

TALLER DE ESCORNABOT

**Club de Tecnología,
Programación y Robótica
de Granada**

<https://clubroboticagranada.github.io/>



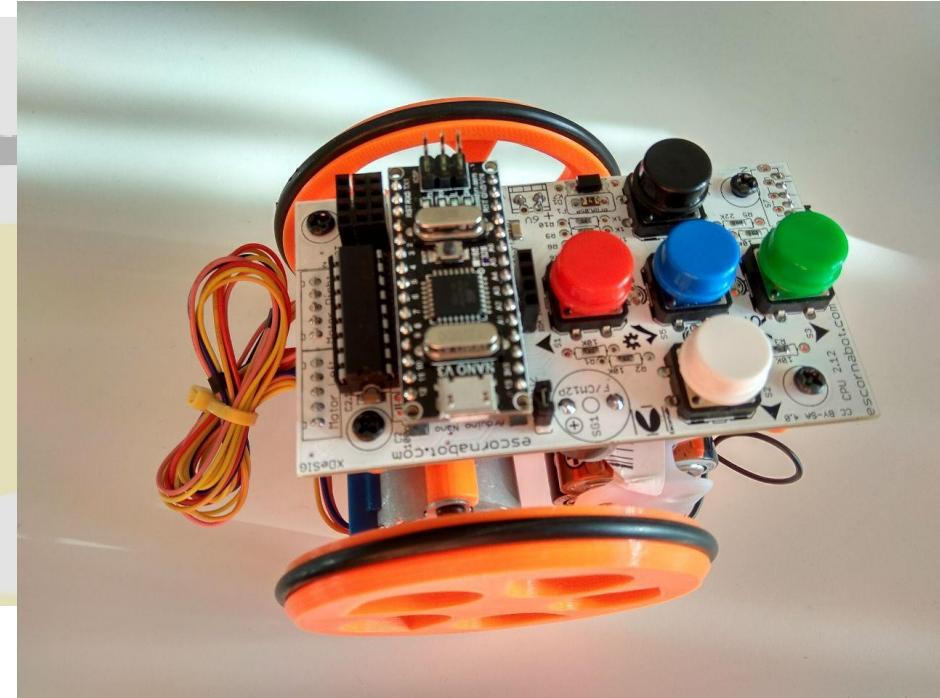
¿Escorna... qué?

ESCORNABOI:
escarabajo
lucanus cervus
en gallego

+

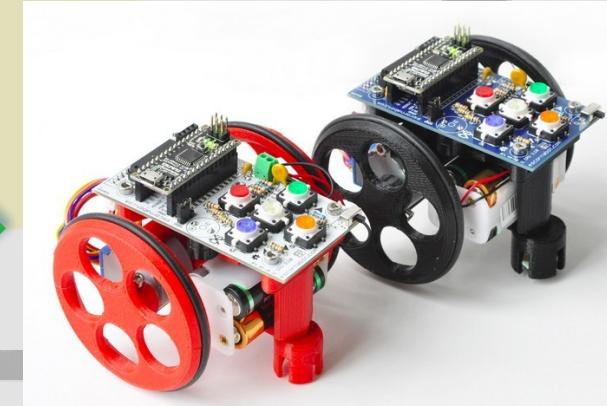
BOT:
robot

martes, 5 de noviembre de 2019



Objetivo

ROBÓTICA Y PROGRAMACIÓN



Sustituye a robots privativos

Características

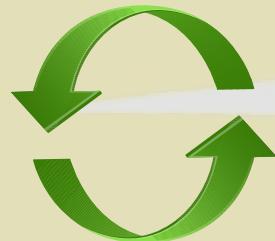


- DIY: Lo haces tú
- Hardware abierto y software libre
- Asequible
- Bien documentado



¿Quién?

Proyecto liderado y soportado por la comunidad:
Profesores, diseñadores, desarrolladores,
traductores...

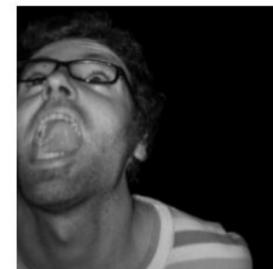


Los tres creadores

Tucho Méndez
[@procastino](#)



Xoán Sampaio
[@xoan](#)



Rafa Couto
[@caligari_pub](#)



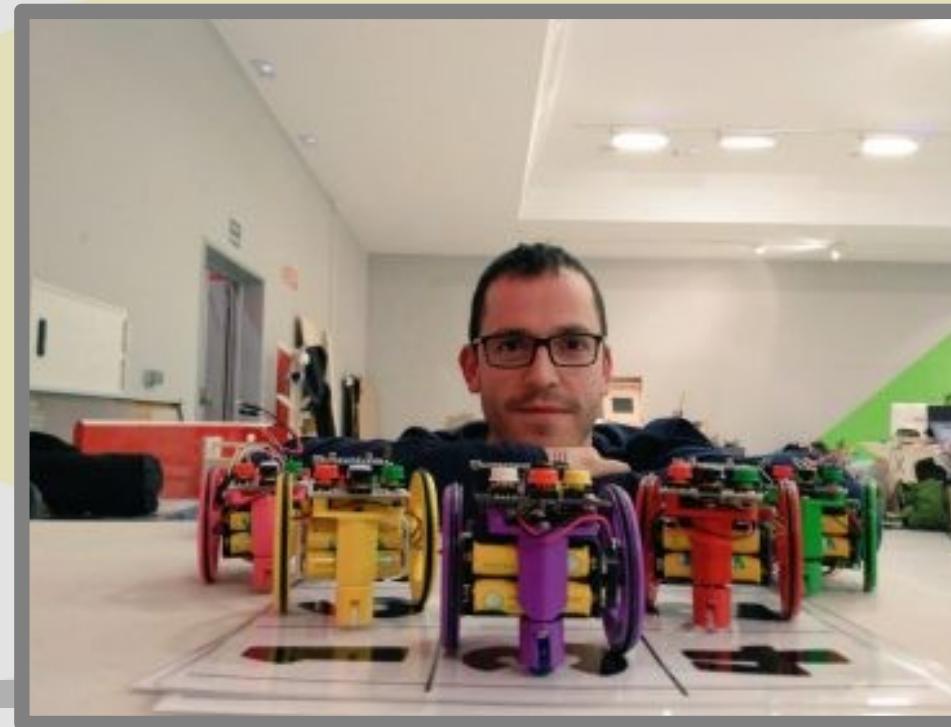
Comunidad: Grupo de Google / Telegram

Github <https://github.com/orgs/escornabot/people/>

iii Escornafan !!!

Pablo Rubio

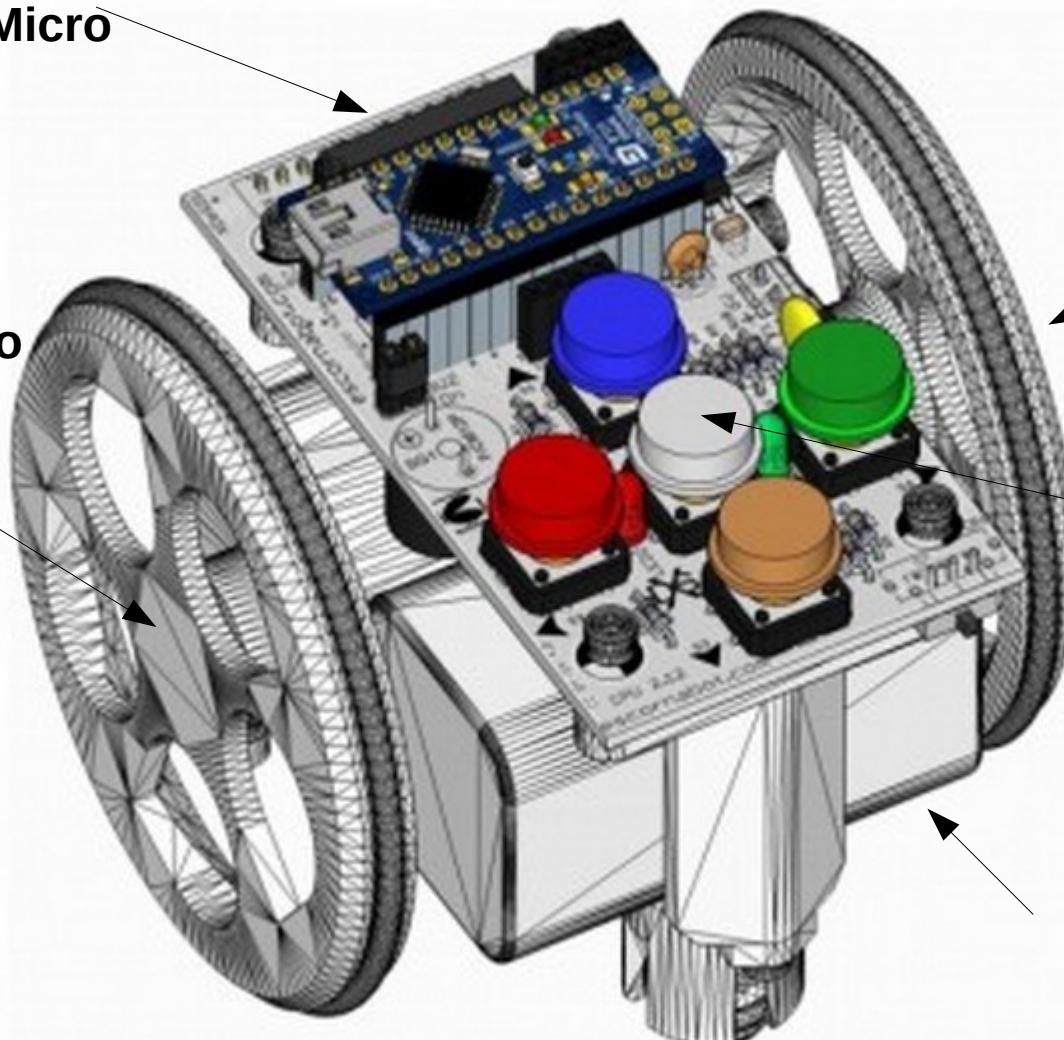
<https://pablorubma.cc/>



Escornabot

Sistema de Control
(Programación)
Arduino Nano o Micro

Actuador 2
Motor Paso a Paso



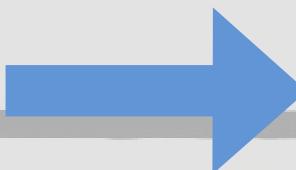
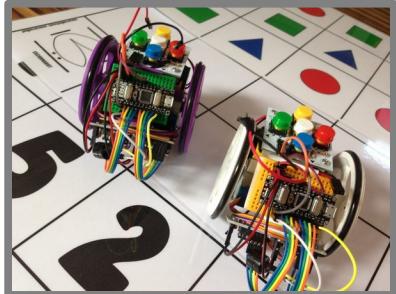
Actuador 1
Motor Paso a Paso

Sensores
5 botones

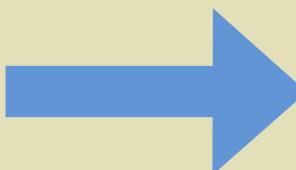
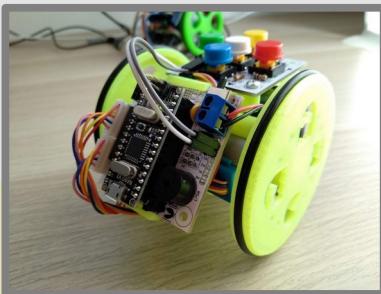
Alimentación
4 pilas



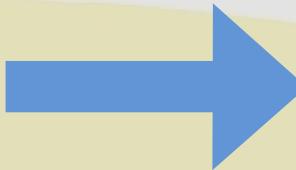
Versiones



DO IT YOURSELF



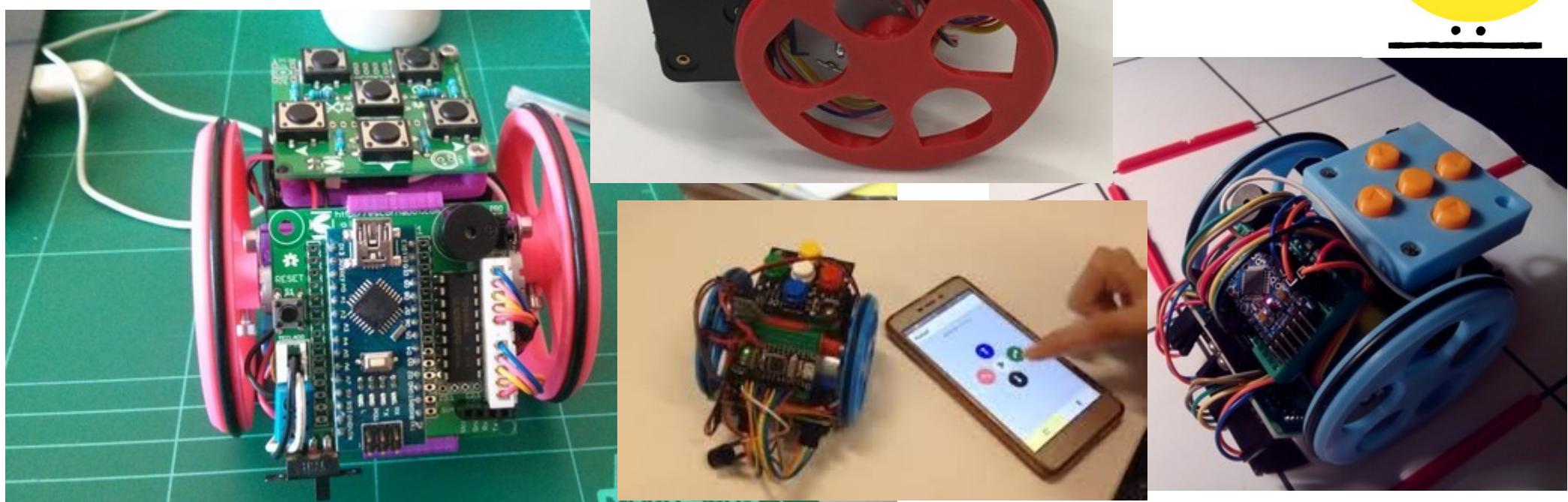
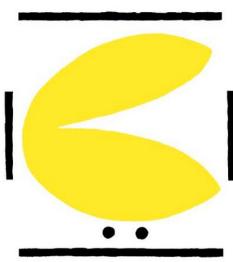
COMPACTUS



PLACA 2.12

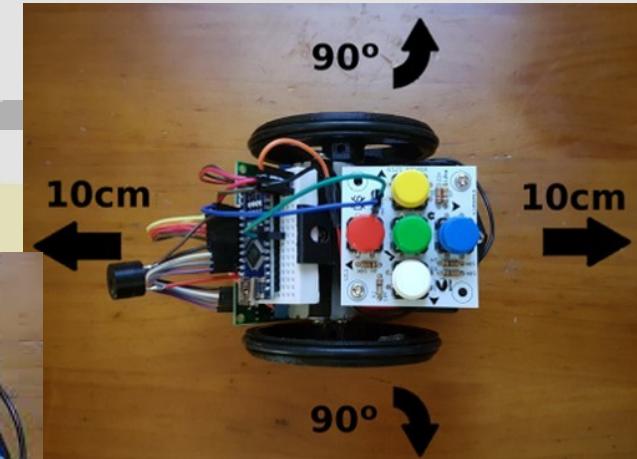
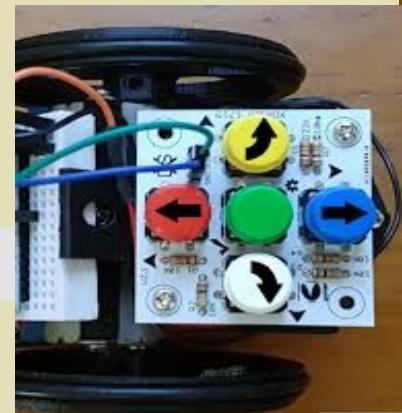


OKAGI



Funcionamiento y programación

- Introduce firmware preparado y se maneja con botonera (modo clásico)



- También se puede programar con librería para Arduino e incluso poner sensores extras

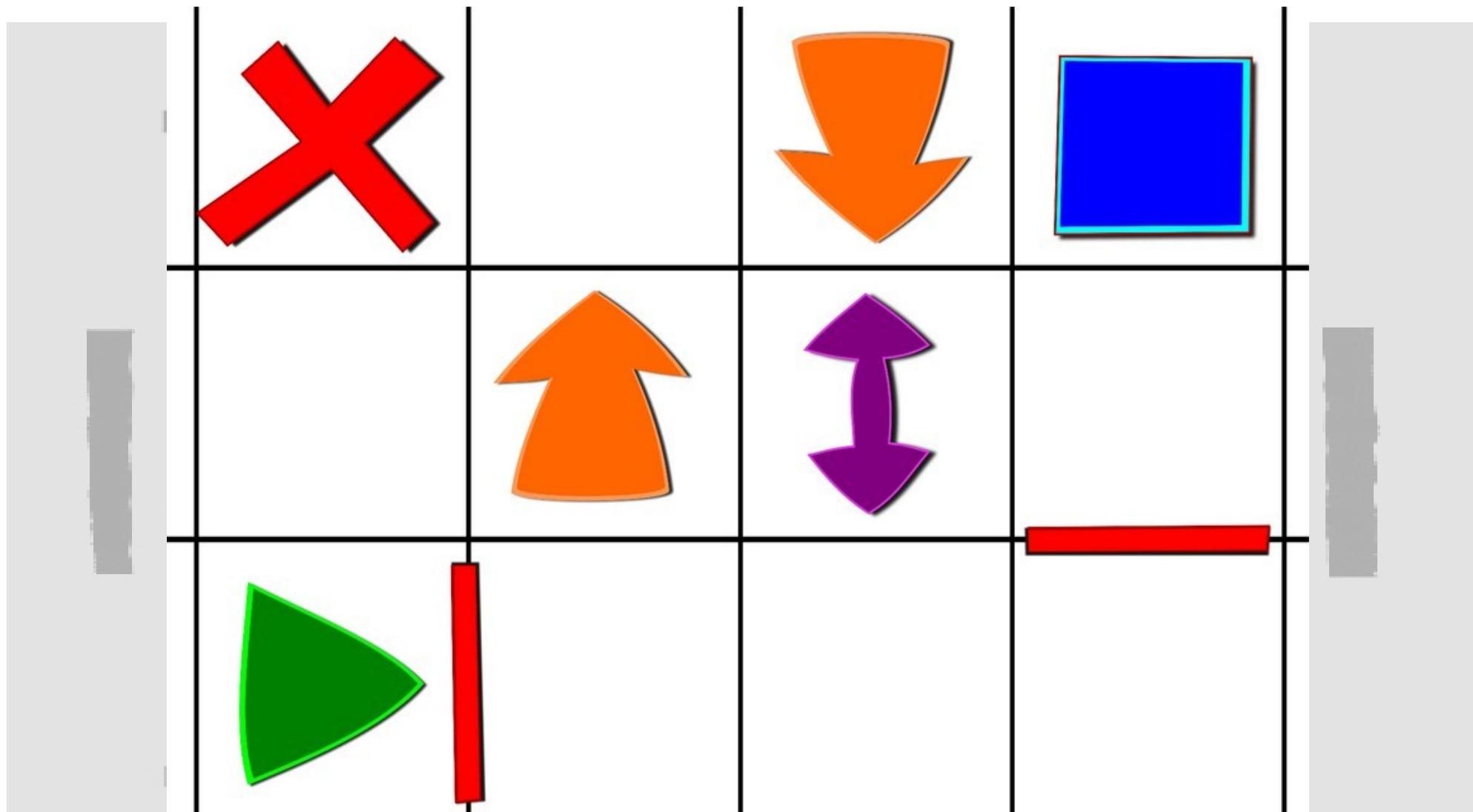
```
Blink | Arduino 1.8.5

This example code is in the public domain.
http://www.arduino.cc/en/Tutorial/Blink
/*
 * the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

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

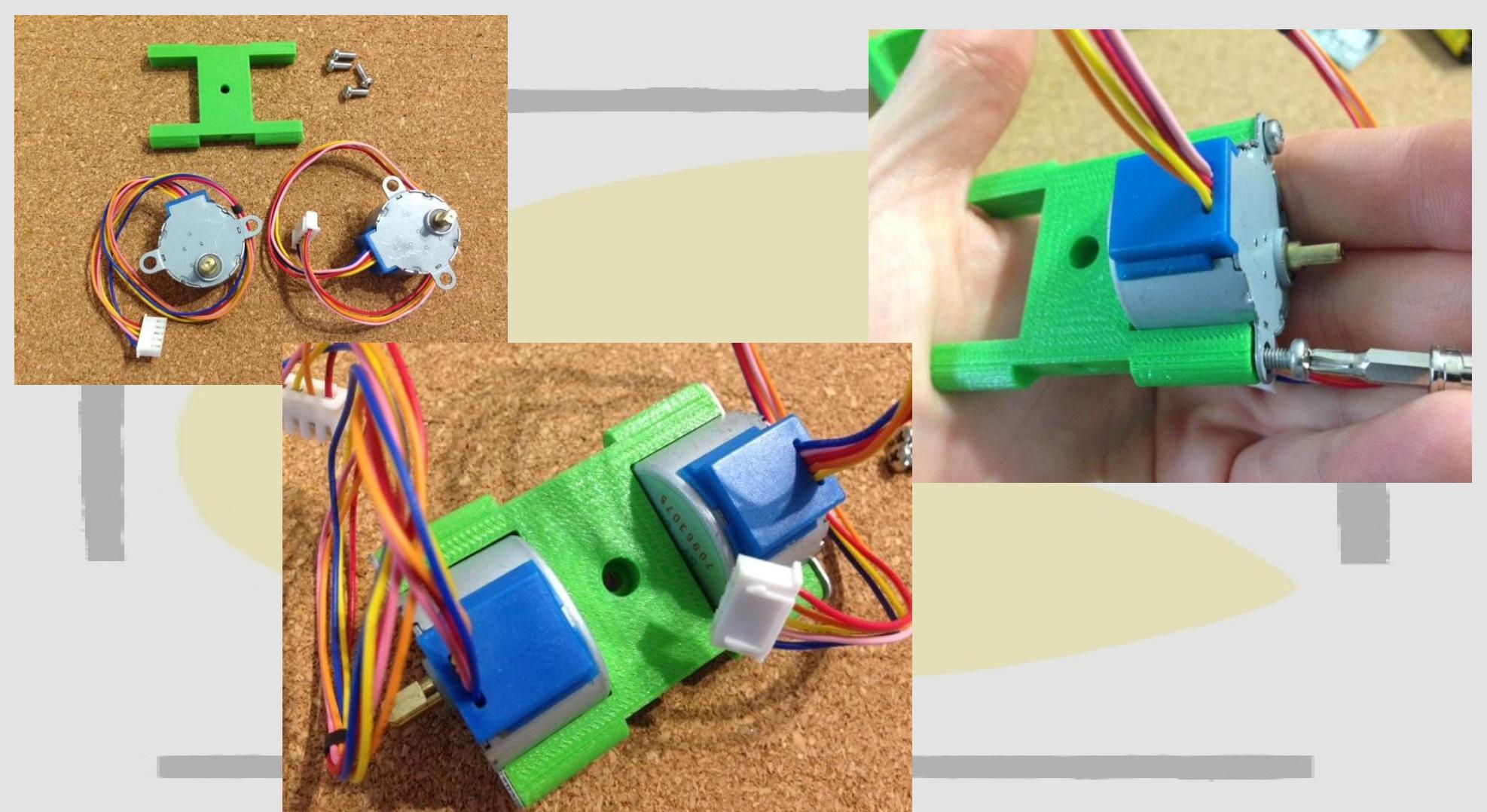
32 Arduino/Genuino Uno on COM1

Juegos con Escornabot

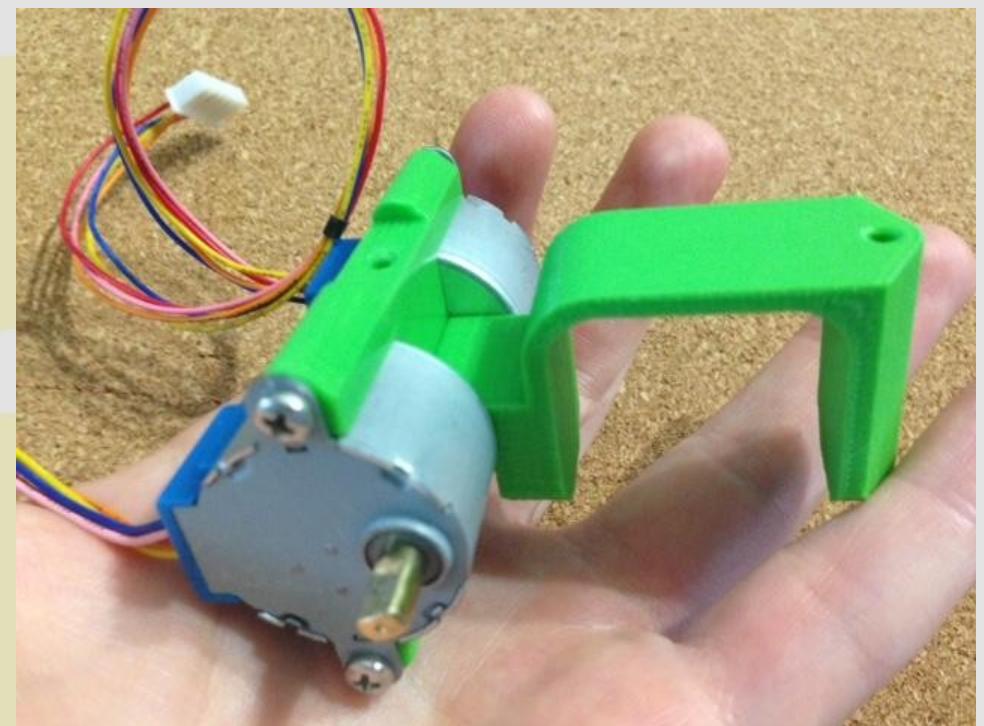




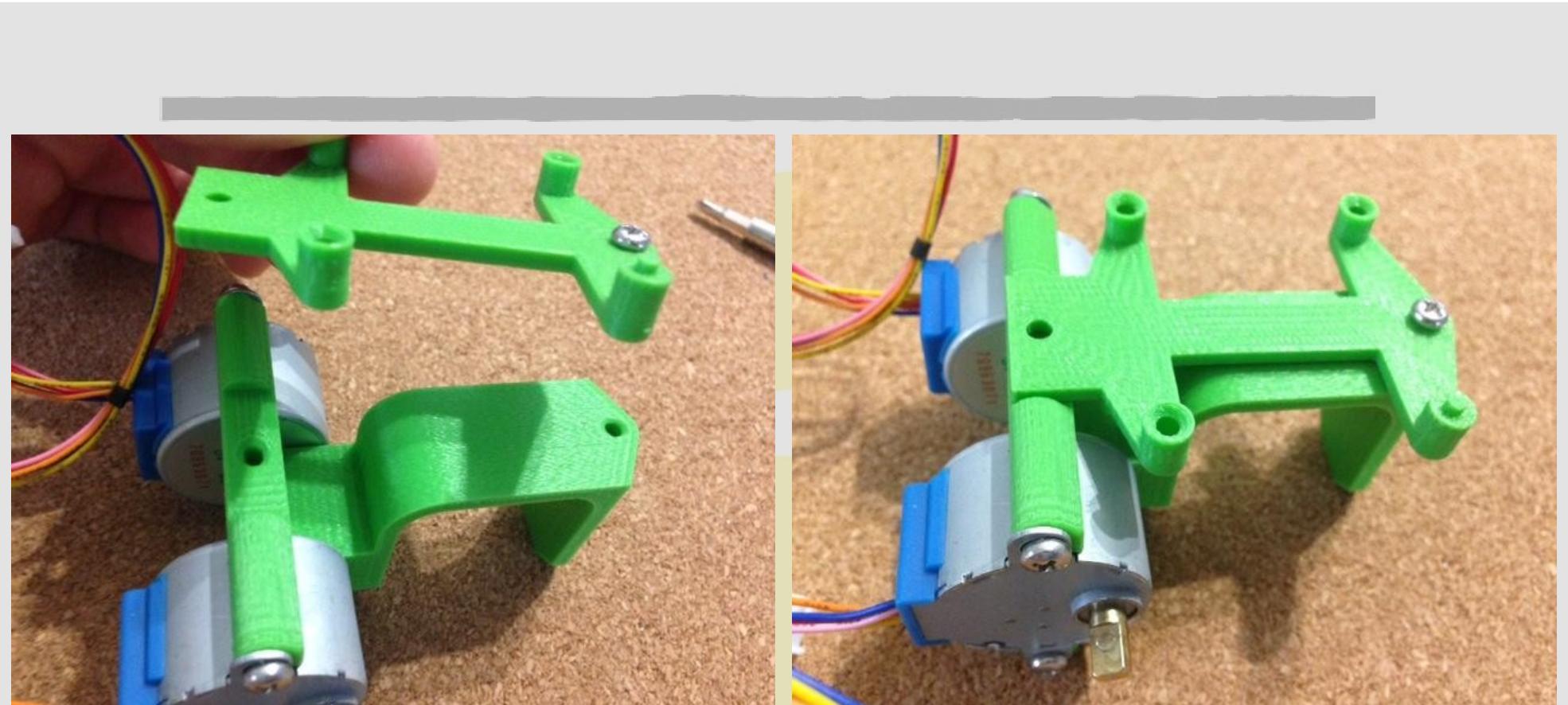
Motores



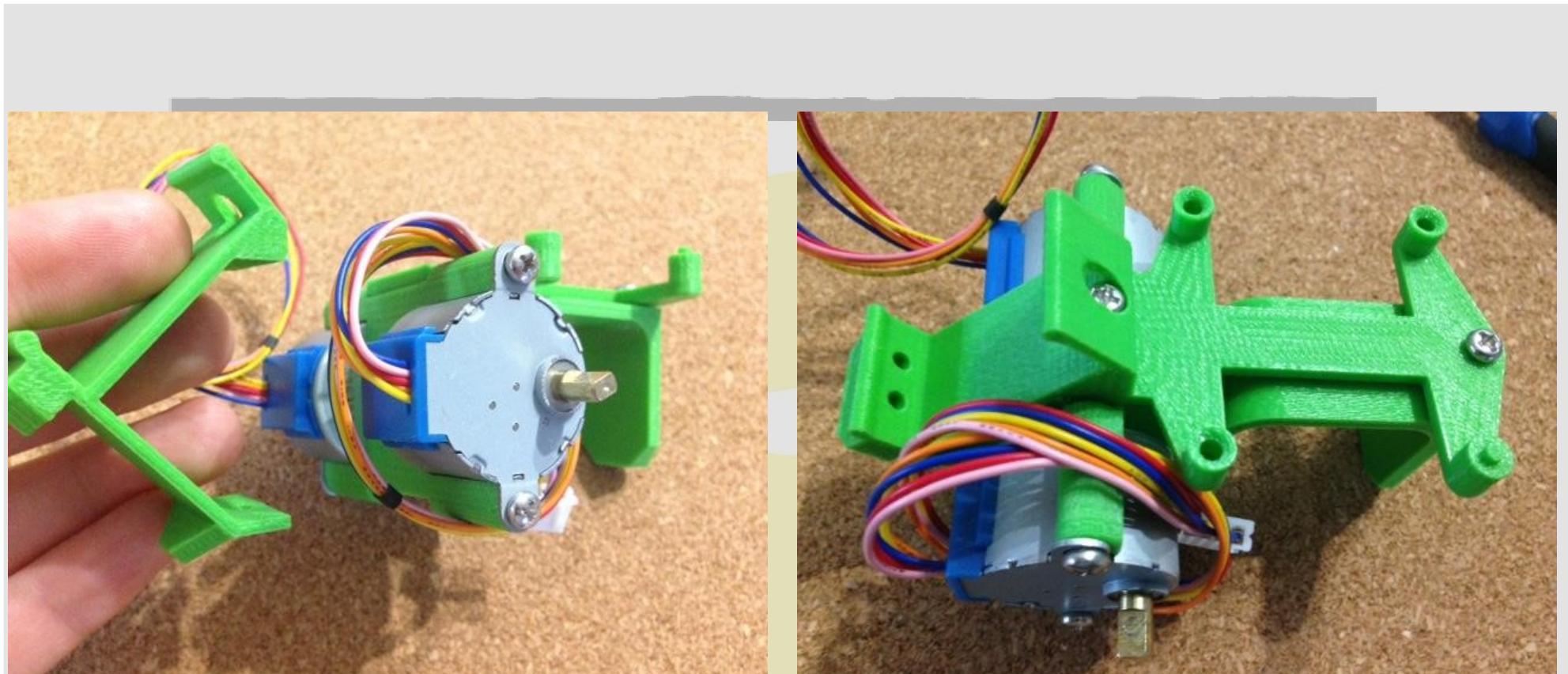
Soporte portapilas



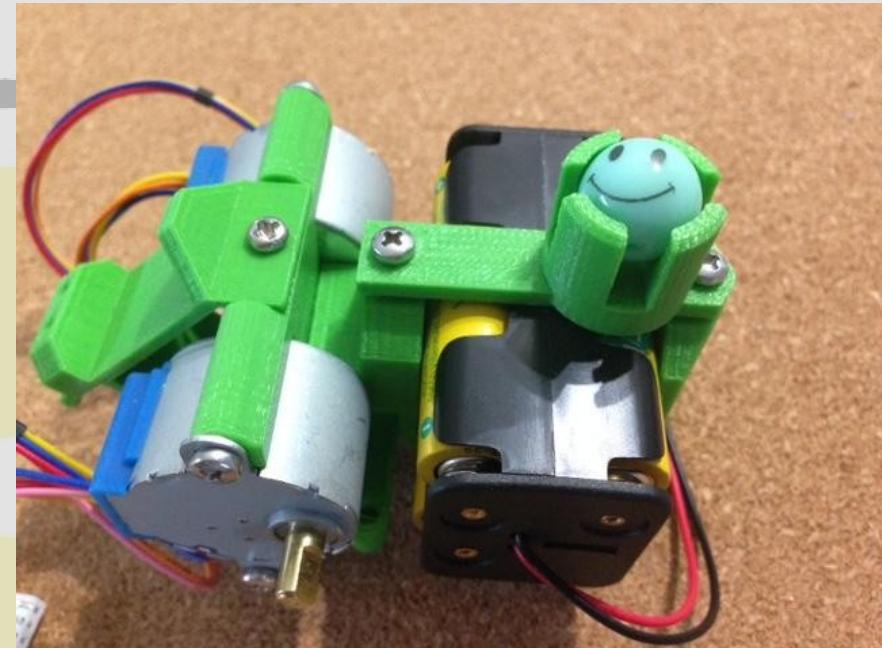
Soporte botonera



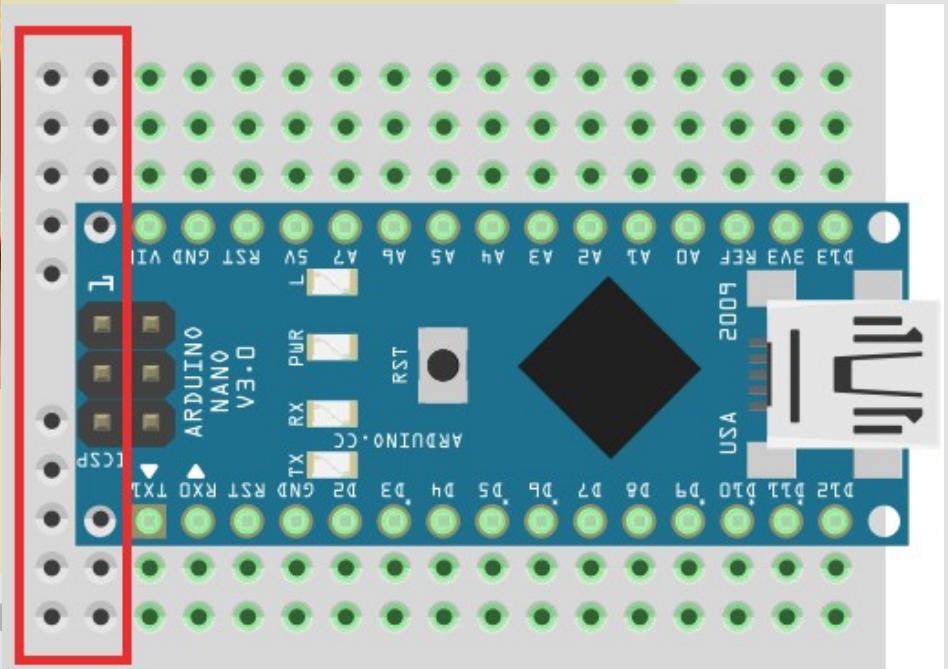
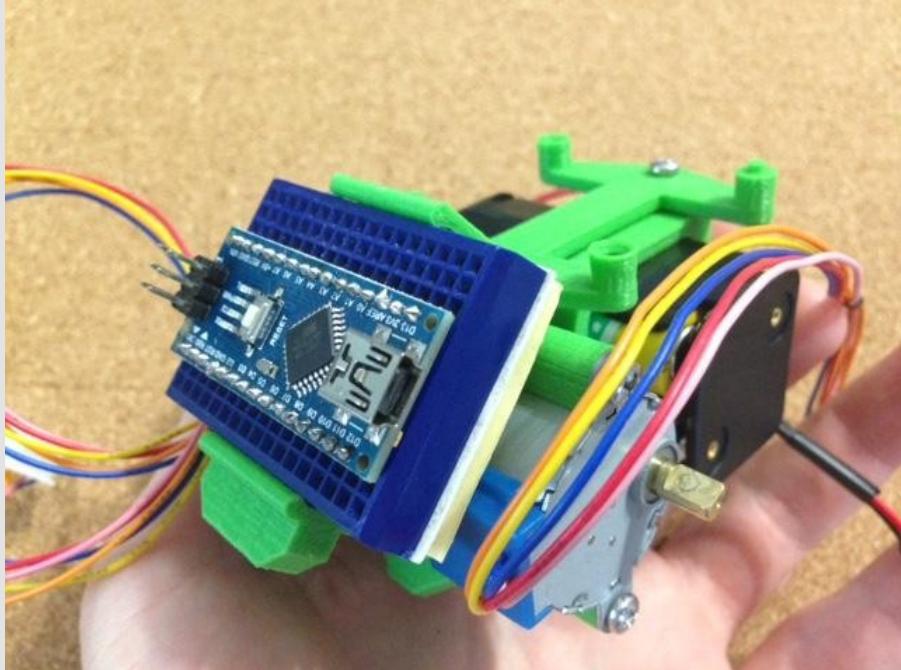
Soporte board



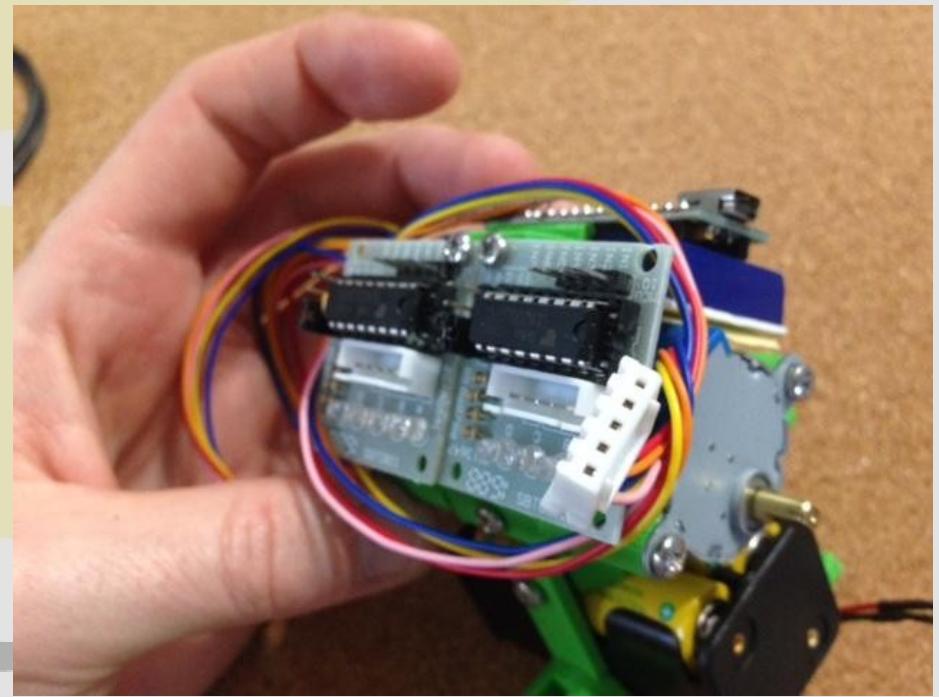
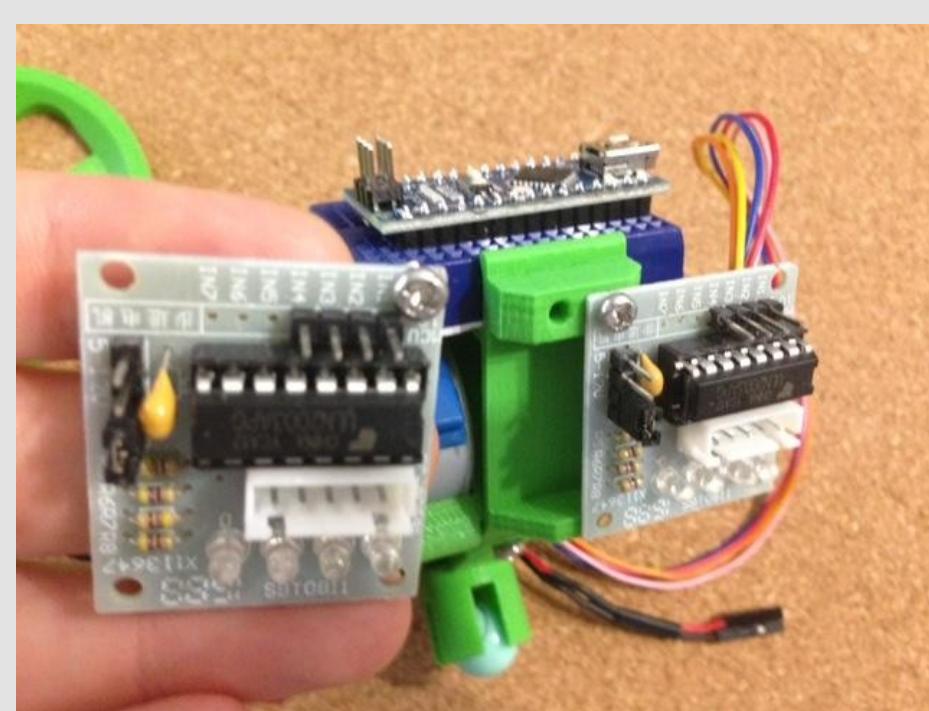
Portapilas y bola



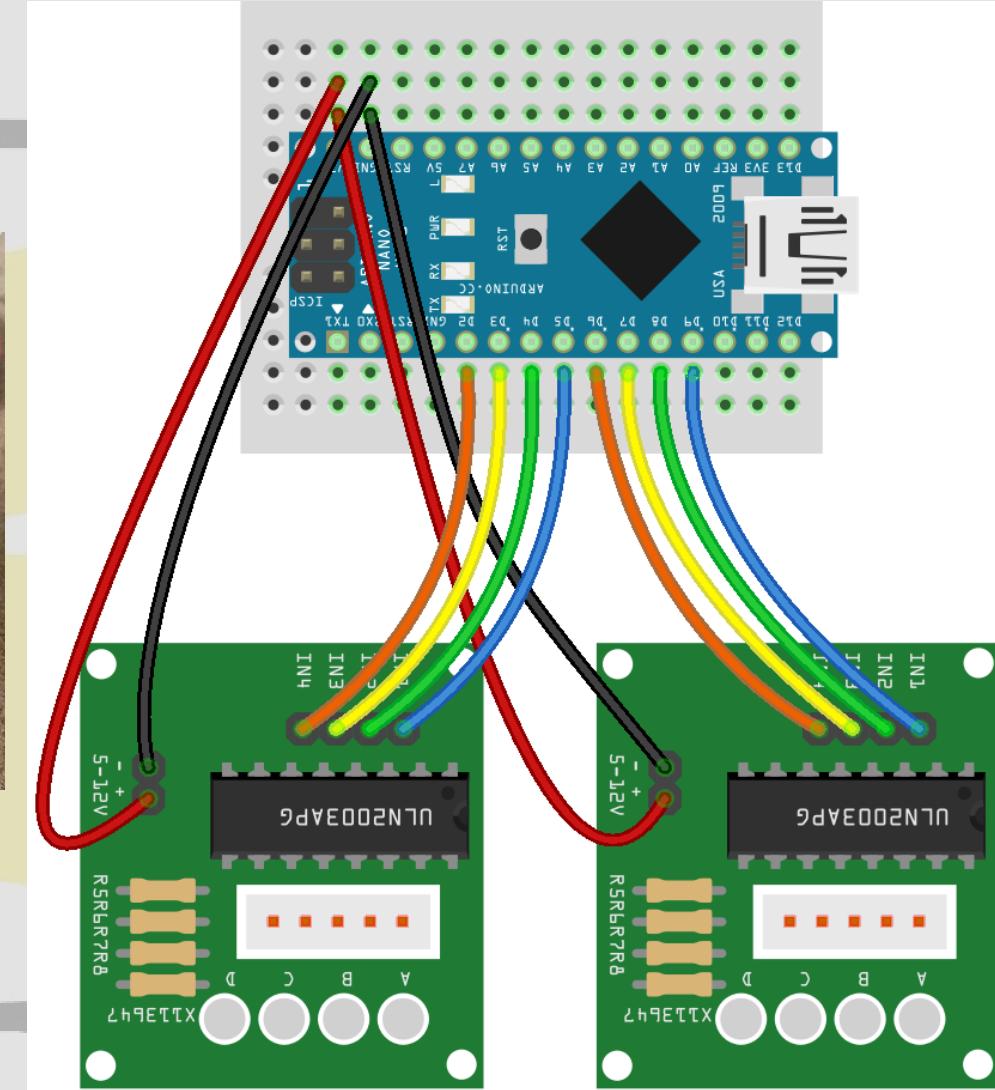
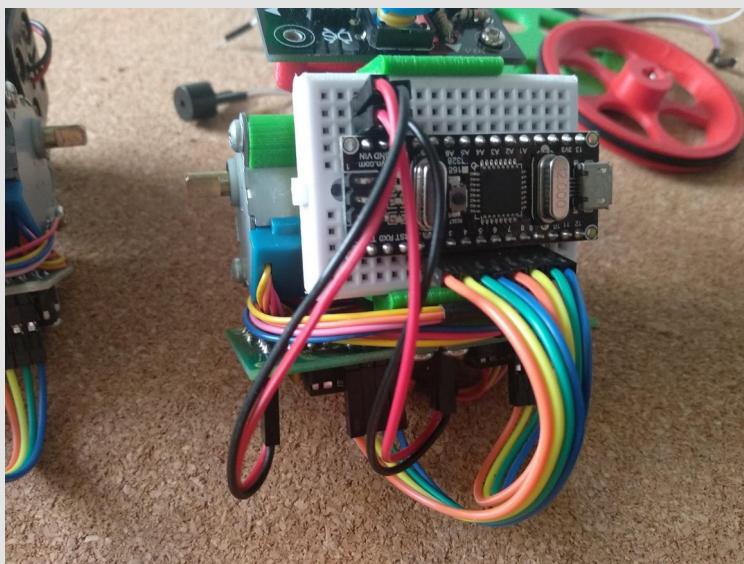
Board



Drivers de motores

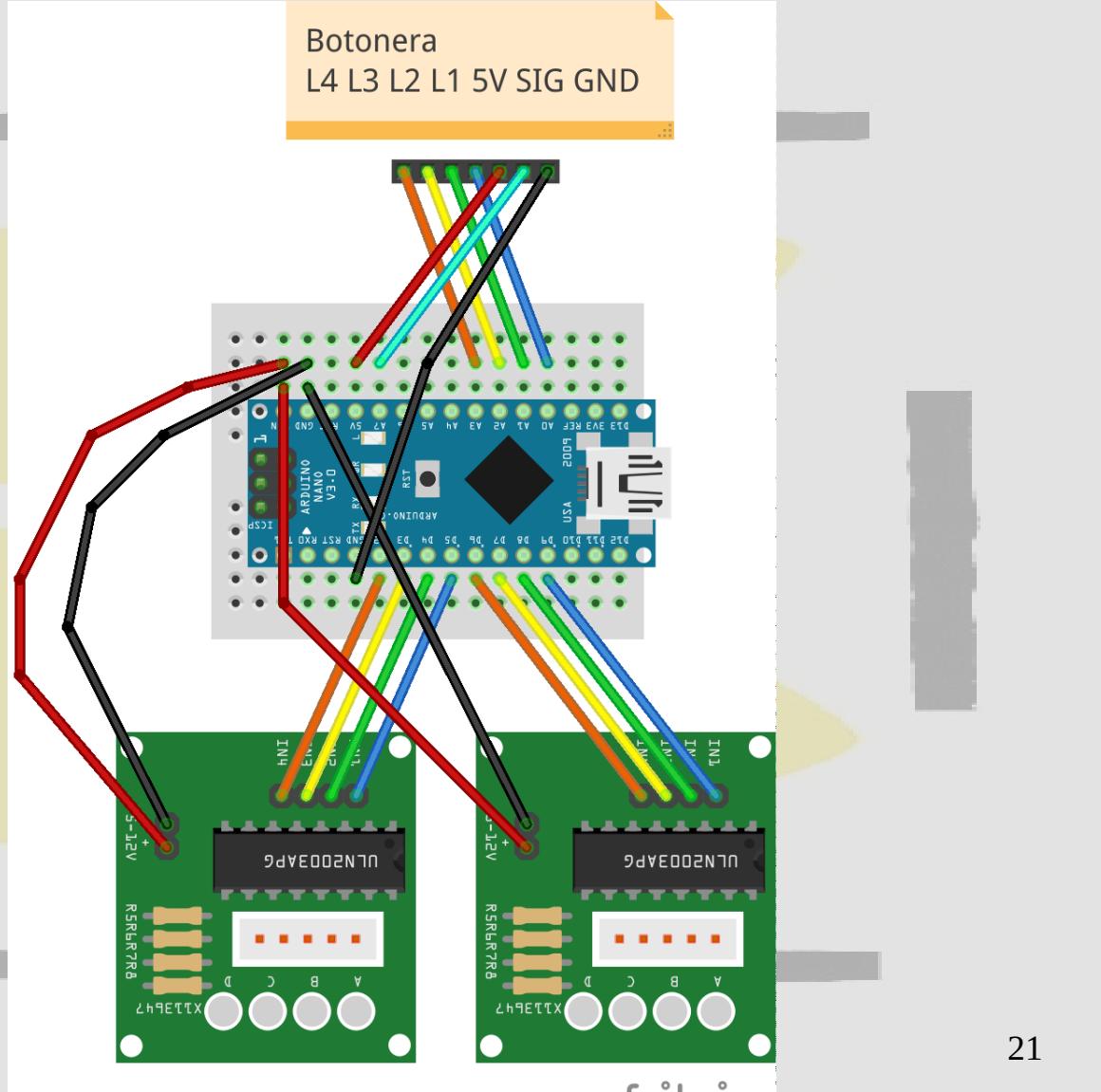


Conexionado drivers



martes, 5 de noviembre de 2019

Conexionado Botonera

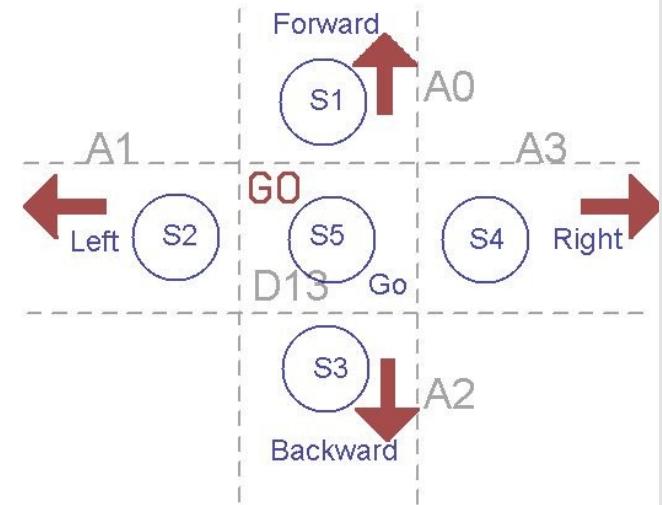
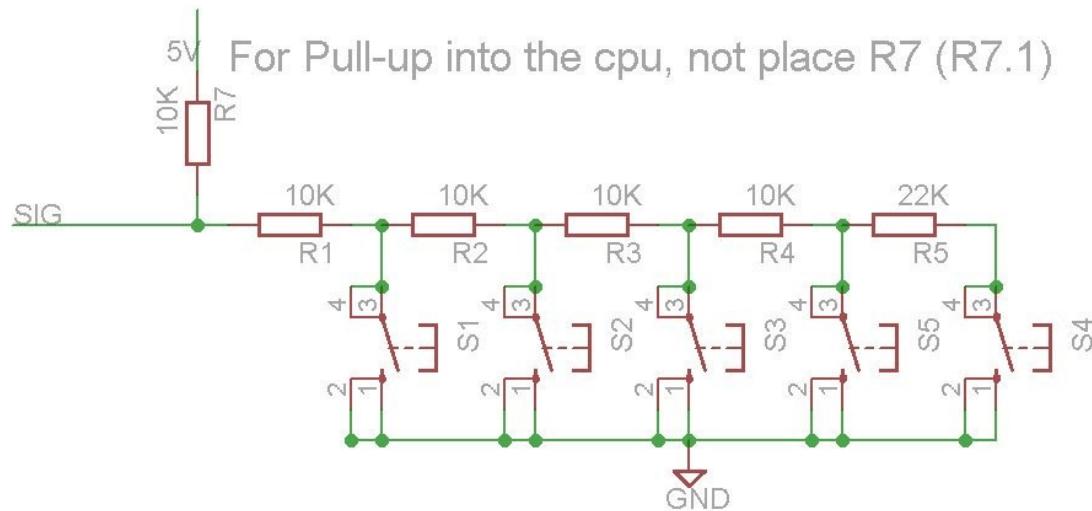


Conexionado botonera

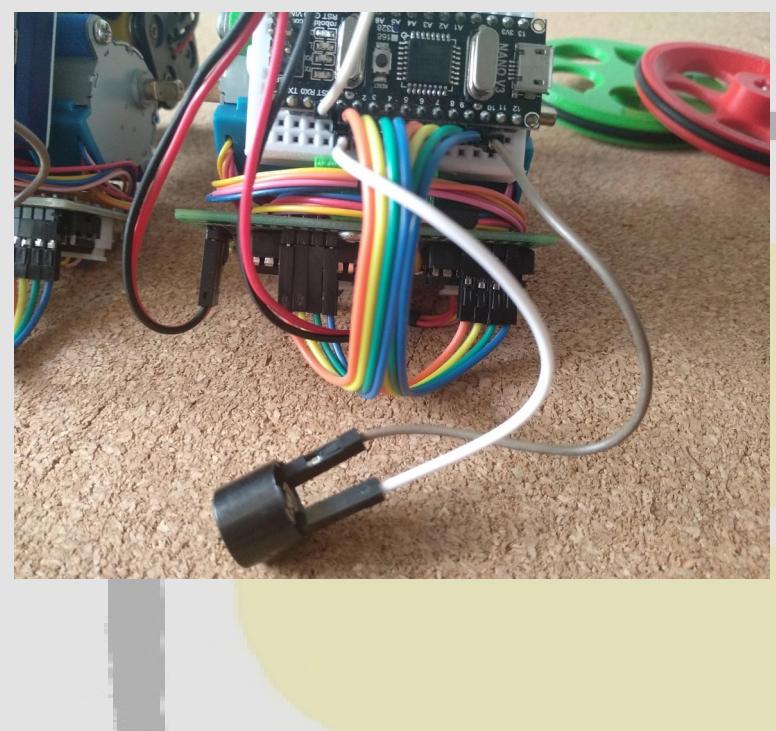
- Pin gnd: gnd de abajo (al lado D2)
- Pin 5V: 5V de arriba
- Pin Signal: A7 (arriba)
- Pin L1: A0
- Pin L2: A1
- Pin L3: A2
- Pin L4: A3

Botonera

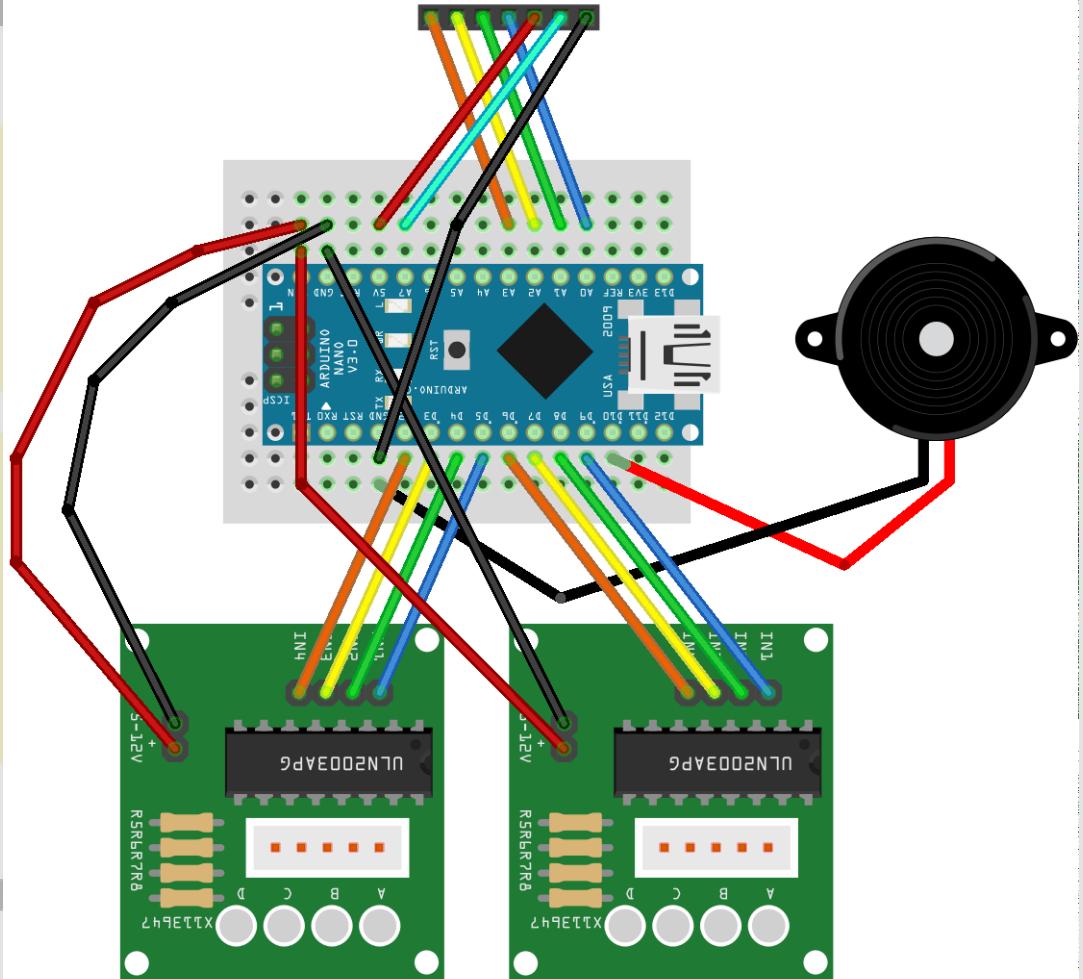
For Pull-up into the cpu, not place R7 (R7.1)



Zumbador y conjunto



Botonera
L4 L3 L2 L1 5V SIG GND



martes, 5 de noviembre de 2019

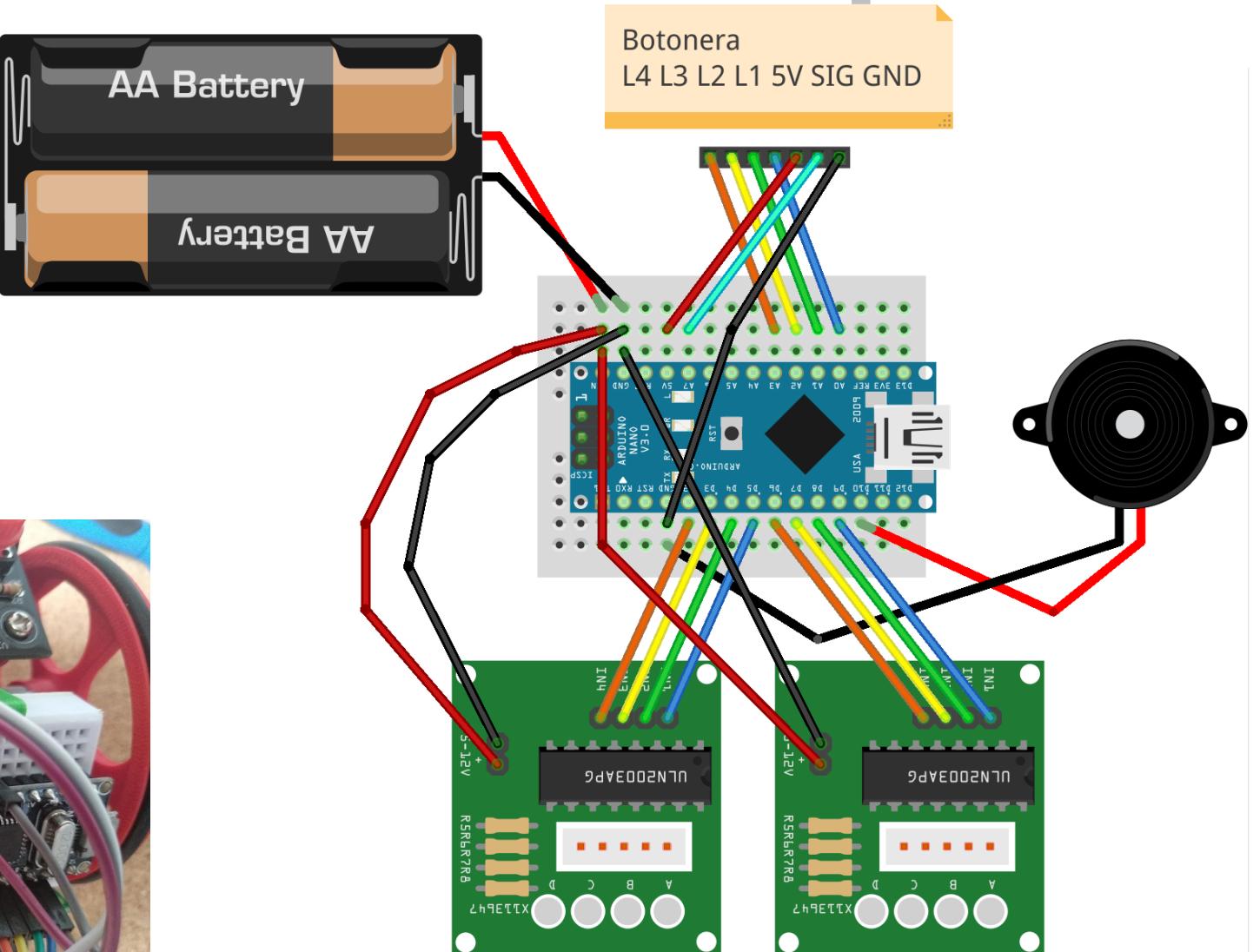
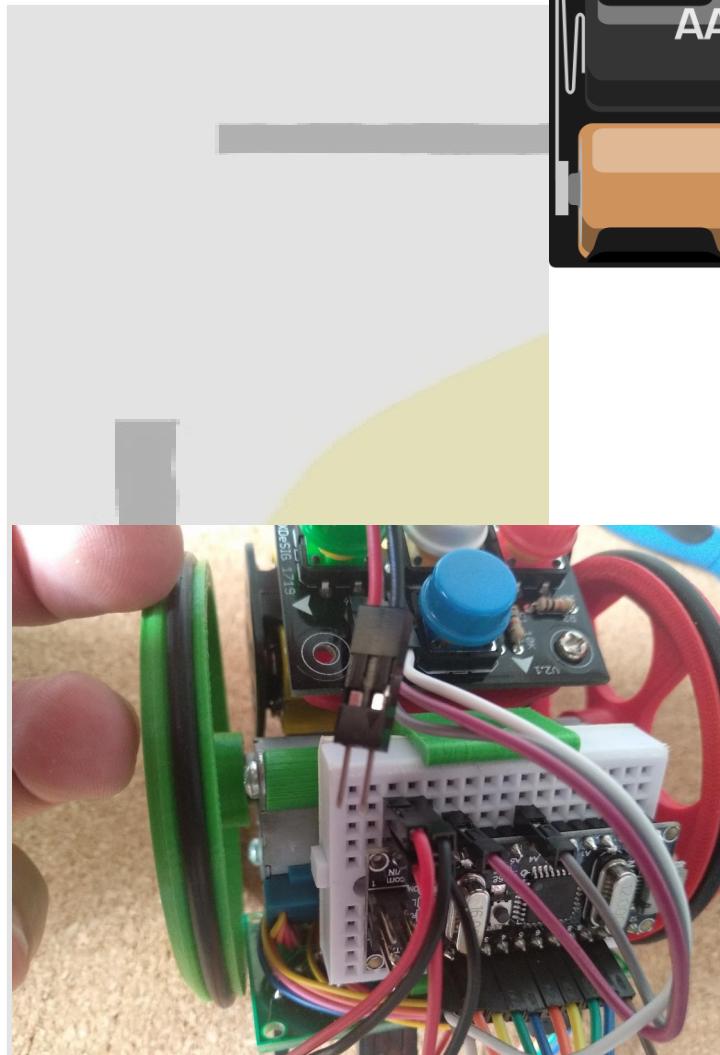
24

fritzing

Ruedas



Conexionado Portapilas



martes, 5 de noviembre de 2019

Modos firmware 1.6.2

- **Modo normal**
 - Pulsación corta: giros 90°
 - Pulsación larga: giros 45°
- **Modo 60° (tecla GO pulsación larga)**
 - Pulsación corta: giros 60°
 - Pulsación larga: giros 120°
- **Pausa:** pulsación larga tecla atrás.

Cambios en firmware

- Abrimos **Escornabot.ino**, pestaña **Configuration.h**
 - **#define STEPPERS_STEPS_PER_SECOND 1000**
Número de pasos por segundo, el tope anda sobre 2300
 - **#define STEPPERS_LINE_STEPS 1738**
Da un avance de 10cm
 - **#define STEPPERS_TURN_STEPS 1024**
Establece un giro de 90º,
Una vuelta completa 4096 pasos

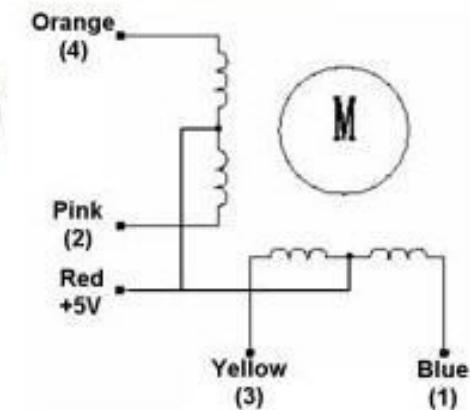
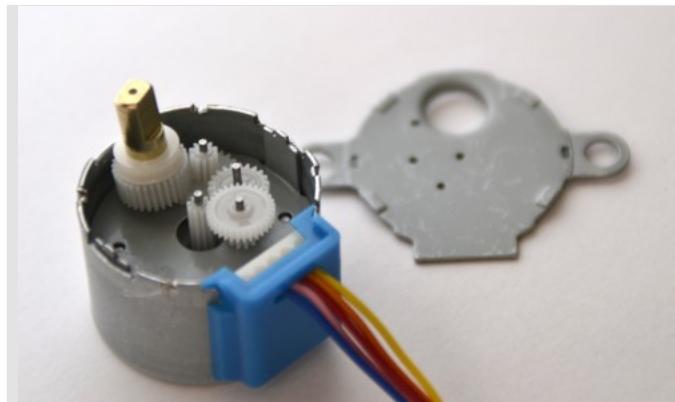
```
#ifdef ENGINE_TYPE_STEPPERS

    // stepper pin setup (digital outputs)
#define STEPPERS_MOTOR_RIGHT_IN1 5
#define STEPPERS_MOTOR_RIGHT_IN2 4
#define STEPPERS_MOTOR_RIGHT_IN3 3
#define STEPPERS_MOTOR_RIGHT_IN4 2
#define STEPPERS_MOTOR_LEFT_IN1 9
#define STEPPERS_MOTOR_LEFT_IN2 8
#define STEPPERS_MOTOR_LEFT_IN3 7
#define STEPPERS_MOTOR_LEFT_IN4 6

    // step calibration
#define STEPPERS_STEPS_PER_SECOND 1000
#define STEPPERS_LINE_STEPS 1738
#define STEPPERS_TURN_STEPS 1024

#endif
```

Motor paso a paso



Half-Step Switching Sequence

Lead Wire Color	---> CW Direction (1-2 Phase)							
	1	2	3	4	5	6	7	8
4 Orange	-	-						-
3 Yellow		-	-	-				
2 Pink					-	-	-	
1 Blue						-	-	-

64 pasos/vuelta x 64 reductora
= **4096 pasos para una vuelta**

