

Proyecto Final: Brazo robótico

*Seminario: Introducción a la Programación de Microcontroladores con
Tecnologías Libres*



**Barreiro Gustavo, Pintos Nicolas Hernan, Soñez
Mailin**

19/12/2019

ÍNDICE

INTRODUCCIÓN	1
MATERIALES	2
HERRAMIENTAS	3
PROCEDIMIENTO Y FOTOS DEL ARMADO	4
Armado de pinza.	4
Armado del brazo	5
Ensamble de la pinza con el brazo	5
Armado de la base con el motor	6
Colocación de motores (Altura y profundidad)	6
Colocación del motor de la pinza	7
CÓDIGO ARDUINO (v. 1.8.9)	8
CÓDIGO PROCESSING (v. 3.5.3)	8
ESQUEMA	10
CONEXIONES	11
VISTA DE INTERFAZ PROCESSING	11
PROBLEMÁTICAS QUE SURGIERON	12
REFERENCIAS	12

INTRODUCCIÓN

El siguiente proyecto tiene como objetivo, armar, ensamblar y configurar un brazo robótico, hecho con impresora 3D y Servos Tower Pro SG90.

El producto final tiene la capacidad de moverse en todas las direcciones (arriba y abajo, adelante y atrás y giro sobre su eje) y agarrar objetos con la pinza mediante la utilización de un programa compilado y ejecutado con Processing.

MATERIALES

NOMBRE	CANTIDAD	CÓMO SE OBTUVO / COSTO
Piezas impresas 3d + Servos Tower Pro SG90	14 piezas, 4 servos	Compra a conocido / \$1200
Tornillos m4 x 16	5	Incluido en compra
Tornillo m4 x 25	1	Incluido en compra
Tornillo M3 paso 05	2	Incluido en compra
Precintos	-	Incluido en compra
Placa Arduino Uno	1	Prestado por la materia
Cables	-	Prestado por la materia

HERRAMIENTAS

NOMBRE	CANTIDAD	CÓMO SE OBTUVO / COSTO
Alicate	1	Prestado
Destornillador Phillips	3	Prestado
Llave tubo	1	Prestado
Pinza	1	Prestado

PROCEDIMIENTO Y FOTOS DEL ARMADO

1. Herramientas a utilizar



2. Partes del brazo robótico



3. PROCEDIMIENTO

- a. Armado de pinza.



b. Armado del brazo



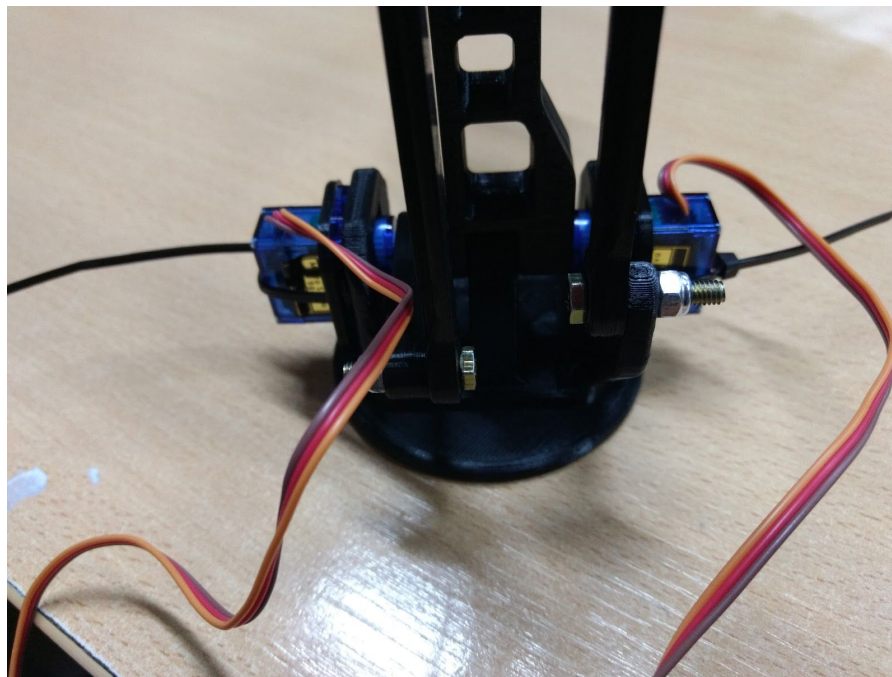
c. Ensamble de la pinza con el brazo



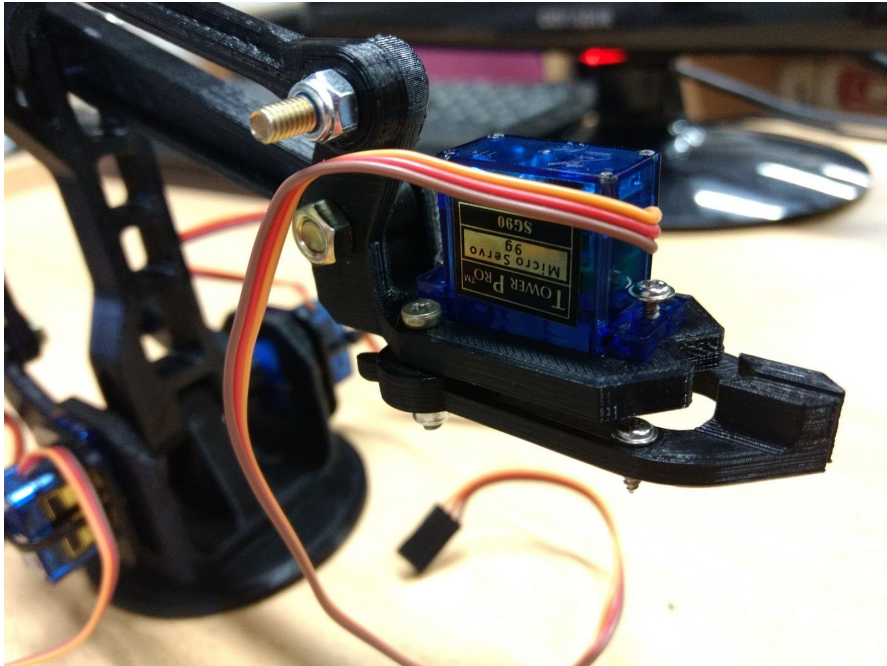
d. Armado de la base con el motor



e. Colocación de motores (Altura y profundidad)



f. Colocación del motor de la pinza



CÓDIGO ARDUINO (v. 1.8.9)

Usamos la librería ServoFirmata que ya viene con el IDE de Arduino.

CÓDIGO PROCESSING (v. 3.5.3)

*Previamente importar biblioteca Arduino(Firmata) y ControlP5

```
import processing.serial.*;
import cc.arduino.*;
import controlP5.*;

Arduino arduino;
ControlP5 cp5;

Slider Servo1;
Slider Servo2;
Slider Servo3;
Slider Servo4;

Button Boton1;
Button Boton2;
Button Boton3;

void setup()
{
  size(950, 900); //Definir el tamaño de la pantalla
  println(Arduino.list()); // Mostrar los puertos COM disponibles
  arduino = new Arduino(this, Arduino.list()[0], 57600); //En los corchetes debes
  introducir el número del COM empezando de 0,1,2,3 dependiendo los dispositivos
  que aparezcan en la lista
  cp5 = new ControlP5(this);

  arduino.pinMode(2, Arduino.SERVO); //Definir el pin 2 para la conexión de un
  servomotor
```

```
arduino.pinMode(3, Arduino.SERVO);//Definir el pin 3 para la conexión de un  
servomotor
```

```
arduino.pinMode(4, Arduino.SERVO);//Definir el pin 4 para la conexión de un  
servomotor
```

```
arduino.pinMode(5, Arduino.SERVO);//Definir el pin 5 para la conexión de un  
servomotor
```

```
Servo1 =  
cp5.addSlider("1").setRange(110,180).setValue(110).setPosition(180,110).setSize(27  
0,25).setNumberOfTickMarks(270).setLabelVisible(false).setColorForeground(colo  
r(255,0,0)).setColorBackground(color(0,255,0)).setColorActive(color(0,0,255));  
Servo2 =  
cp5.addSlider("2").setRange(0,180).setValue(0).setPosition(180,210).setSize(270,25).  
setNumberOfTickMarks(270).setLabelVisible(false).setColorForeground(color(255,  
0,0)).setColorBackground(color(0,255,0)).setColorActive(color(0,0,255));  
Servo3 =  
cp5.addSlider("3").setRange(70,120).setValue(100).setPosition(180,310).setSize(270,  
25).setNumberOfTickMarks(270).setLabelVisible(false).setColorForeground(color(  
255,0,0)).setColorBackground(color(0,255,0)).setColorActive(color(0,0,255));  
Servo4 =  
cp5.addSlider("4").setRange(100,180).setValue(140).setPosition(180,410).setSize(27  
0,25).setNumberOfTickMarks(270).setLabelVisible(false).setColorForeground(colo  
r(255,0,0)).setColorBackground(color(0,255,0)).setColorActive(color(0,0,255));  
}
```

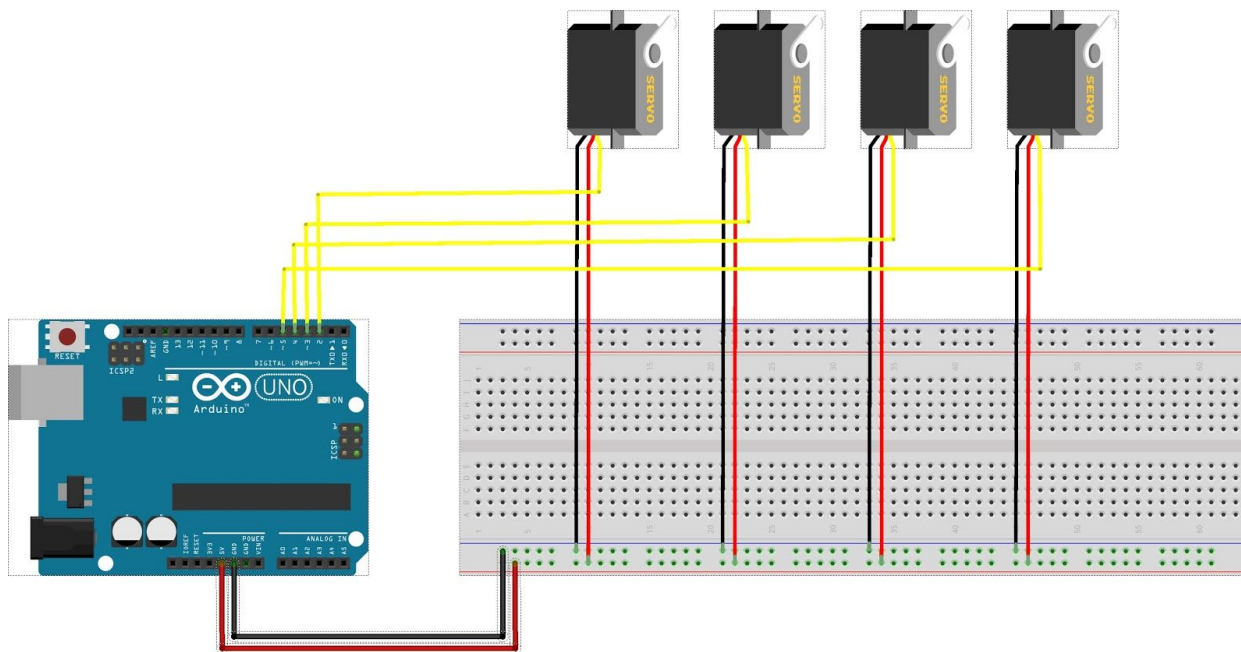
```
void draw()  
{  
background(color(10,10,10));  
arduino.servoWrite(2, int(Servo1.getValue()));  
arduino.servoWrite(3, int(Servo2.getValue()));  
arduino.servoWrite(4, int(Servo3.getValue()));  
arduino.servoWrite(5, int(Servo4.getValue()));  
  
textSize(50);  
text("TP Brazo Robótico", 300, 75);  
  
textSize(25);
```

```
text("Pinza", 20, 125);
text("Giro", 20, 225);
text("Profundidad", 20, 325);
text("Altura", 20, 425);

textSize(18);
text("[Pin 2]", 25, 150);
text("[Pin 3]", 25, 250);
text("[Pin 4]", 25, 350);
text("[Pin 5]", 25, 450);

textSize(25);
text(int(Servo1.getValue()+"°", 390, 130);
text(int(Servo2.getValue()+"°", 390, 230);
text(int(Servo3.getValue()+"°", 390, 330);
text(int(Servo4.getValue()+"°", 390, 430);
}
```

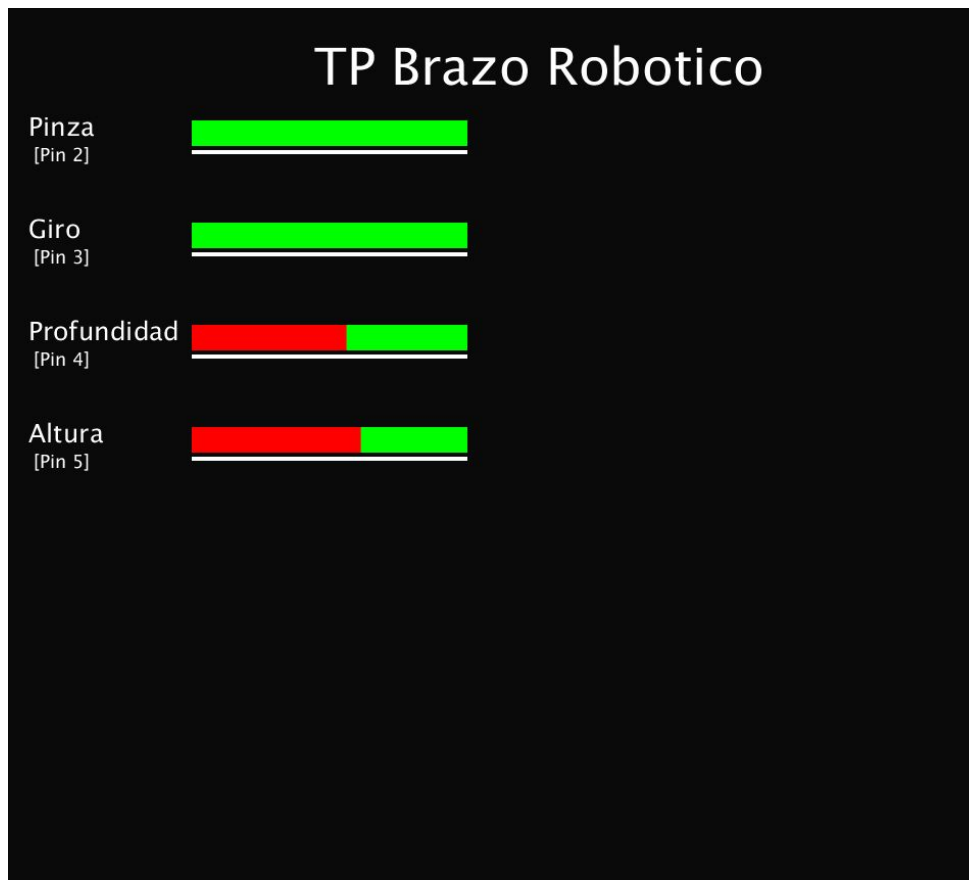
ESQUEMA



CONEXIONES

- Servo1 (pinza) → Pin Digital 2
- Servo2 (giro) → Pin Digital 3
- Servo3 (profundidad) → Pin Digital 4
- Servo4 (altura) → Pin Digital 5

VISTA DE INTERFAZ PROCESSING



PROBLEMÁTICAS QUE SURGIERON

La mayor problemática que tuvimos fue conectar el brazo robótico con la placa arduino, en el primer intento, el día 28 de Noviembre de 2019, luego de finalizar el armado el brazo robótico fuimos probando uno a uno las diferentes partes del brazo para verificar que funcione correctamente, inicialmente funcionaban sin inconvenientes, siempre y cuando se encuentre conectado solo un potenciómetro, por que al conectar los otros potenciómetros el brazo realizaba movimientos inesperados.

El día 05 de Diciembre, al dedicarnos de lleno al problema que habíamos tenido la clase anterior nos encontramos con que ahora no funcionaba ni siquiera con un solo potenciómetro, lo probamos de varias maneras no obteniendo un resultado favorable.

Finalmente pudimos solucionar el inconveniente sin utilizar los potenciómetros, que tantos problemas nos trajeron, y optamos por utilizar una interfaz con Processing para el manejo de cada uno de los motores, modificando las variables de los ángulos de cada servo para que el movimiento sea más fluido. Ésto tuvimos que hacerlo porque si todos los motores se movían de 0 a 180 grados, los movimientos eran muy bruscos y la fuerza de los mismos no era suficiente para volver a sus posiciones.

Por último, detectamos que si queremos utilizar los 4 servomotores concurrentemente, debemos usar un adaptador para la protoboard para poder usar una fuente externa, o dos placas arduino y conectar 2 servos a cada una. De no hacerlo, el voltaje sobrepasa el que brinda el puerto USB de la pc.

REFERENCIAS

1. <https://www.youtube.com/watch?v=r0965C587mE>
2. <https://www.instructables.com/id/EEZYbotARM/>
3. <https://www.taloselectronics.com/blogs/tutoriales/servo-tester-con-arduino-y-processing>