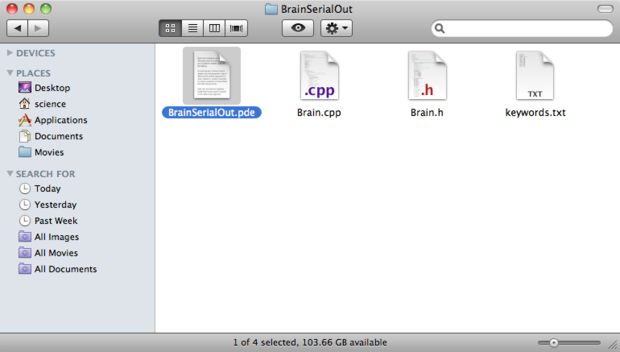
**You've Seen many cool gadgets on news like, controlling robot.  
I've been pretty shaken up with that one. So after some 6 months of research, I made my own!  
All the parts cost me less then 100$! Thats cool isn't it?**  
  
**Materials:  
  
- 1 MindFlex - 50$ <= Got mine from**[**EBay**](http://www.ebay.com/sch/i.html?rt=nc&LH_BIN=1&_nkw=mindflex&_trksid=p3286.c0.m301)**!  
- 1 Arduino Microcontroller <= Visit the official Aruino Site**[**Here**](http://arduino.cc/)**!  
- Some Wires <= Duh! I got it from my old mashed up computer...  
- A Personal Computer <= Everyone has it. Right?  
- Some time  
  
Software & Files:  
  
- Arduino GUI <= Download Right**[**Here**](http://arduino.cc/en/Main/Software)**!  
- Arduino Brain Example Code <= Download at the bottom!  
  
Tools:  
  
- Soldering Iron <= You'll need it.  
- Hot Glue gun <= Not REALLY needed, but we have to secure wires right?  
- Drill <= Not really needed, its used to make a hole so wire can go through the case  
- Other tools (Pliers, Wire Cutters, Wire Strippers, etc.) <= This will help ya!  
  
Well, Thats about it! Lets move on!**

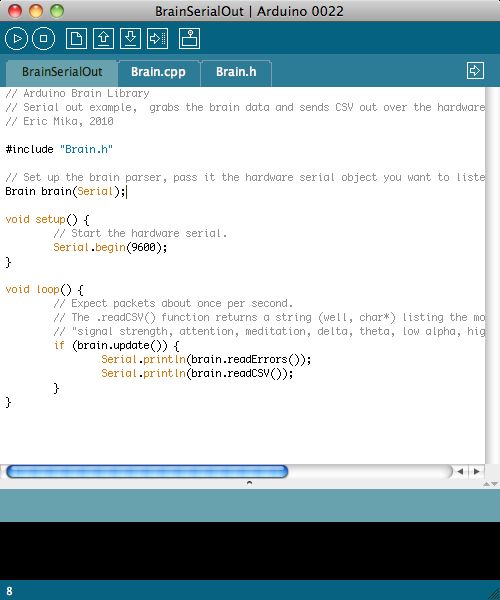
[[http://www.instructables.com/static/defaultIMG/file/ZIP.gif](http://www.instructables.com/files/orig/FFC/JM9C/GTZK8EJI/FFCJM9CGTZK8EJI.zip)**Brain.zip**](http://www.instructables.com/files/orig/FFC/JM9C/GTZK8EJI/FFCJM9CGTZK8EJI.zip)16 KB

|  |  |  |
| --- | --- | --- |
|  |  |  |

Remove these adsRemove these ads by [**Signing Up**](http://www.instructables.com/account/gopro?sourcea=removeads&nxtPgName=Control+any+gadgets+with+your+Brain%21+%28Neurofeedback+with+Arduino+Microcontroller%29&nxtPg=/id/Control-any-gadgets-with-your-Brain-Nero-Feedbac/?ALLSTEPS)

## Step 1: Prepare your Arduino Software!

[[](http://cdn.instructables.com/F2J/VUZ7/GTXKH9I1/F2JVUZ7GTXKH9I1.LARGE.jpg)](http://cdn.instructables.com/F2J/VUZ7/GTXKH9I1/F2JVUZ7GTXKH9I1.LARGE.jpg)

[[](http://cdn.instructables.com/F2H/G48R/GTXK8AQZ/F2HG48RGTXK8AQZ.LARGE.jpg)](http://cdn.instructables.com/F2H/G48R/GTXK8AQZ/F2HG48RGTXK8AQZ.LARGE.jpg)

**First, Make sure your arduino is connected to your computer with USB connector and  
the Arduino GUI is installed.  
I will assume that you already had experiance with uploading code into your arduino. If not, Check**[**This**](http://arduino.cc/en/Guide/HomePage)**guide!  
  
When thats done, Open up the downloaded 'Brain' folder and click'Examples'. Then you will see 2 folders  
For now, We will use the program 'BrainSerialOut'. So open the folder and Double-Click the  
'BrainSerialOut.pde' file and upload it into your Arduino.  
  
Thats all the software you will need!! :D  
  
Helpful Notes: The 'Brain.cpp' and 'Brain.h' is a library for this program. Make sure the'BrainSerialOut.pde' file is in the same folder with 'Brain.cpp' and 'Brain.h' or else it wont work.**

## Step 2: Time for hardware hacking!! - Opening your Headset

[[](http://cdn.instructables.com/F53/BFSY/GTYSUZ77/F53BFSYGTYSUZ77.LARGE.jpg)](http://cdn.instructables.com/F53/BFSY/GTYSUZ77/F53BFSYGTYSUZ77.LARGE.jpg)

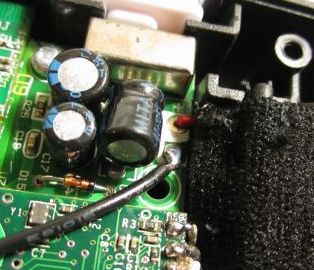
**OK! Time for Hardware Hacking!!!! >:D**

**To hack our victim (aka Mindflex Headset), We need to open it. Well, Go get your screwdriver and start opening!**

## Step 3: Soldering some wires

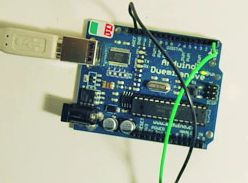
[[](http://cdn.instructables.com/FDO/VIZQ/GTYSUZ7O/FDOVIZQGTYSUZ7O.LARGE.jpg)](http://cdn.instructables.com/FDO/VIZQ/GTYSUZ7O/FDOVIZQGTYSUZ7O.LARGE.jpg)

[[](http://cdn.instructables.com/FZ3/YWNM/GTZK8EAH/FZ3YWNMGTZK8EAH.LARGE.jpg)](http://cdn.instructables.com/FZ3/YWNM/GTZK8EAH/FZ3YWNMGTZK8EAH.LARGE.jpg)

[[](http://cdn.instructables.com/FH0/I9LY/GTXKH9JQ/FH0I9LYGTXKH9JQ.LARGE.jpg)](http://cdn.instructables.com/FH0/I9LY/GTXKH9JQ/FH0I9LYGTXKH9JQ.LARGE.jpg)

**Now our headset is opened, Examine your circuitboard and find the label named ' T '.  
The ' T ' stands for Transmit. Which is the pin that sends us information.  
There will be a little solder glob below the label ' T '. Solder a wire to the glob (look up for pictures!)  
  
Once your done, solder a another wire to the ground solder glob(See picture for details).  
  
You will have total 2 wires soldered total! Lets move on!**

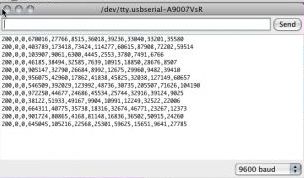
## Step 4: Connecting wires to the Arduino

[[](http://cdn.instructables.com/FJN/0DIC/GTXK8AXO/FJN0DICGTXK8AXO.LARGE.jpg)](http://cdn.instructables.com/FJN/0DIC/GTXK8AXO/FJN0DICGTXK8AXO.LARGE.jpg)

**To get acess to our Headset here, what you'll need to do is to connect the wires that are coming  
from the headset to the correct pin of our Arduino.  
  
So what pin? Connect the wire that we connected to the ' T ' blob earlier to the Arduino pin RX,  
and for the other wire, connect it to the Arduino Ground (I'll say it again, check out the pictures above)  
  
Tips: The Arduino pin 'RX' stands for Receiver, which receives data from the headset.  
  
Thats all! :D**

## Step 5: Test em' out!

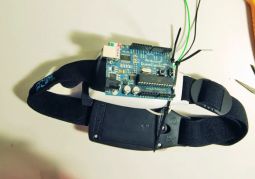
[[](http://cdn.instructables.com/F45/G09U/GTXK8AYF/F45G09UGTXK8AYF.LARGE.jpg)](http://cdn.instructables.com/F45/G09U/GTXK8AYF/F45G09UGTXK8AYF.LARGE.jpg)

[[](http://cdn.instructables.com/FHZ/3UTT/GTXKH9LQ/FHZ3UTTGTXKH9LQ.LARGE.jpg)](http://cdn.instructables.com/FHZ/3UTT/GTXKH9LQ/FHZ3UTTGTXKH9LQ.LARGE.jpg)

**So lets test this out!  
First, connect the arduino to your computer with your USB cable. Then open up the Arduino GUI  
and click the Serial Monitor (The logo thats has box with a lollipop sticking out on the top).**

**Attach your headset (Make sure you have batteries) and you will see bunch of data spitting out.  
If it have numbers somthing like this:  
0, 634, 5344, 5334, 78934, 6544, 63434, 9834, 3574, 30564, 67534  
  
Then you have a good connection!**

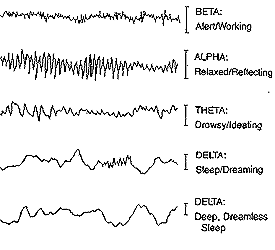
## Step 6: Finishing it up.

[[](http://cdn.instructables.com/FCU/LZH1/GTXKH9MK/FCULZH1GTXKH9MK.LARGE.jpg)](http://cdn.instructables.com/FCU/LZH1/GTXKH9MK/FCULZH1GTXKH9MK.LARGE.jpg)

**Once we checked its working, Lets now finish this gadget up!**

**I drilled a small hole in the case so the wires can go through.  
Also, I used some hot glue gun to secure wires and attatch the Arduino to the headset.  
  
The final gadget with some finishing touches are shown above.**

## Step 7: Understanding the data!

[[](http://cdn.instructables.com/FI2/TFDP/GTXK8B1H/FI2TFDPGTXK8B1H.LARGE.gif)](http://cdn.instructables.com/FI2/TFDP/GTXK8B1H/FI2TFDPGTXK8B1H.LARGE.gif)

**Now lets examine the data this thing is spitting out!  
The long number this spits out can be breaked down into this:  
  
=> signal strength, attention, meditation, delta, theta, low alpha, high alpha, low beta, high beta, low gamma, high gamma  
  
If you dont know what is gamma or alpha or you think this is toooo hard, check out the  
wiki about brain signals**[**here**](http://en.wikipedia.org/wiki/Electroencephalography#Comparison_table)**! (Also check the picture above)  
  
Now lets make our Real-Life-Application(Well, sort of)!**

## Step 8: Our First Real-Life-Application (Sort of)

[[](http://cdn.instructables.com/F7J/I4QM/GTYSUZG7/F7JI4QMGTYSUZG7.LARGE.jpg)](http://cdn.instructables.com/F7J/I4QM/GTYSUZG7/F7JI4QMGTYSUZG7.LARGE.jpg)

**Now lets get in more deeply! How about creating a 'Attention Analyzer'??  
  
Open up the 'Brain' folder, navigate to 'Examples' folder. And this time, goto the 'BrainTest' folder and click the 'BrainTest.pde'.  
Now upload it to your arduino!  
  
So what does it do? Well, If the person's (who is wearing the headset ) attention goes up, the blinking of the LED goes faster.  
(Mentioning about the LEDs, The LED that blinks is built-in to the Arduino so you dont have to worry about adding a LED.)  
The picture shoes Arduino's built in LED that will blink in this experiment.**

## Step 9: So what now?

**So what now?**  
  
The title says that you can control anything with this gadget. Well, controlling a gadget can be break down into these steps:  
  
Determine what brain data you will need => Examine the data by putting the headset on and thinking it => Get the data range  
=> Write a Code => Make a circuit that turns the circuit ON/OFF with transisters => Connect the gdaget  
  
I suggest that these steps can be performed by who has atleast 5 years of electronic experiance.  
Brain data management to control any gadget is not easy as you think.  
  
**But for all the remaining people out there, There is some cool example programs all around the internet!  
Like this cool Analyzer Program (Download Below) <- You may need to install the controlP5 library (Search Google)**

**Thanks for taking your time reading this instructables! God Bess!**

[[http://www.instructables.com/static/defaultIMG/file/ZIP.gif](http://www.instructables.com/files/orig/FAV/I57J/GTYSUZNL/FAVI57JGTYSUZNL.zip)**KitschPatrol BrainGrapher.zip**](http://www.instructables.com/files/orig/FAV/I57J/GTYSUZNL/FAVI57JGTYSUZNL.zip)897 KB

<https://www.youtube.com/watch?v=7CMez9dXAaE>

**Pubblicato il 10 set 2012**

Tesi di Luca Vassallo per la maturità 2012, classe 5M Itis Castelli: realizzazione di una mano meccanica, azionata da servo motori pilotati da Arduino, con input attraverso un guanto dotato di sensori di flessione delle dita. Se qualcuno fosse interessato ai dettagli tecnici può scrivere direttamente a Luca a questo indirizzo: lucavassallo1 at virgilio.it

<https://www.youtube.com/watch?v=t52edTD9RA0>