



Arduino MaxMSP Tutorial #1 - MaxMSP invia dati ad Arduino.

Materiale utilizzato – Arduino Uno, Grove Shield, 1 servo motore
Per vedere tutti i tutorial: www.marcopucci.it/category/maxmsp-arduino/

Questo tutorial è stato realizzato in collaborazione con www.robotics-3d.com.

Qui sotto i link per acquistare i materiali utilizzati in questo tutorial

Arduino Uno - [clicca qui](#)
Grove shield - [clicca qui](#)
Servo motore - [clicca qui](#)

Clicca qui per scaricare tutti gli sketch utilizzati nel tutorial Arduino MaxMSP

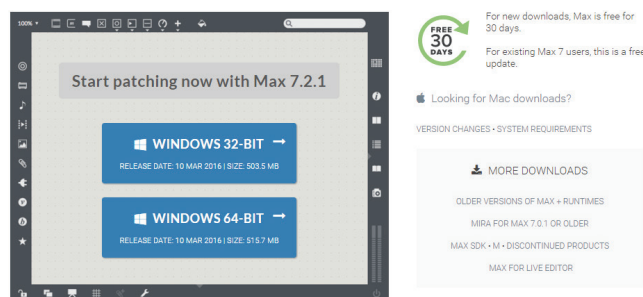
MaxMSP è un software utilizzato da compositori e artisti per realizzare installazioni sonore e multimediali. La programmazione è molto semplice perchè modulare, è quindi possibile grazie a dei blocchi realizzare i propri sketch in maniera facile e veloce.

In questi tutorial MaxMSP verrà collegato fin da subito ad Arduino. In questo modo saremo in grado di costruire installazione sonore e video interattivi che vengono modificati in base alla lettura dei sensori. Arduino in questo caso viene utilizzato solamente per trasmettere dati a Max che a sua volta li analizzerà. Se non sapete utilizzare di Arduino vi invito a leggere i primi 4 tutorial presenti sul mio sito a [questo link](#).

MaxMSP non è un software libero. E' possibile scaricare la versione gratuita che dura 30 giorni e poi acquistare diverse tipologie di licenze (in fondo alla pagina alcune differenti licenze aggiornate a marzo 2016):

Collegiamoci al sito <https://cycling74.com/downloads/> per effettuare il download.

Ps: se avete Windows a 64 bit vi consiglio di installare comunque la versione a 32 bit. E' più stabile.



Una volta installato MaxMSP è possibile iniziare a realizzare le proprie patch però non è possibile salvare. Cliccate su Save e apparirà una scritta che ci avvisa che dobbiamo registrarci. Procediamo con la registrazione (nome, cognome, email, ecc...). Verrà inviata un' email con un codice per attivare la licenza di prova. Una volta inserito il codice la licenza gratuita di 30 giorni è attivata.

Se volete acquistare MaxMSP qui sotto trovate le diverse tipologie di licenze: <https://cycling74.com/shop/>

COMPARE YOUR OPTIONS	Monthly Subscription	Annual Subscription	Permanent License	BUY MAX 7 →
Payment options	\$9.99/mo PAID MONTHLY	\$8.25/mo PAID ANNUALLY	\$399 PAID ONCE	
Educational discount available, for individual students and teachers PROOF OF ID REQUIRED UPON PURCHASE	✗	✓	✓	<ul style="list-style-type: none"> Monthly and annual subscriptions are set to recurring billing. You must cancel manually. Upgrades from previous versions of Max 5 and 6 are also available.
Cancel and re-activate anytime	✓	✓	YOURS FOREVER	



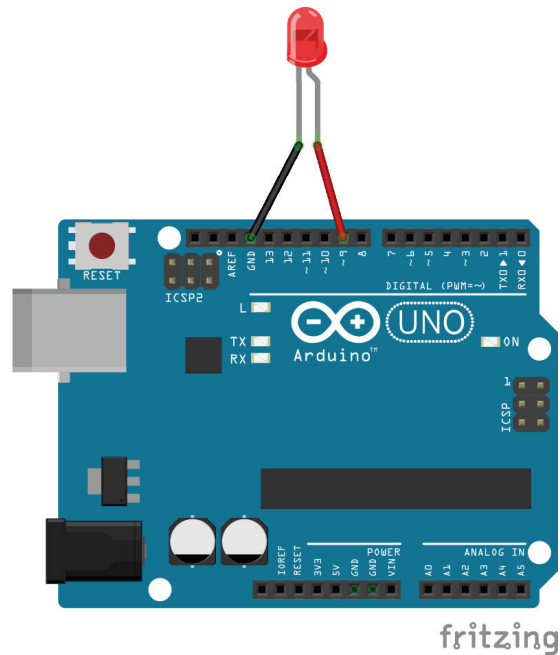
E' possibile acquistare la licenza mese per mese al prezzo di 9\$ oppure quella annuale. Inoltre se si è studenti o docenti ci sono ulteriori sconti a questo link <https://cycling74.com/special-pricing/>

Arduino

MaxMsp e Arduino si scambiano dati tramite la porta USB.

Lo sketch di Arduino è semplicissimo. Dobbiamo solamente leggere o scrivere (Serial.read o print) come abbiamo fatto in tutti i tutorial di Arduino per accendere led o stampare sul Serial Monitor i dati di un sensore. In questo primo tutorial spiego come inviare dati da MaxMsp ad Arduino.

Collegiamo un led al pin 9 di Arduino. Accenderemo questo led tramite i valori inviati da MaxMsp.



Apriamo il file **MaxMsp_invia_dati_ad_Arduino.ino**

```
int ledPin = 9;
void setup()
{
  Serial.begin(9600);
  pinMode(ledPin, OUTPUT);
}

void loop() {
  int luminosita;
  if (Serial.available()) {
    luminosita = Serial.read();

    analogWrite(ledPin, luminosita);
  }
}
```

Utilizzo il pin 9 per accendere e spegnere il led

Se è presente un dato nella porta seriale (inviato da Max) allora questo è assegnato alla variabile luminosità

Scriviamo sul pin 9 il valore che arriva dalla porta seriale. In questo modo il led si accenderà gradualmente.

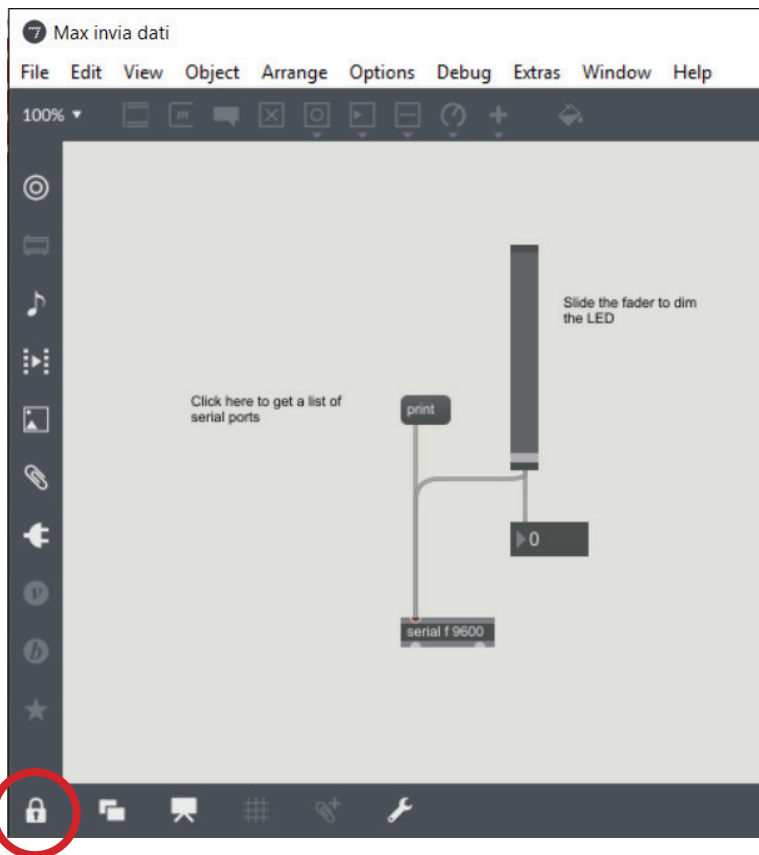
N.B: Vi ricordo che il comando analogWrite è possibile effettuarlo solamente nei pin 3, 5, 6, 9, 10 e 11 di Arduino (pin PWM)



Uplodiamo il codice sulla scheda.
Arduino è dunque pronto a ricevere dati da MaxMsp.

MaxMsp

Apriamo il file **Max invia dati.maxpat**



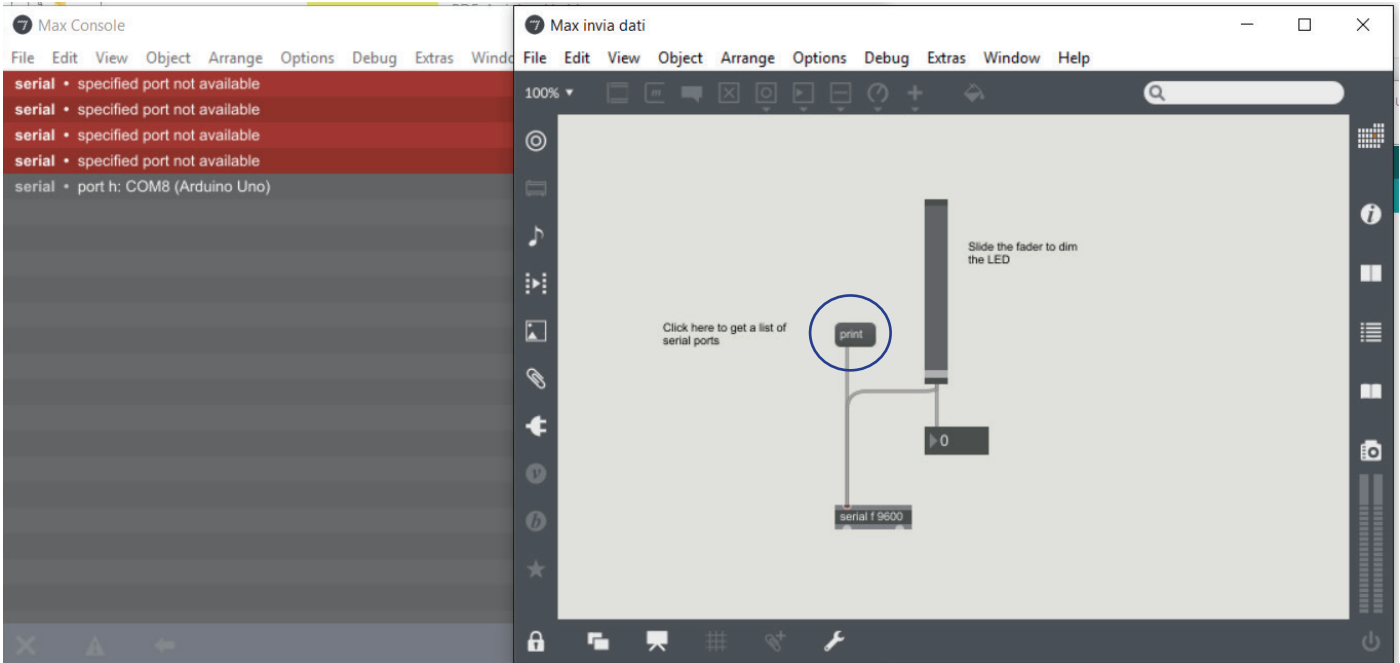
Imparerete ad utilizzare MaxMsp nel corso di questi tutorial in maniera semplice modificando patch già realizzate. Iniziamo a capire la logica di programmazione che questa volta avviene inserendo dei blocchi all'interno dell'area di lavoro.

Max funziona in due modalità: esecuzione e programmazione. Per passare da una modalità all'altra dobbiamo cliccare sull'icona in basso a sinistra a forma di lucchetto.

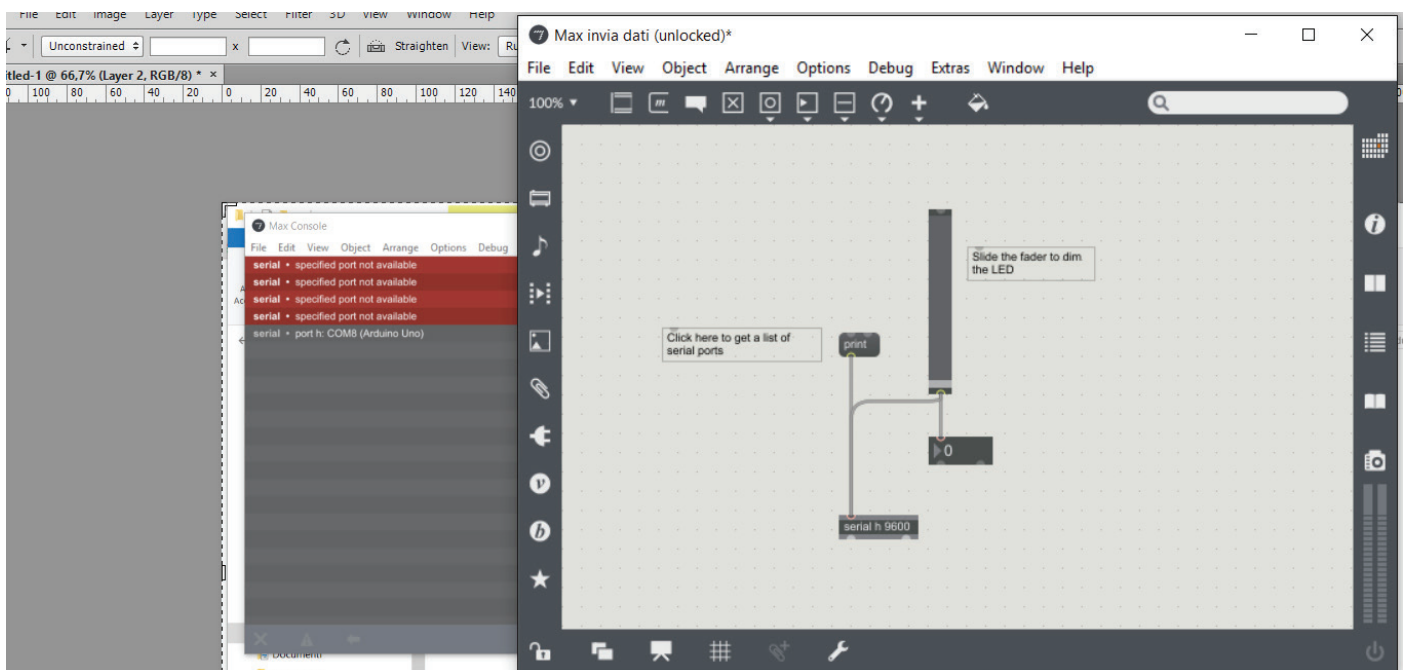
Collegiamo subito Max ad Arduino. Click sul lucchetto (da ora in poi indicherò questo comando con Bloccato e Sbloccato) per rendere la patch in modalità eseguibile.

Click su Windows / MaxConsole. Si apre una finestra di dialogo che ci serve per vedere su quale porta USB sta comunicando Arduino.

Seguite le istruzioni come nella figura qui sotto.

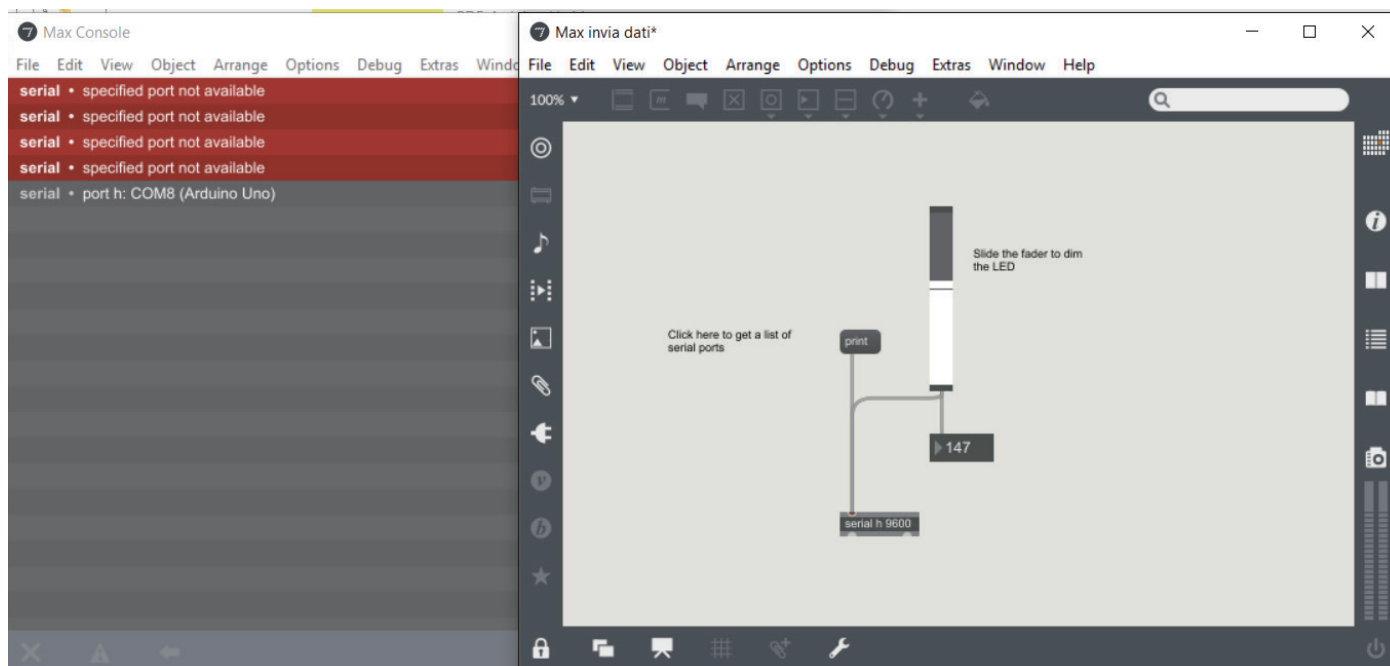


Click su print (evidenziato con il cerchio blu). Nella schermata di MaxConsole (a sinistra) appare una scritta: serial port **h**: COM8 (Arduino Uno). La lettera "h", che nel vostro caso sarà differente, è il nome della porta USB a cui è collegato Arduino (se non appare niente provate a cambiare porta USB di Arduino). Sblocciamo la patch e clicchiamo all'interno del blocco serial f 9600. Dobbiamo solamente modificare la lettera "f" con quella apparsa nel Max Console (nel mio caso diventa serial h 9600).





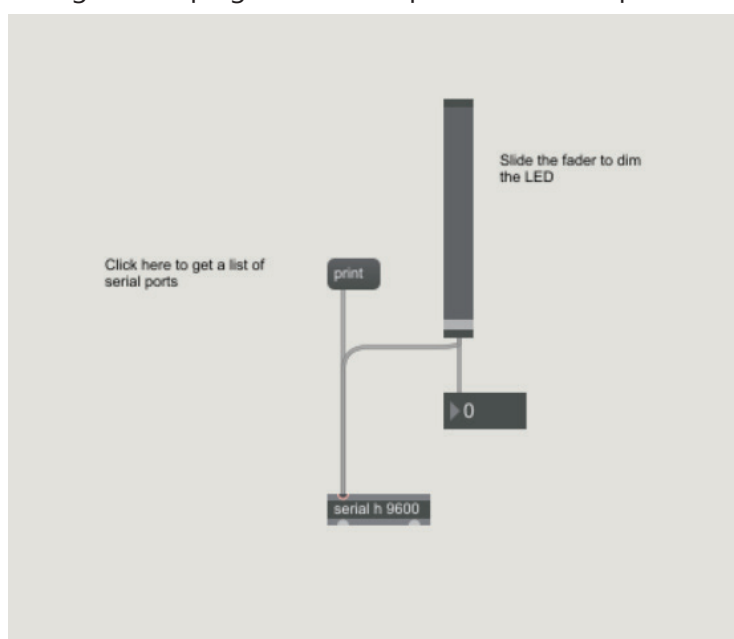
Blocciamo la patch e muoviamo con il mouse lo slider (come nella figura qui sotto).



Se tutto è andato bene muovendo lo slider in su e in giù il led si accenderà.

ATTENZIONE: se Arduino e MaxMsp non dovessero comunicare provate a chiudere il software di Arduino. Chiudete MaxMsp, riavviate e provate di nuovo.

Di seguito la spiegazione della patch di MAXMSP.



Lo slider è un componente che genera un numero quando viene mosso con il mouse.

Lo slider è collegato sotto con un secondo blocco (Number) che visualizza semplicemente il numero generato dallo slider.

Questo numero viene inviato ad Arduino tramite il blocco "serial f 9600" e contemporaneamente viene visualizzato nel Number.

Dai prossimi tutorial realizzeremo diverse patch per gestire file audio e video in maniera interattiva.

Per concludere questo primo tutorial vediamo come muovere un servo motore attraverso MaxMsp.

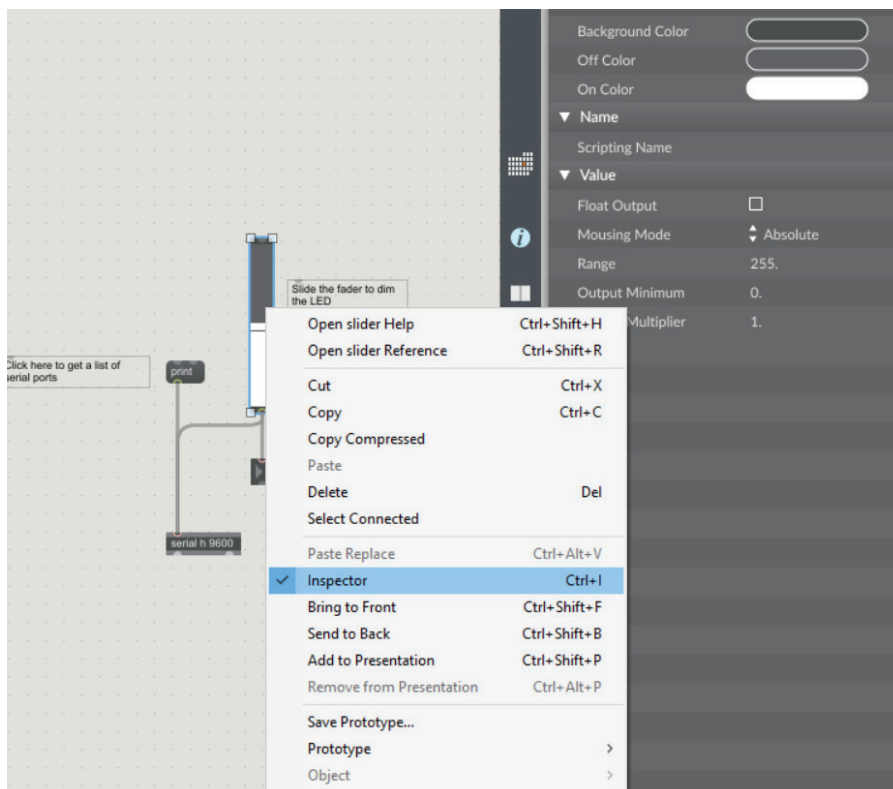


Gestione di un servo motore tramite MaxMsp.

Nell'esempio precedente abbiamo acceso un led spostando uno slider da MaxMsp. Questo slider genera un valore che va da 0 a 255. Il motorino servo invece ha bisogno di un valore che va da 0 a 180°.

Dobbiamo semplicemente modificare le proprietà dello slider.

In modalità Sbloccata clicchiamo con il tasto destro del mouse sullo slider e selezioniamo **Inspector**.

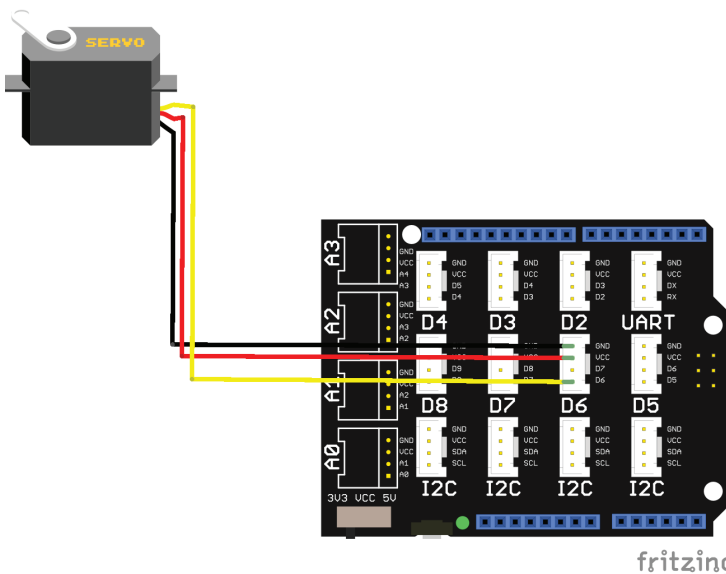


A destra si apre una finestra dove appare la scritta Range 255 e Output Minimum 0.

Lasciamo il valore 0 ma modifichiamo Range con **180**.

Blochiamo la patch e muoviamo lo Slider. Il numero sotto varierà tra 0 e 180 e verrà inviato ad Arduino che muoverà il servo motore.

Arduino



Collegiamo un servo motore 180° ad Arduino. Qui a fianco il semplice collegamento utilizzando Arduino Uno, una shield Grove e un servo motore Grove (se non sai cosa sono i componenti Grove leggi il precedente tutorial di Arduino, [clicca qui](#)).

Potete comunque realizzare il circuito con qualsiasi scheda Arduino e motorini servo.

Se non avete mai utilizzato i servo motori vi invito a leggere questo tutorial di Arduino, [cliccate qui](#).



Apriamo il file **Max_muove_motorino.ino**

```
#include <Servo.h>
Servo myservo;
int pos = 0;
```

```
void setup()
```

```
{
  Serial.begin(9600);
  myservo.attach(6);           // Collego il motorino al pin 6
}
```

```
}
```

```
void loop() {
```

```
  if (Serial.available()) {
    pos = Serial.read();
    myservo.write(pos);
    delay(50);
  }
}
```

Se è presente un dato nella porta seriale (inviato da Max) allora questo è assegnato alla variabile pos
Con questo comando sposto il motorino in base al valore proveniente da MaxMsp

Prima di Uplodare questo codice dobbiamo chiudere MaxMsp altrimenti i due programmi vanno in conflitto perchè stanno utilizzando la stessa porta USB.

Chiudiamo Max.

Uplodiamo il codice di Arduino.

Apriamo Max.

Impostiamo nuovamente la porta USB (click su print e cambiare la lettera nel blocco "serial h 9600").

Muoviamo lo slider e il motorino inizierà a muoversi.