Arduino básico



Arduino





Logo de Arduino

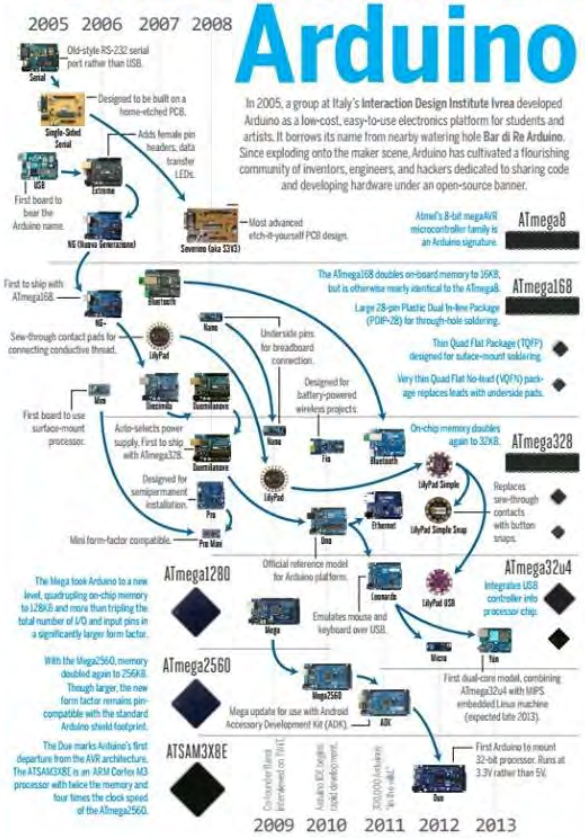


Hernando Barragán



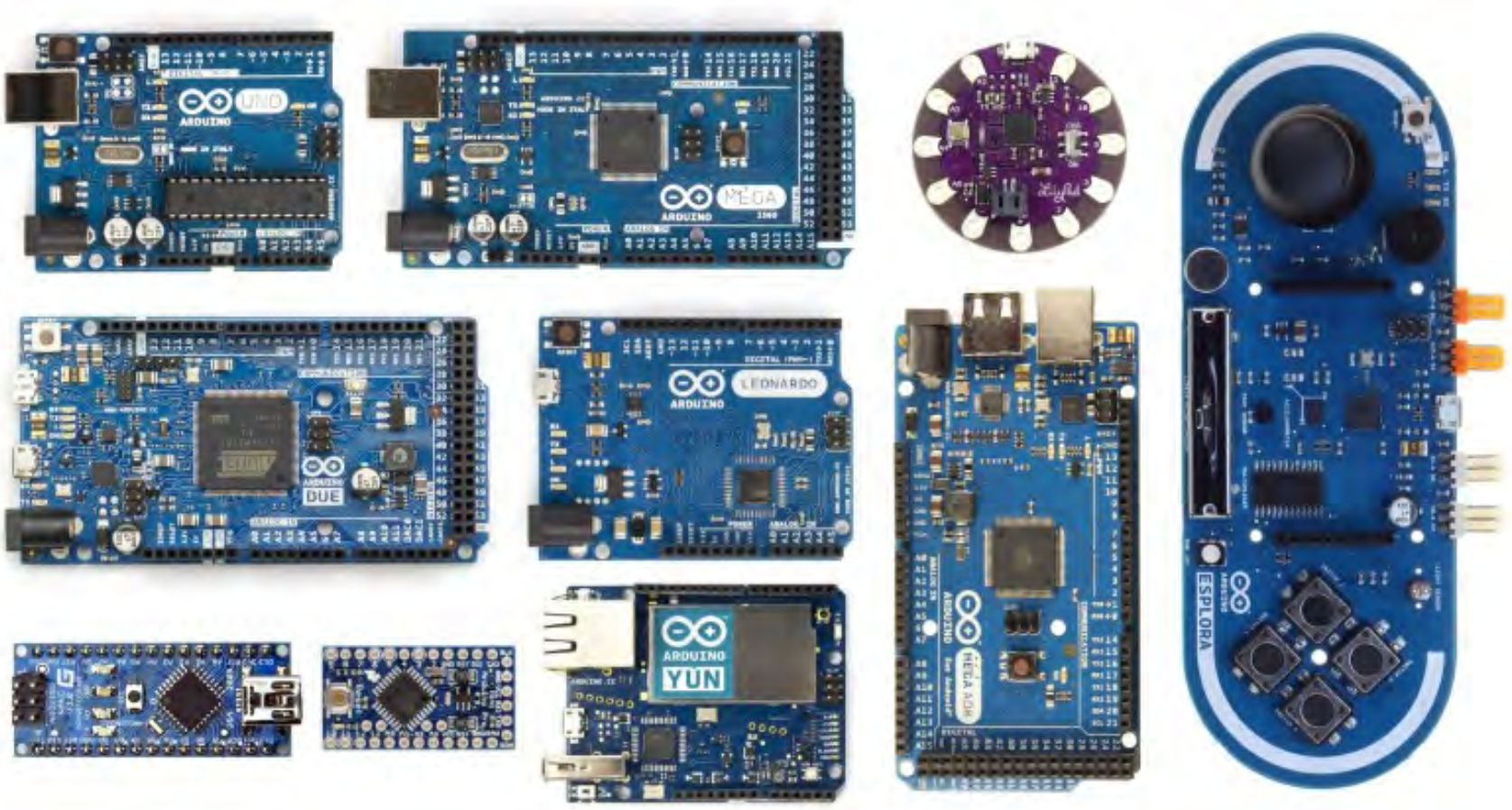
Equipo de Arduino

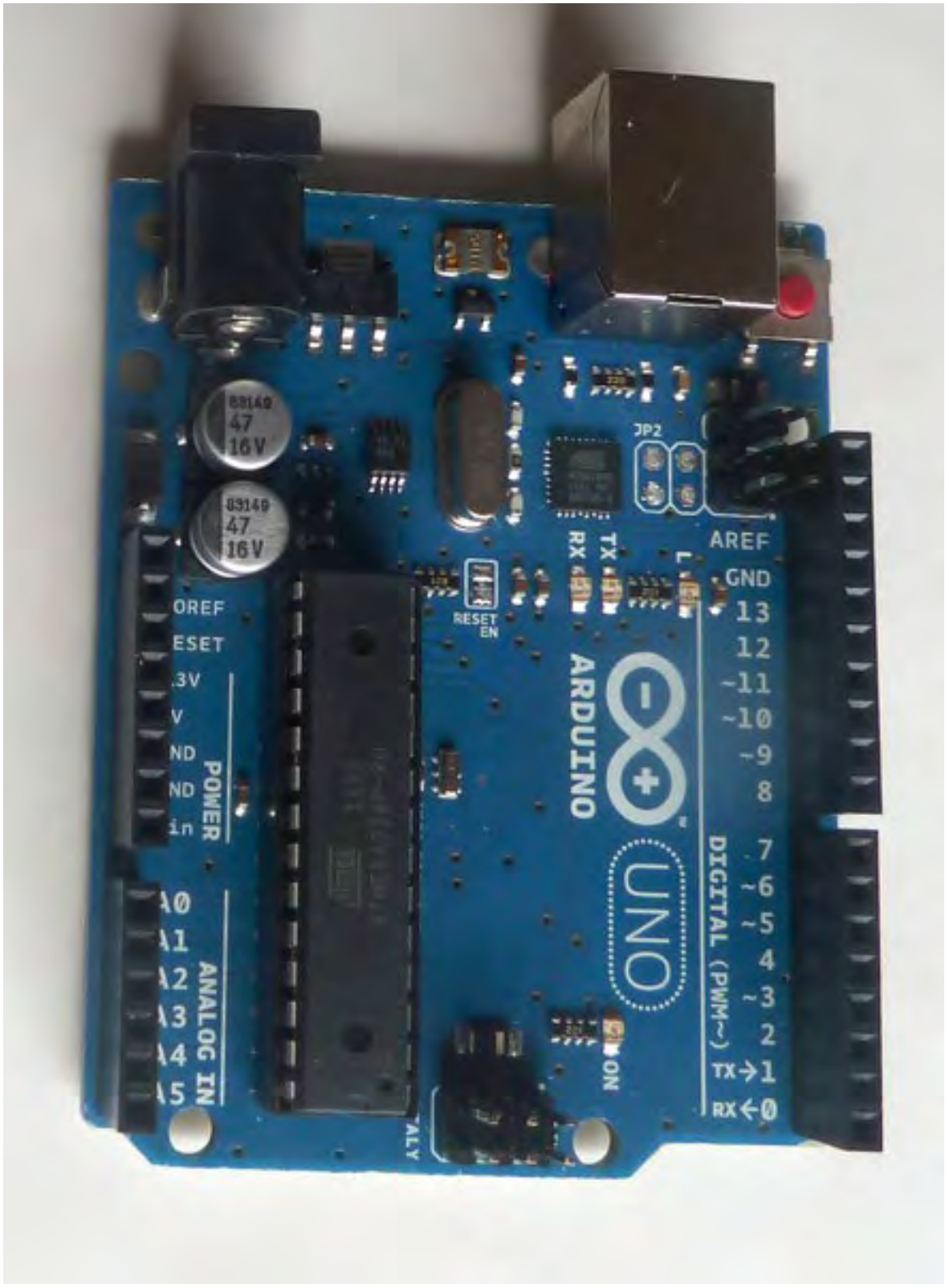
Evolution of Arduino

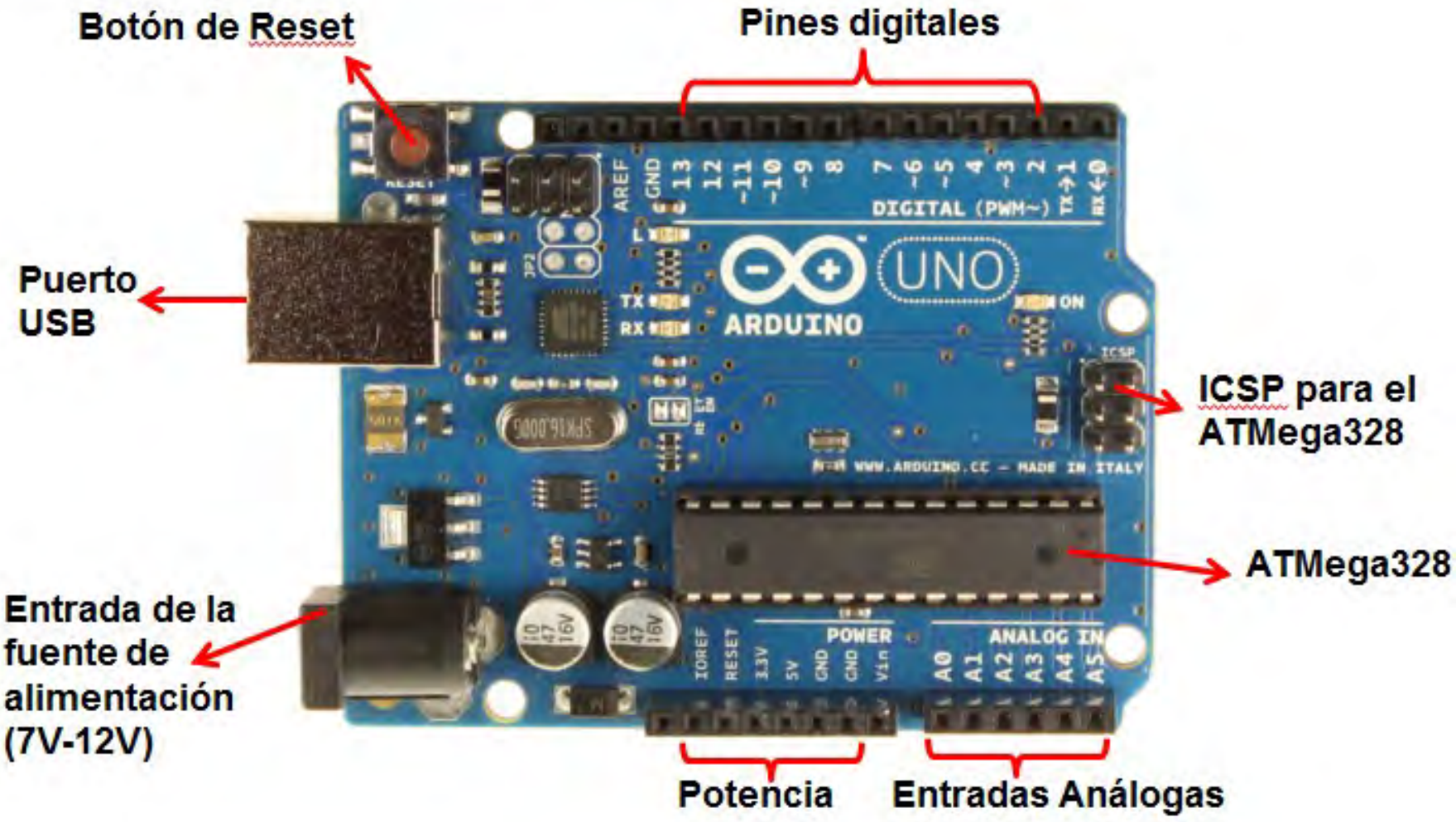


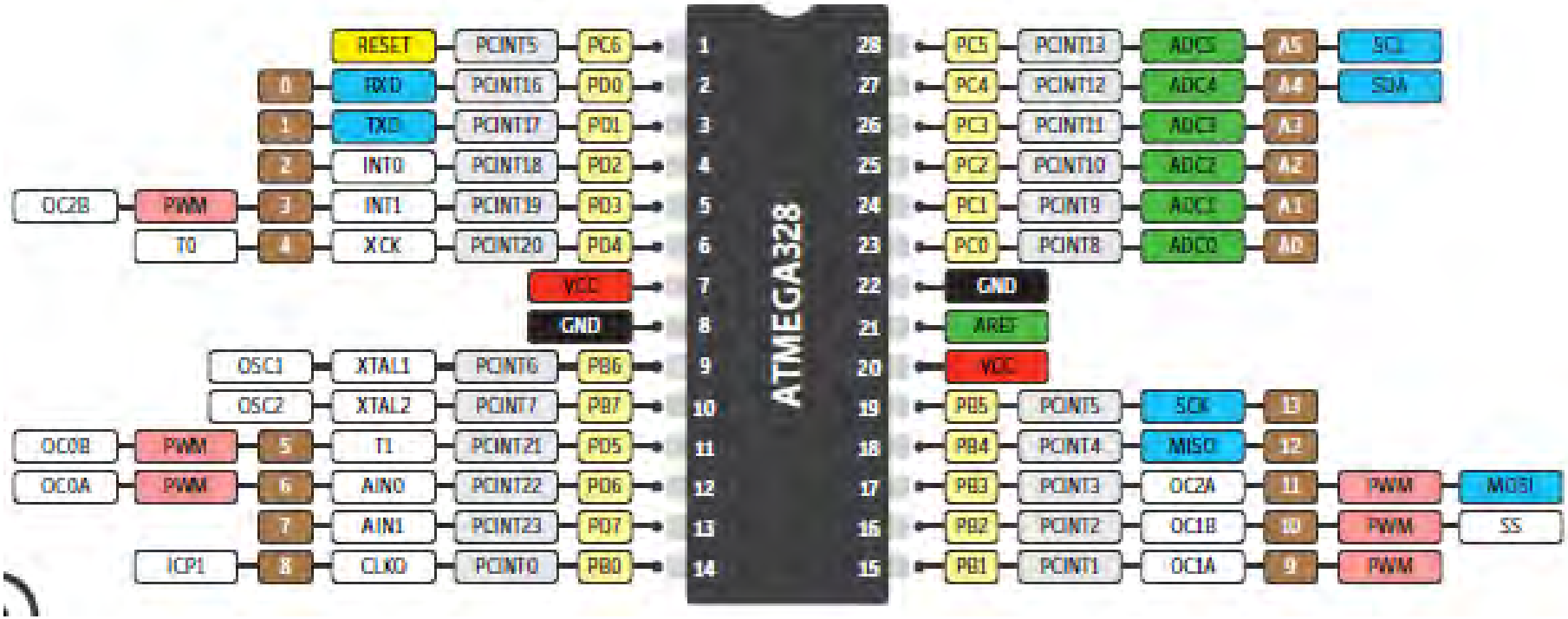
Make:

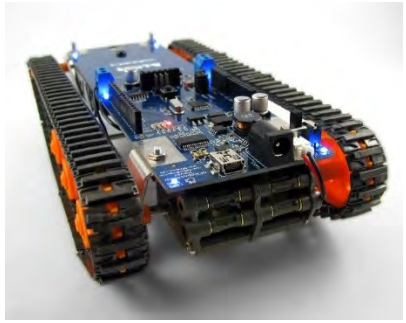

Ardupilot Mega 2.5.2
ARDUCOPTER


ARDUINO
UNO
MEGA
DUE
LEONARDO
YUN
ESPLORA

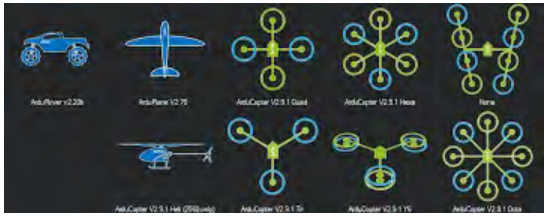

JP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM~)
ARDUINO
UNO
RESET
EN
TX
RX
ON
POWER
ANALOG IN
A0
A1
A2
A3
A4
A5
47
16V

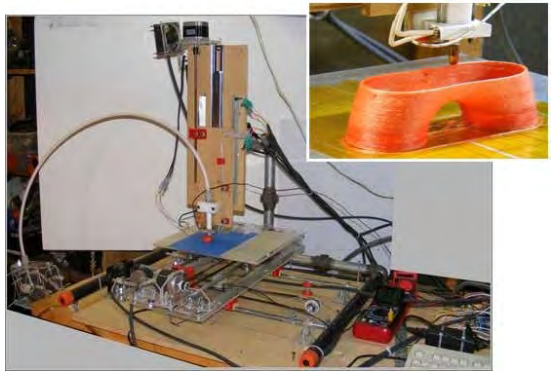

Botón de Reset
Pines digitales
Puerto USB
ICSP para el ATMega328
ATMega328
Entrada de la fuente de alimentación (7V-12V)
Potencia
Entradas Análogas
DIGITAL (PWM~)
POWER
ANALOG IN
ARDUINO
UNO
WWW.ARDUINO.CC – MADE IN ITALY

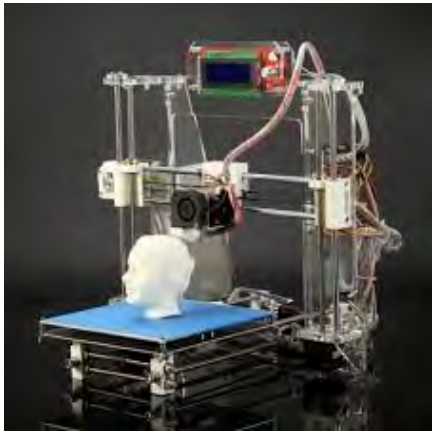

ATMEGA328
RESET PCINT14 PC6 1
0 RXD PCINT16 PD0 2
1 TXD PCINT17 PD1 3
2 INT0 PCINT18 PD2 4
OC2B PWM 3 INT1 PCINT19 PD3 5
T0 4 XCK PCINT20 PD4 6
VCC 7
GND 8
OSC1 XTAL1 PCINT6 PB6 9
OSC2 XTAL2 PCINT7 PB7 10
OC0B PWM 5 T1 PCINT21 PD5 11
OC0A PWM 6 AIN0 PCINT22 PD6 12
7 AIN1 PCINT23 PD7 13
ICP1 8 CLKO PCINT0 PB0 14
28 PC5 PCINT13 ADC5 A5 SCL
27 PC4 PCINT12 ADC4 A4 SDA
26 PC3 PCINT11 ADC3 A3
25 PC2 PCINT10 ADC2 A2
24 PC1 PCINT9 ADC1 A1
23 PC0 PCINT8 ADC0 A0
22 GND
21 AREF
20 VCC
19 PB5 PCINT5 SCK 13
18 PB4 PCINT4 MISO 12
17 PB3 PCINT3 OC2A 11 PWM MOSI
16 PB2 PCINT2 OC1B 10 PWM SS
15 PB1 PCINT1 OC1A 9 PWM

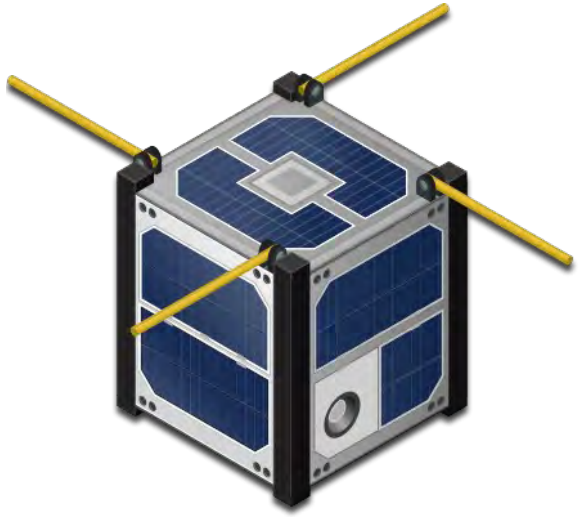


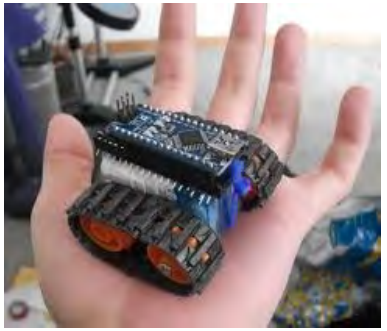












Ahora

Vamos a jugar

Prender un LED

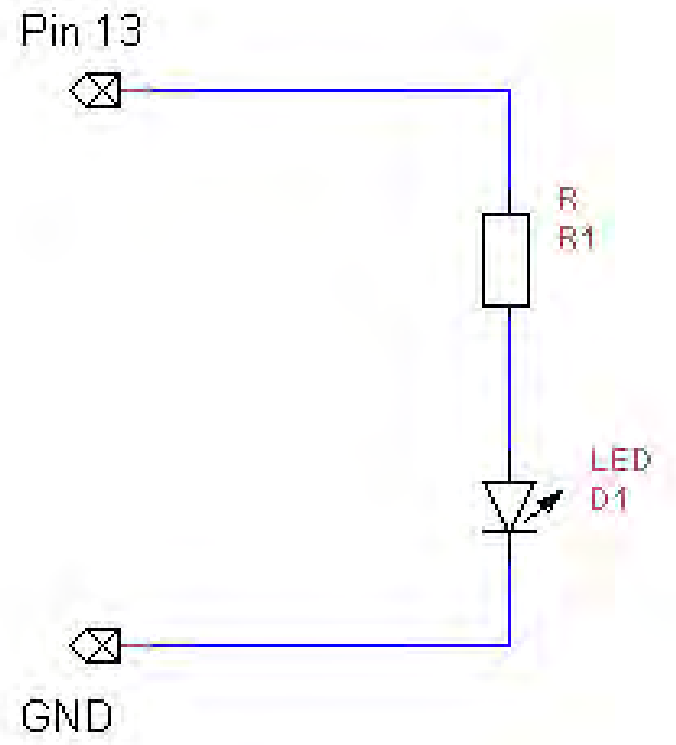
Hola mundo arduino

Programamos el LED 13

Salidas digitales

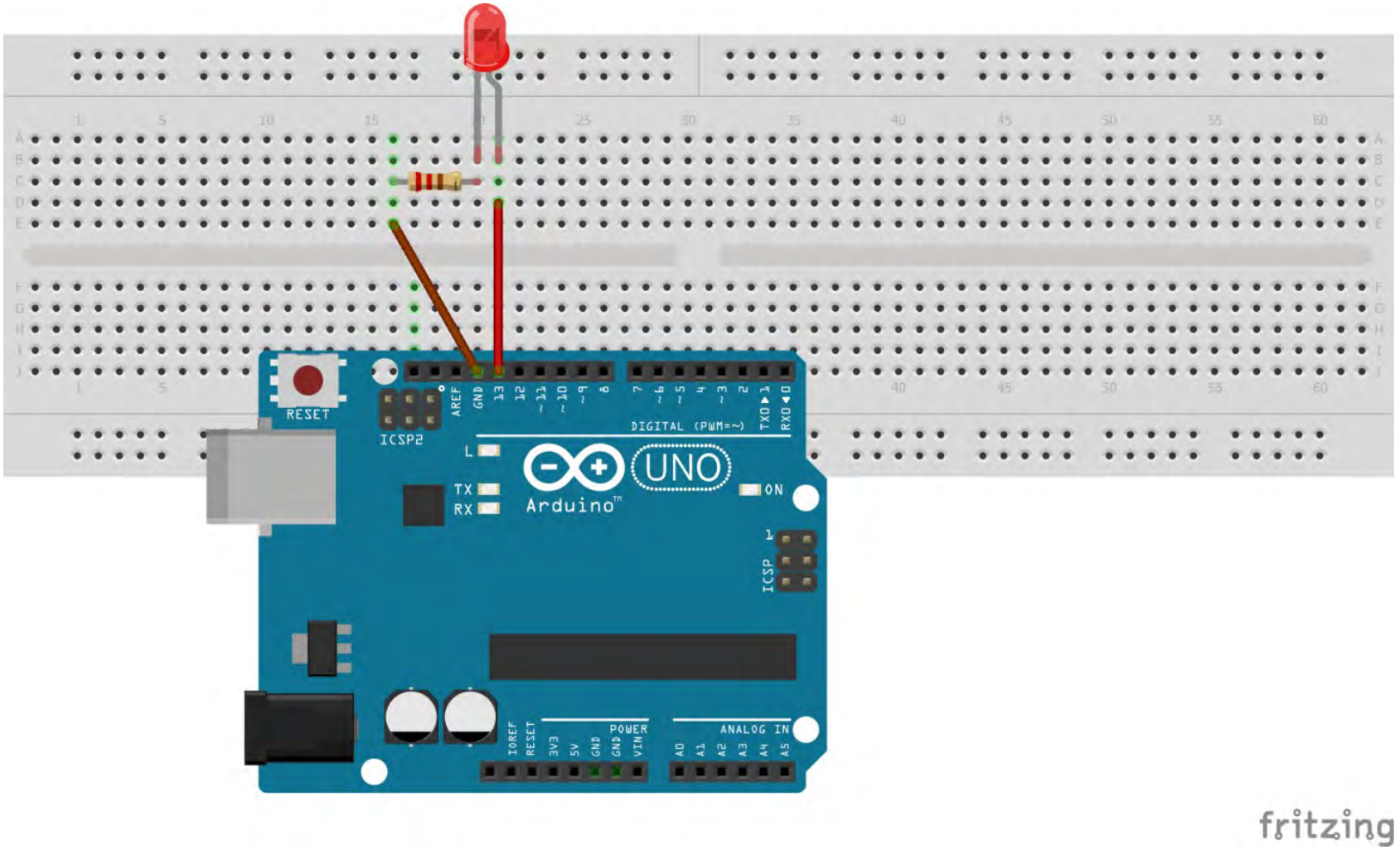
Prender un LED

Hola mundo arduino



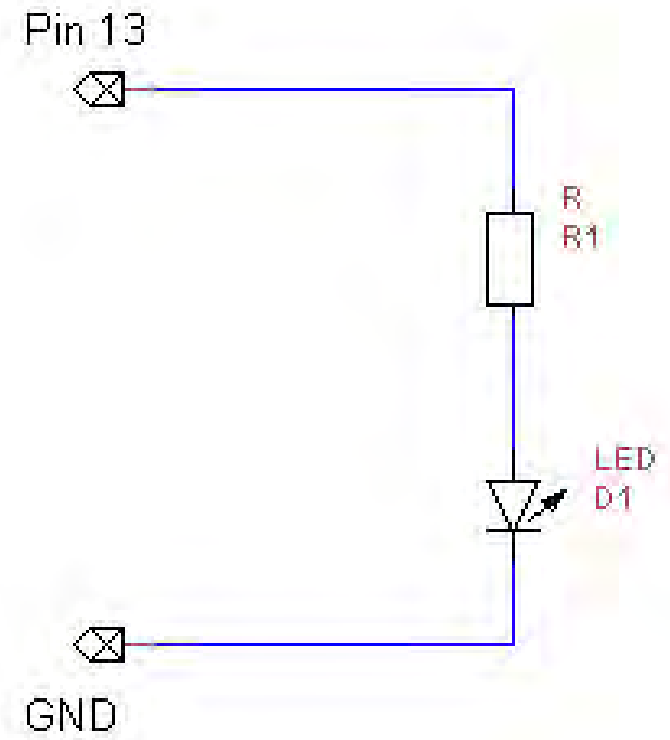
Material:

- Arduino Uno
- Un LED
- Cables
- Resistencia 220
- Protoboard


RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM=~)
L
TX
RX
Arduino™
UNO
ON
ICSP
POWER
ANALOG IN
IOREF
RESET
3V3
5V
GND
GND
VIN
A0
A1
A2
A3
A4
A5

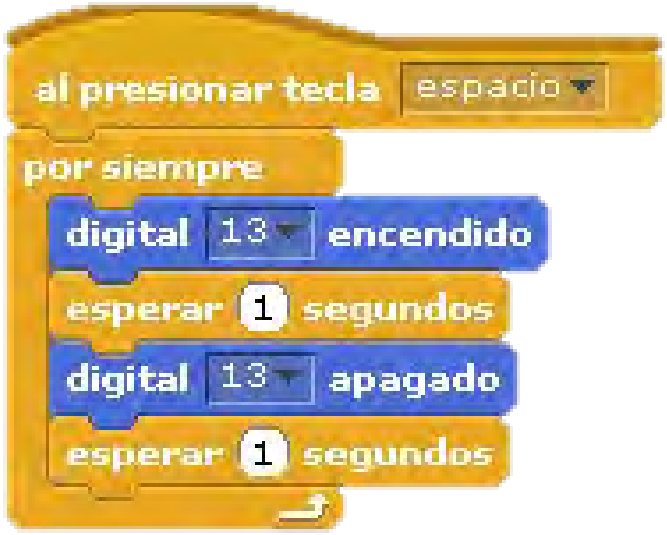
fritzing

LED intermitente




al presionar
por siempre
digital 13 encendido
esperar 1 segundos
digital 13 apagado
esperar 1 segundos

LED con teclado - lectura serial



Archivo / Ejemplo / basics / 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
}
```

```
int ledPin =  13;      // the number of the LED pin

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
    digitalWrite(ledPin, HIGH);
   delay(1000);
    digitalWrite(ledPin, LOW);
   delay(1000);
  }
```

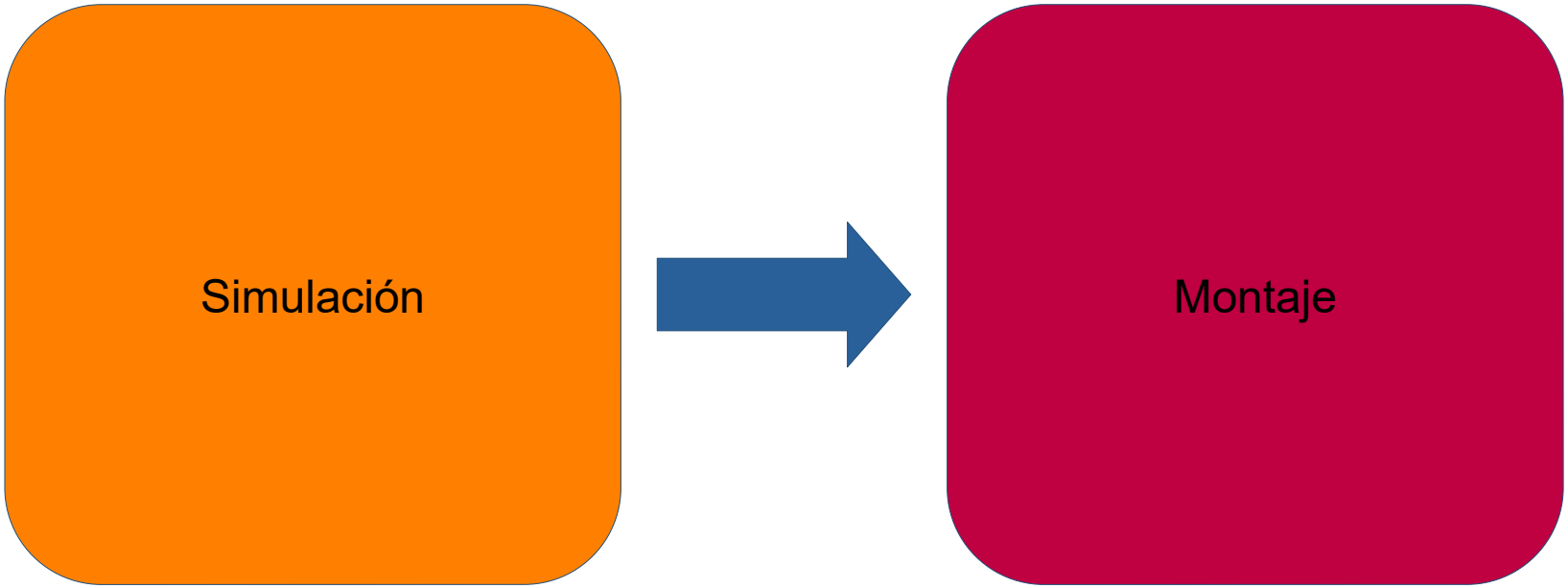
Simulación

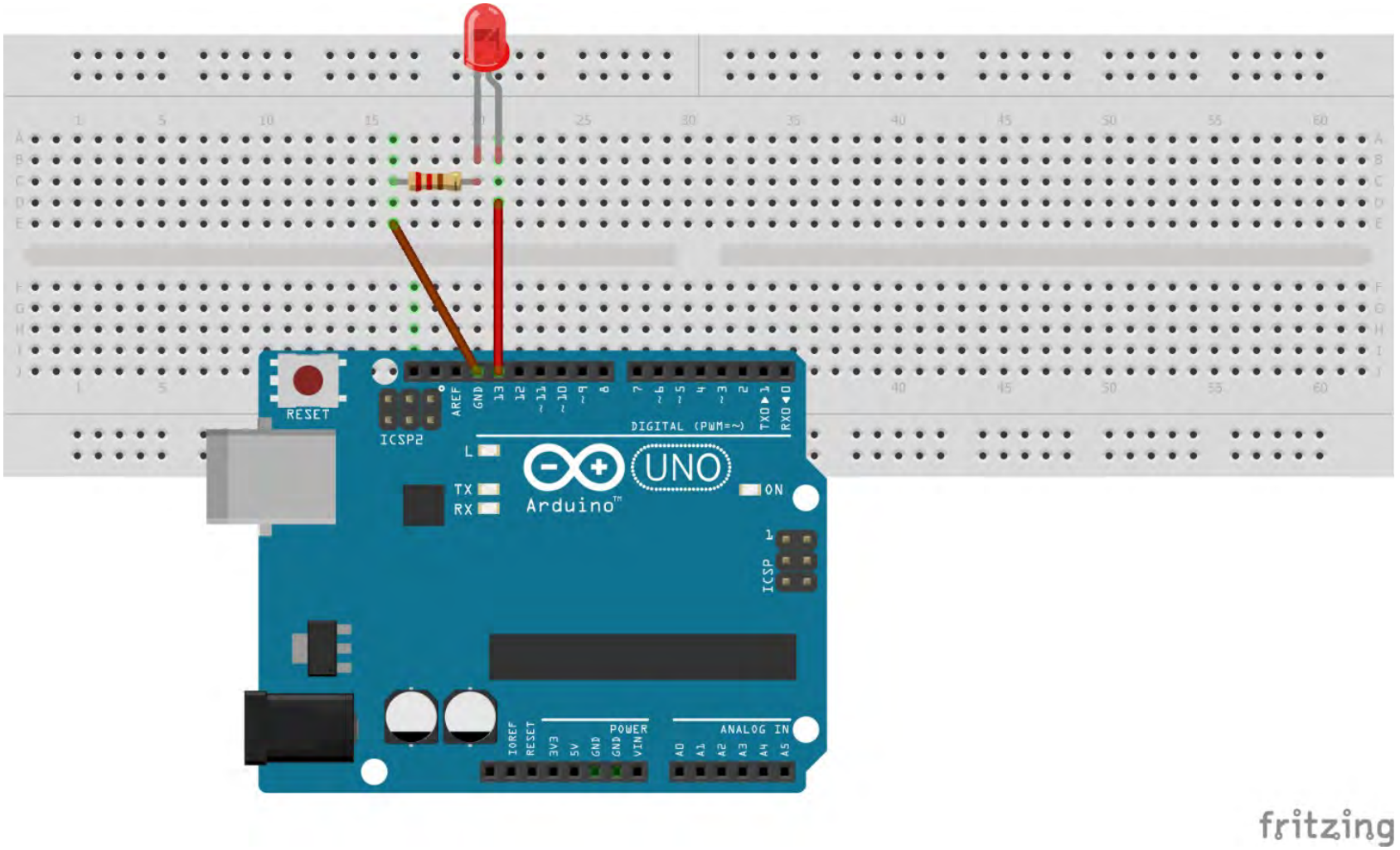
Tinkercad

https://www.tinkercad.com/



AUTODESK®
TINKERCAD®


Simulación
Montaje


RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM=~)
L
TX
RX
Arduino™
UNO
ON
ICSP
POWER
ANALOG IN
IOREF
RESET
3V3
5V
GND
GND
VIN
A0
A1
A2
A3
A4
A5

fritzing

Reto 1



Conectar 1 LED

Cambiar el tiempo en que prende y apaga el LED para una lámpara.

Inspiración



Reto 2

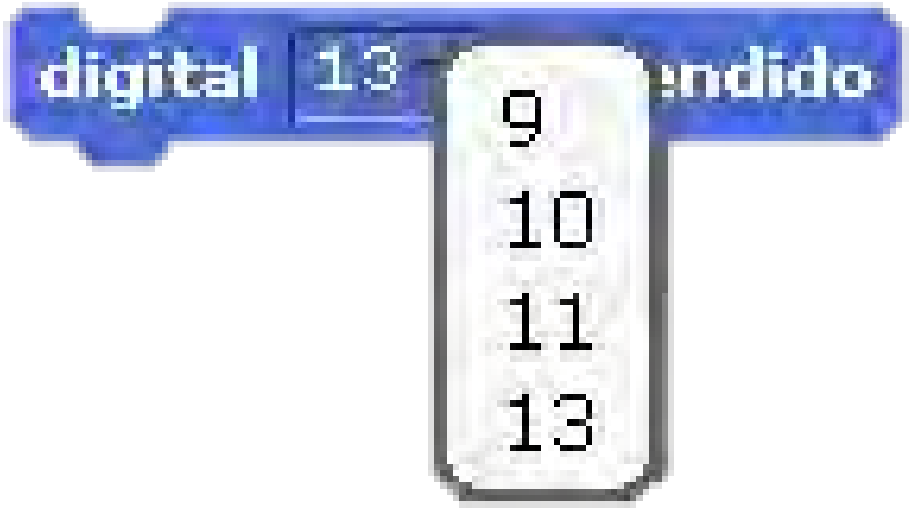


Conectar 3 LEDs

Intermitentes al tiempo

LED1	LED2	LED3
ON	ON	ON
OFF	OFF	OFF
ON	ON	ON
OFF	OFF	OFF

Los LEDs solo se pueden poner como digitales en las salidas que menciona el bloque de “digital”



Reto 3



Conectar 3 LEDs

Encender después del otro

LED1	LED2	LED3
ON	ON	ON
ON	OFF	OFF

Reto 4

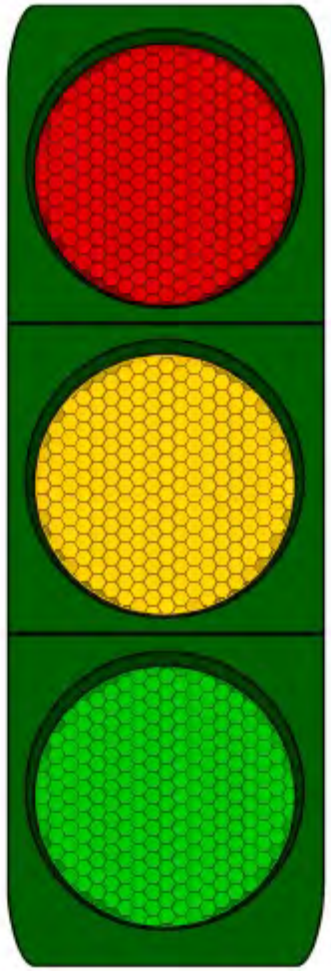


1.semáforo peatonal





1.semáforo vehicular



Salida intermitente con visualización de estado





servo

conexión servo:

café: GND
rojo: +
naranja: señal (en este caso pin 3 y 5)

Tipos de servo



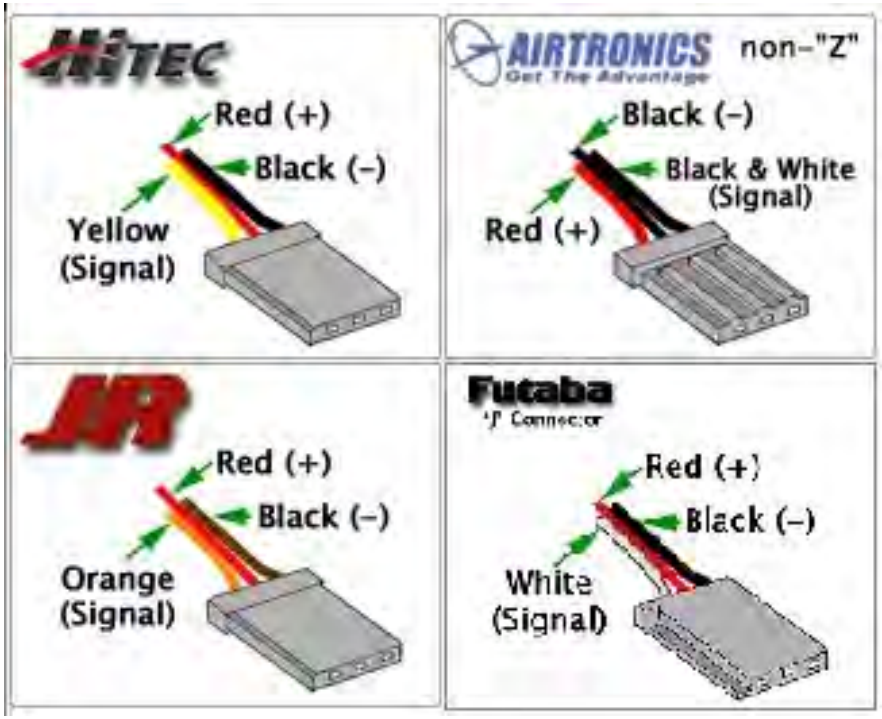




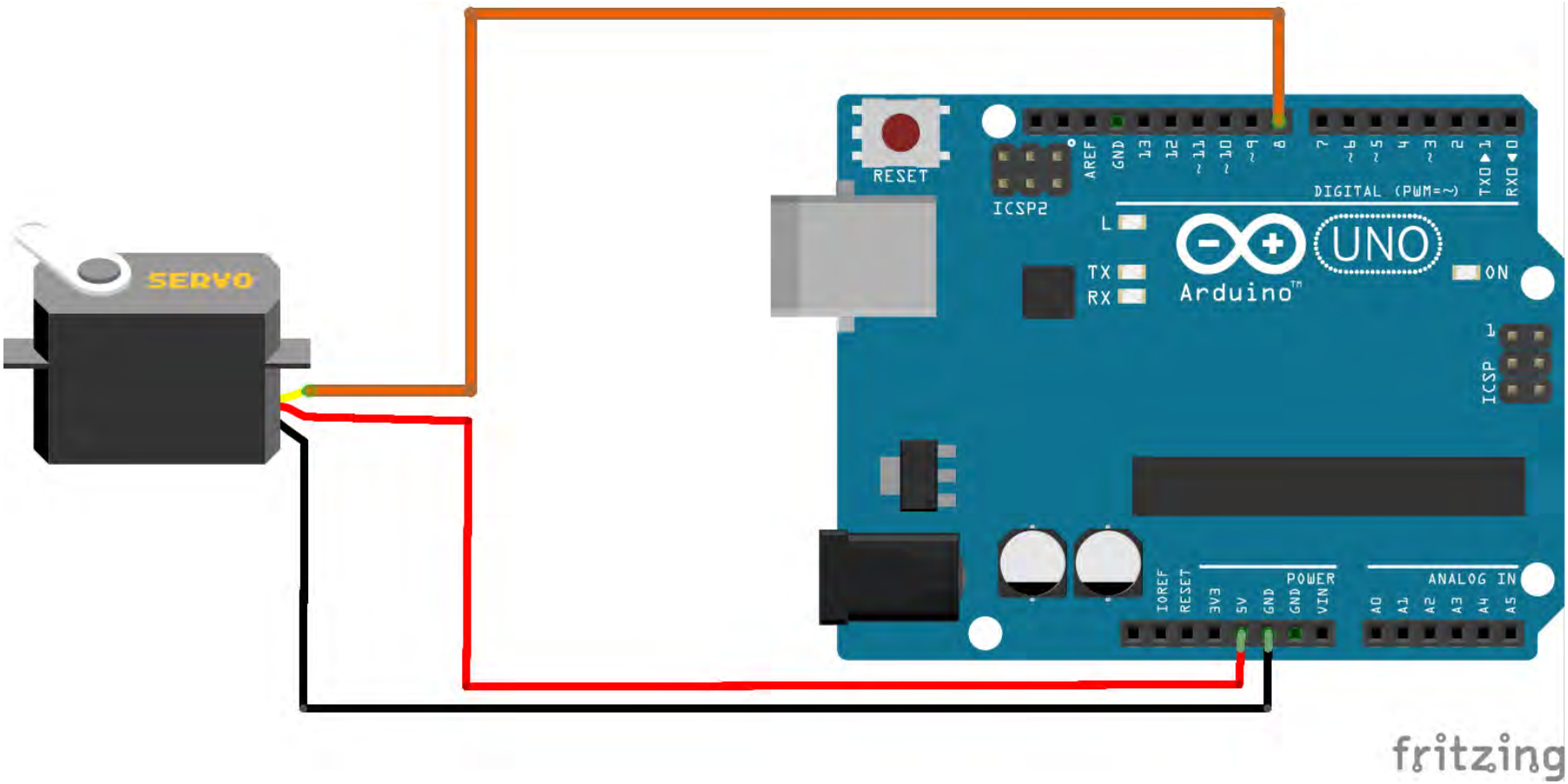
Como funciona un servo

https://www.youtube.com/watch?v=84mxq41zdwE

Tipos de conexiones de un servo



Conexión de servo a arduino




al presionar tecla espacio
por siempre
motor 8 ángulo 0
esperar 1 segundos
motor 8 ángulo 180
esperar 1 segundos

Archivo / Ejemplo / servo / sweep

```
/* Sweep
 by BARRAGAN <http://barraganstudio.com>
 This example code is in the public domain.

 modified 8 Nov 2013
 by Scott Fitzgerald
 https://www.arduino.cc/en/Tutorial/LibraryExamples/Sweep
*/

#include <Servo.h>

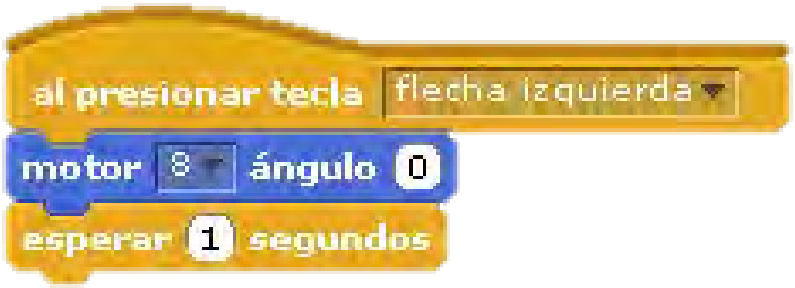
Servo myservo;  // create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position

void setup() {
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}

void loop() {
  for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees
    // in steps of 1 degree
    myservo.write(pos);              // tell servo to go to position in variable 'pos'
    delay(15);                       // waits 15 ms for the servo to reach the position
  }
  for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees
    myservo.write(pos);              // tell servo to go to position in variable 'pos'
    delay(15);                       // waits 15 ms for the servo to reach the position
  }
```

Controlando el giro del servo





Reto 5

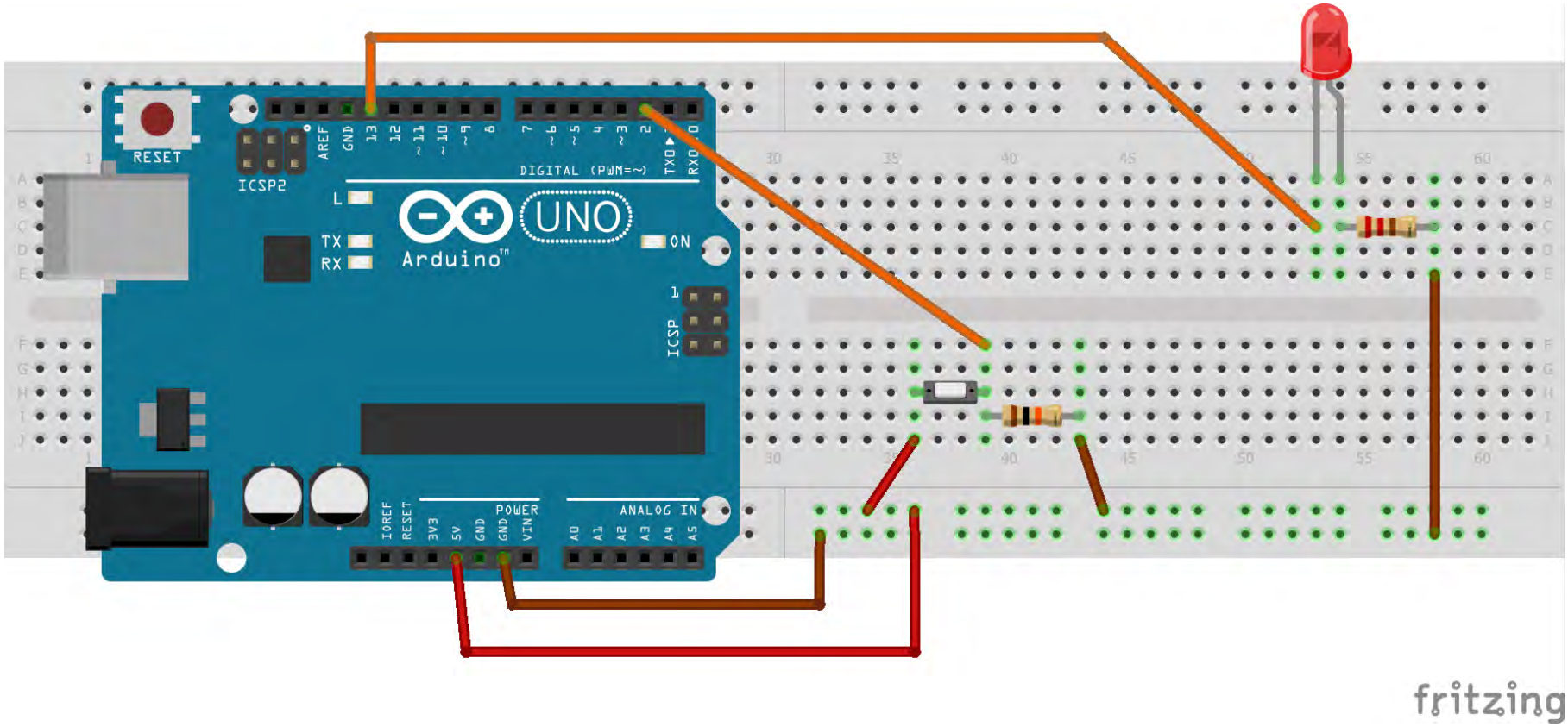


Realiza una mano en cartón o papel y pégala al servomotor

1. Haz que salude tu mano 1 vez
2. Haz que salude tu mano 5 veces
3. Haz que cuando salude 10 veces y cada 3 veces encienda un led

Led con pulsador




RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TXD
RXD
DIGITAL (PWM=~)
L
TX
RX
UNO
Arduino™
ON
ICSP
POWER
ANALOG IN
IOREF
RESET
3V3
5V
GND
GND
VIN
A0
A1
A2
A3
A4
A5
fritzing


al presionar
por siempre
si
¿sensor Digital2 presionado?
digital 13 encendido
si no
digital 13 apagado

```
// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}

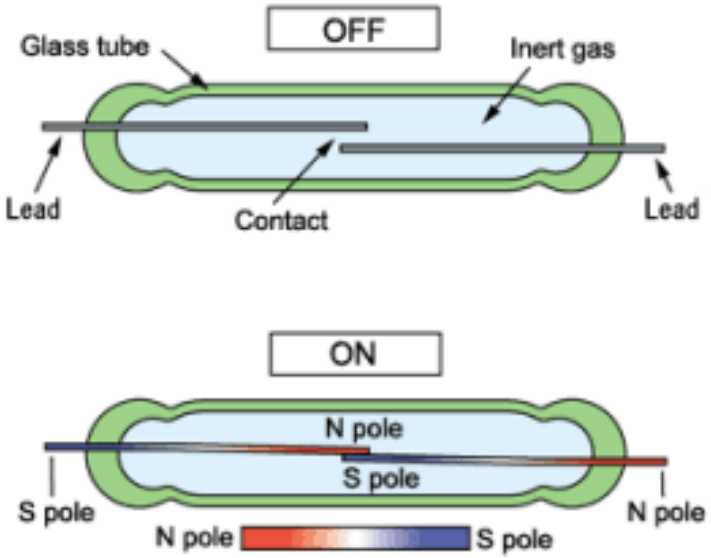
void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

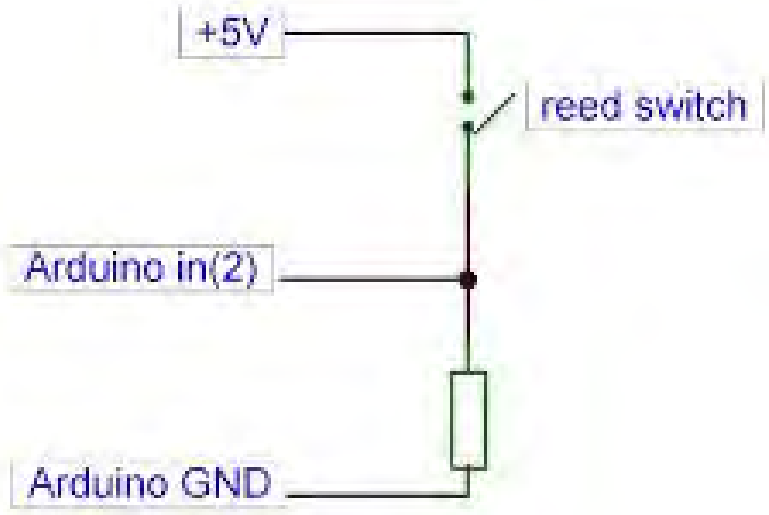
  // check if the pushbutton is pressed. If it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn LED on:
    digitalWrite(ledPin, HIGH);
  } else {
    // turn LED off:
    digitalWrite(ledPin, LOW);
  }
}
```

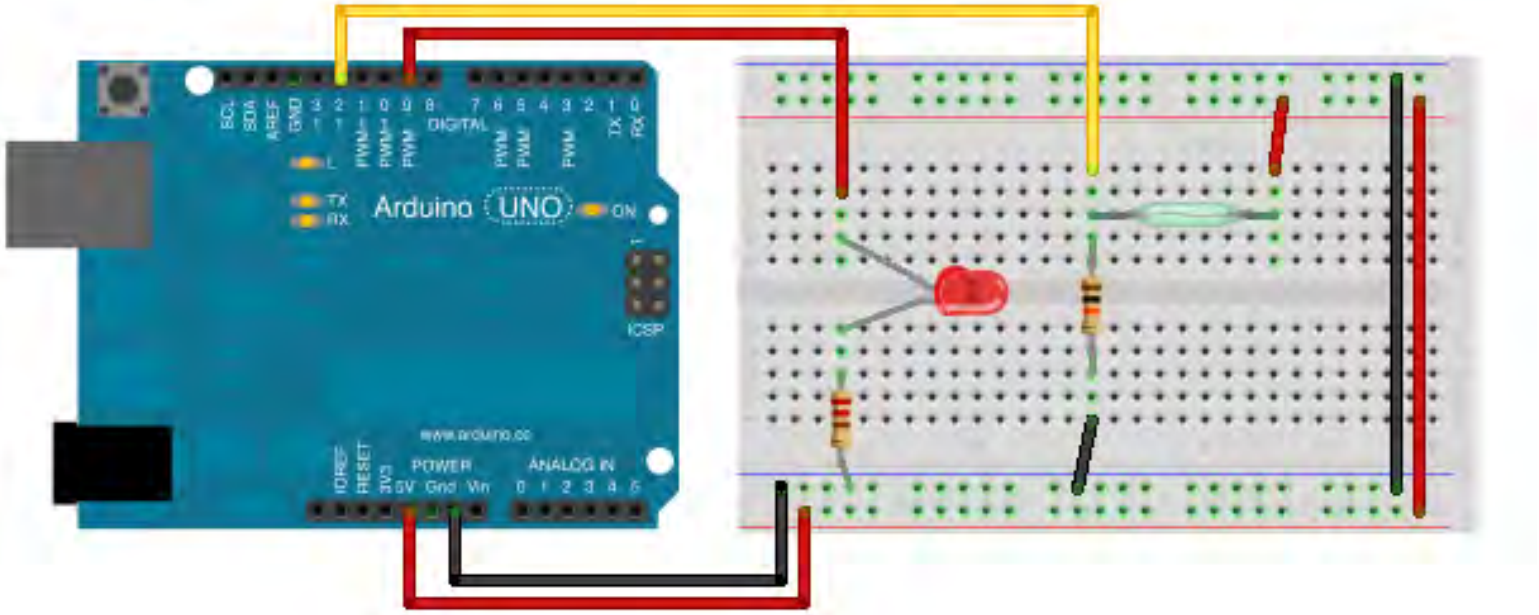
Archivo/Ejemplo/Digital/Button

reed switch




OFF
Glass tube
Inert gas
Lead
Contact
Lead
ON
N pole
S pole
S pole
N pole
N pole
S pole


+5V
reed switch
Arduino in(2)
Arduino GND


SCL
SDA
AREF
GND
DIGITAL
PWM
TX
RX
L
TX
RX
Arduino
UNO
ON
ICSP
www.arduino.cc
IOREF
RESET
3V3
5V
Gnd
Vin
POWER
ANALOG IN
0 1 2 3 4 5

```
// constants won't change. They're used here to set pin numbers:
const int buttonPin = 2;     // the number of the pushbutton pin
const int ledPin =  13;      // the number of the LED pin

// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
}

void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  // check if the pushbutton is pressed. If it is, the buttonState is HIGH:
  if (buttonState == HIGH) {
    // turn LED on:
    digitalWrite(ledPin, HIGH);
  } else {
    // turn LED off:
    digitalWrite(ledPin, LOW);
  }
}
```

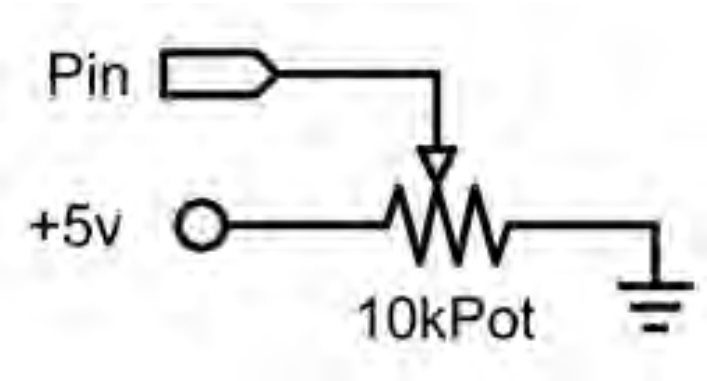
Archivo/Ejemplo/Digital/Button

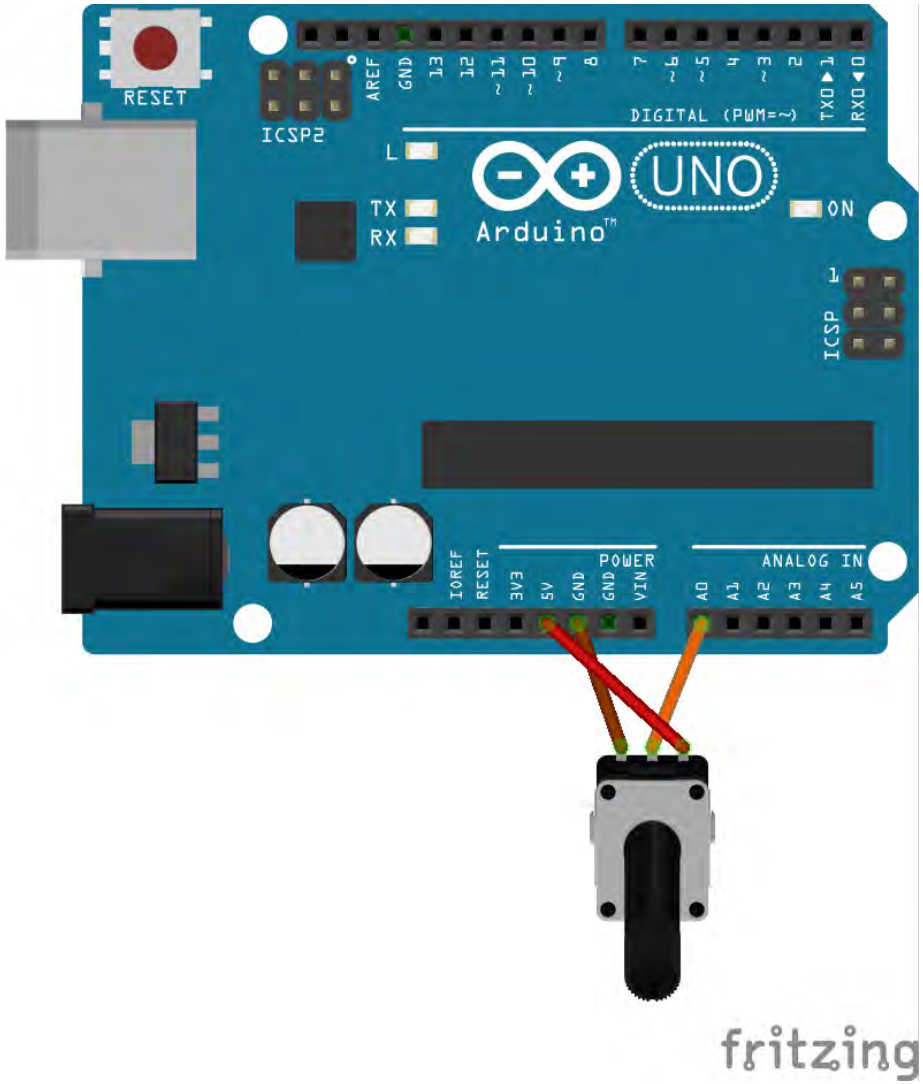
Y editar

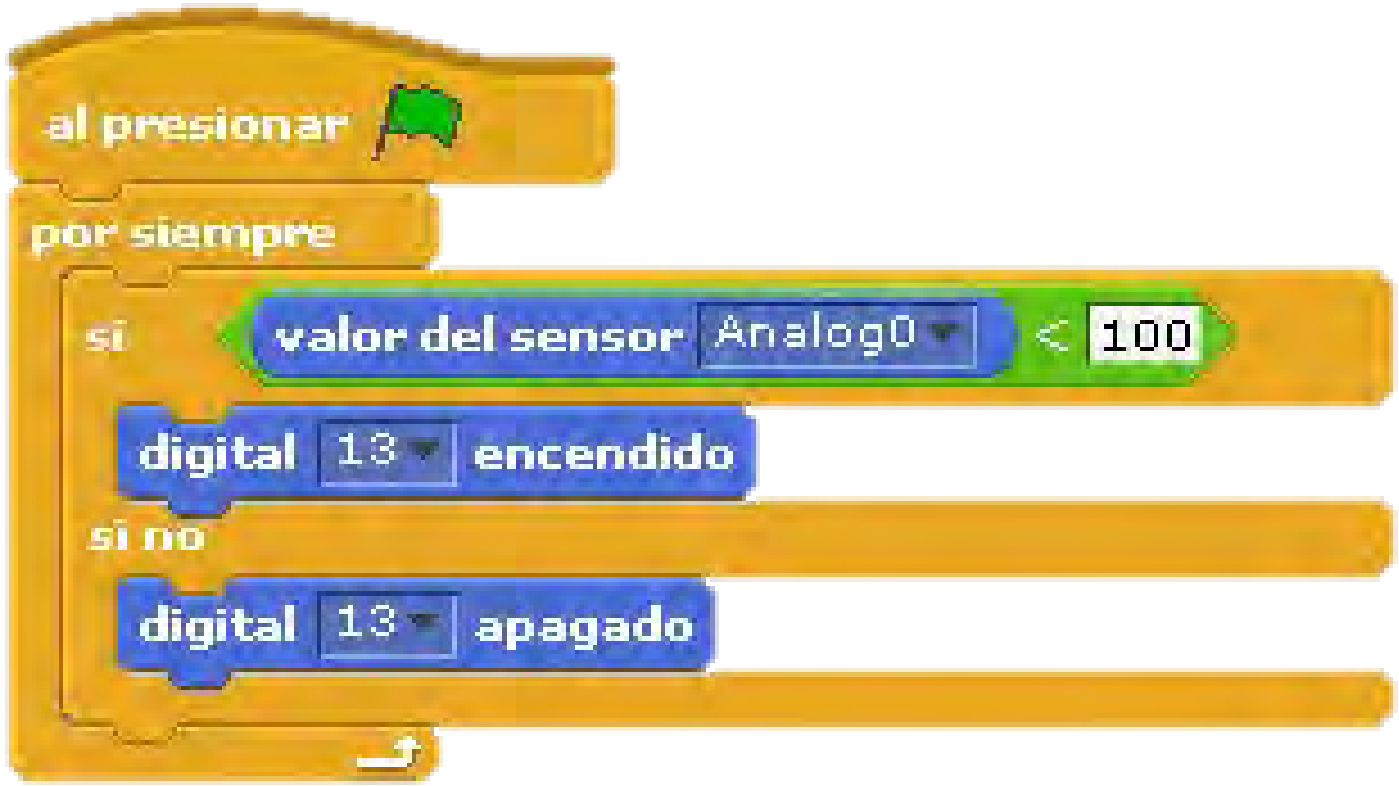
Entrada Análoga

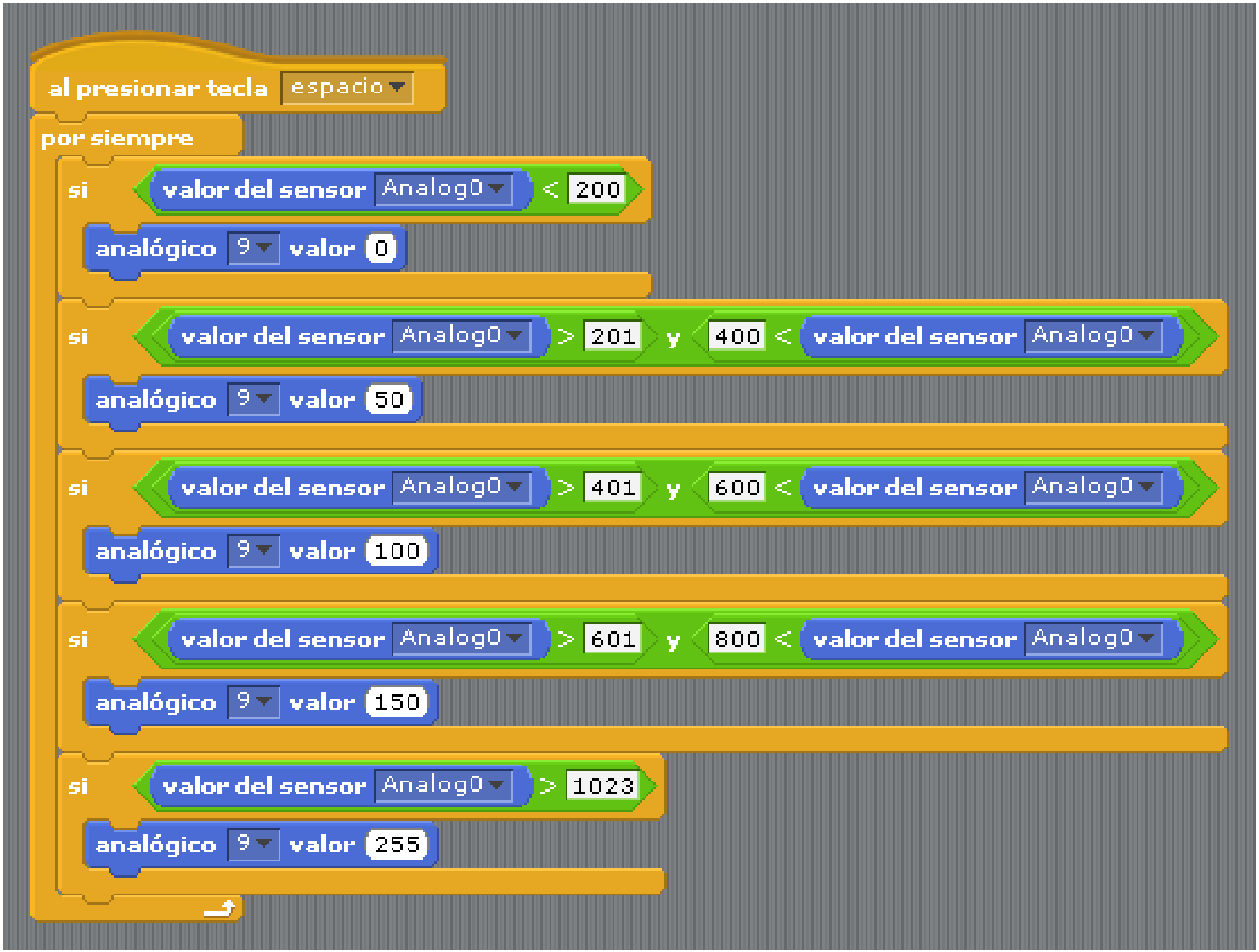


Potenciómetro con arduino



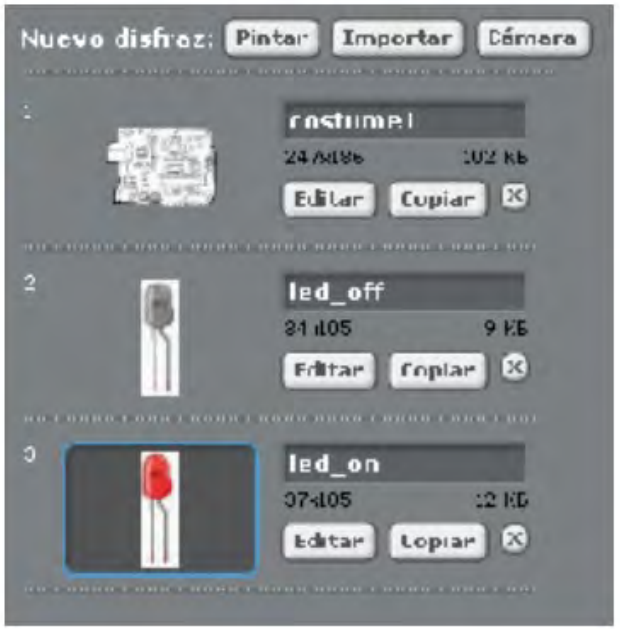

RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM=~)
L
TX
RX
Arduino™
UNO
ON
ICSP
IOREF
RESET
3V3
5V
GND
GND
VIN
POWER
A0
A1
A2
A3
A4
A5
ANALOG IN
fritzing

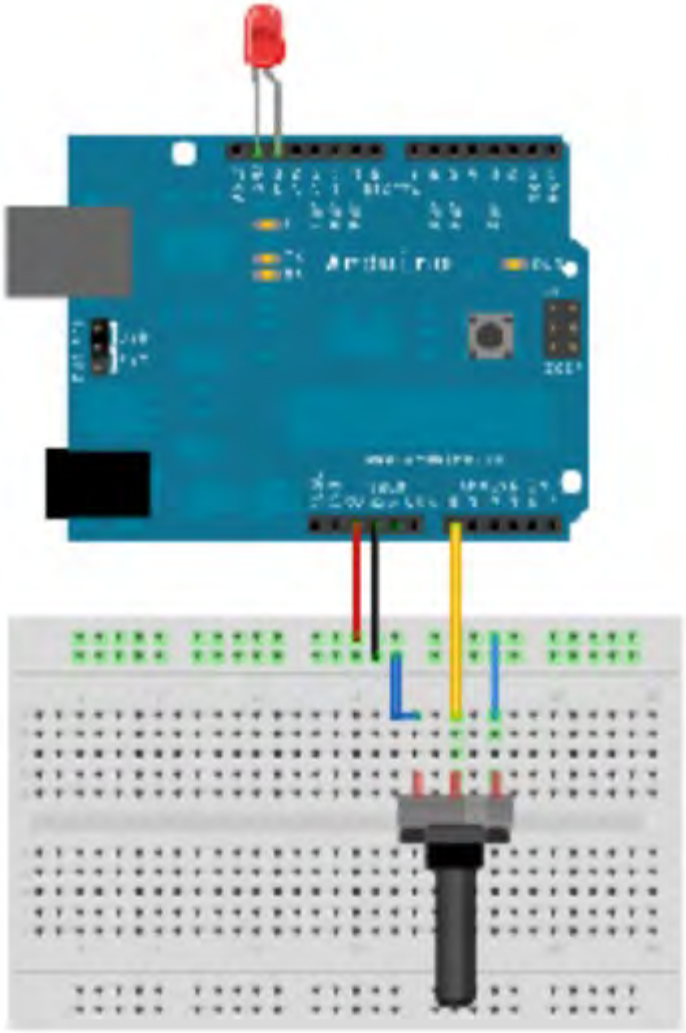

al presionar
por siempre
si
valor del sensor
Analog0
< 100
digital
13
encendido
si no
digital
13
apagado


al presionar tecla espacio
por siempre
si valor del sensor Analog0 < 200
analógico 9 valor 0
si valor del sensor Analog0 > 201 y 400 < valor del sensor Analog0
analógico 9 valor 50
si valor del sensor Analog0 > 401 y 600 < valor del sensor Analog0
analógico 9 valor 100
si valor del sensor Analog0 > 601 y 800 < valor del sensor Analog0
analógico 9 valor 150
si valor del sensor Analog0 > 1023
analógico 9 valor 255

Salida intermitente con visualización de estado en pantalla y control de frecuencia mediante un canal de entrada analógica.







```
int sensorPin = A0;    // select the input pin for the potentiometer
int ledPin = 13;      // select the pin for the LED
int sensorValue = 0;  // variable to store the value coming from the sensor

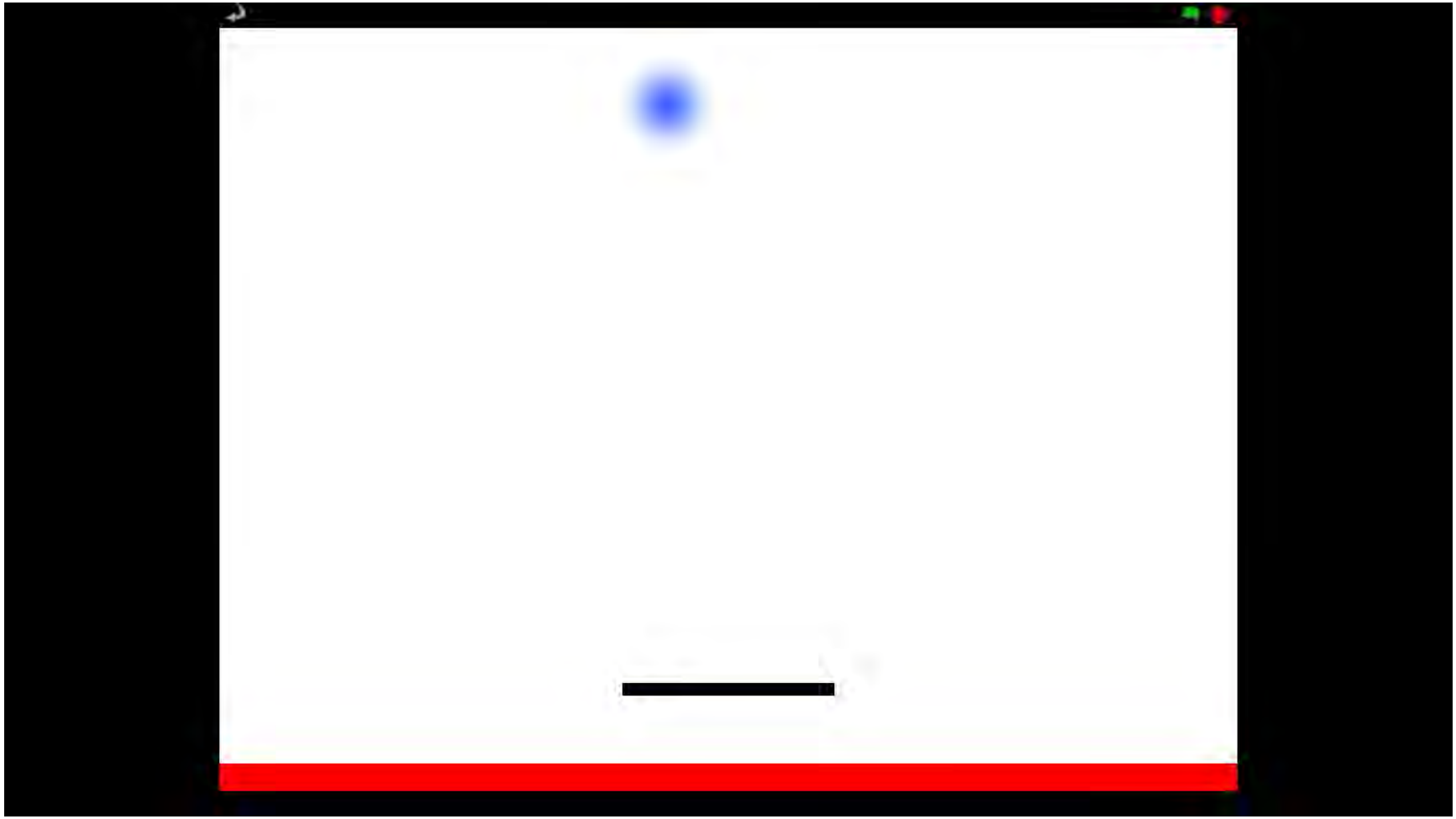
void setup() {
  // declare the ledPin as an OUTPUT:
  pinMode(ledPin, OUTPUT);
}

void loop() {
  // read the value from the sensor:
  sensorValue = analogRead(sensorPin);
  // turn the ledPin on
  digitalWrite(ledPin, HIGH);
  // stop the program for <sensorValue> milliseconds:
  delay(sensorValue);
  // turn the ledPin off:
  digitalWrite(ledPin, LOW);
  // stop the program for for <sensorValue> milliseconds:
  delay(sensorValue);
}
```

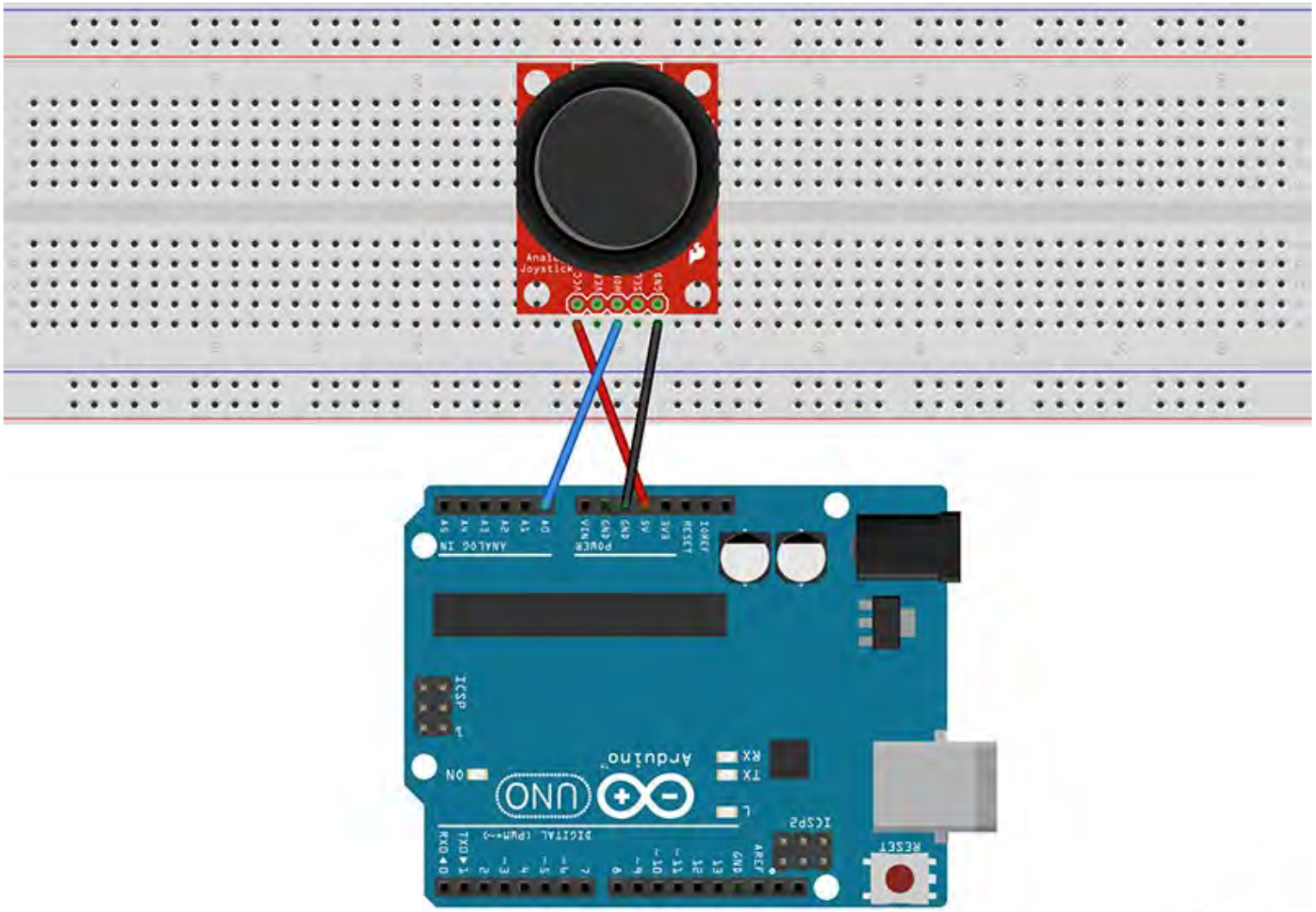
Archivo/Ejemplo/Analog/AnalogInput

Videojuego con potenciómetro

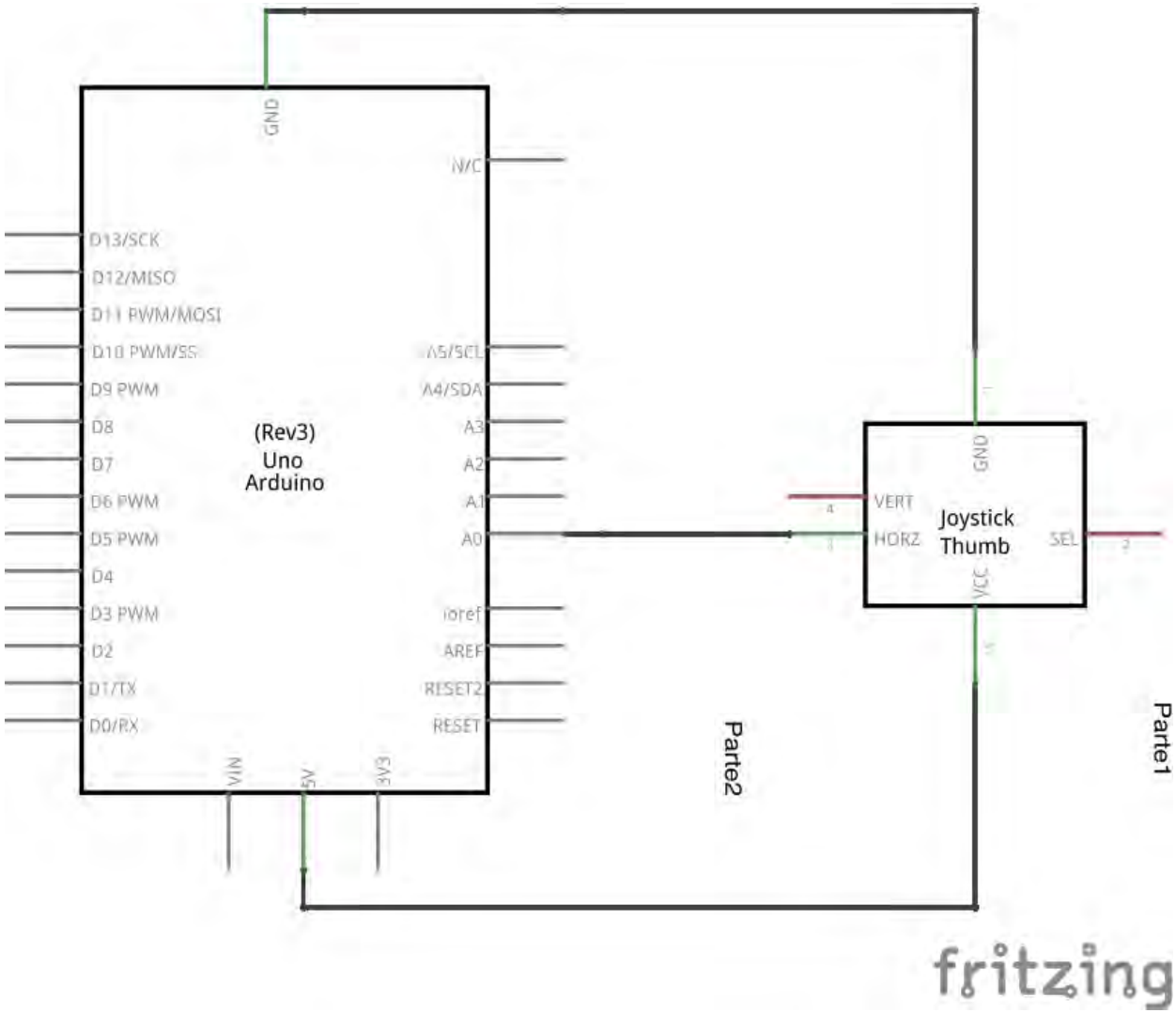


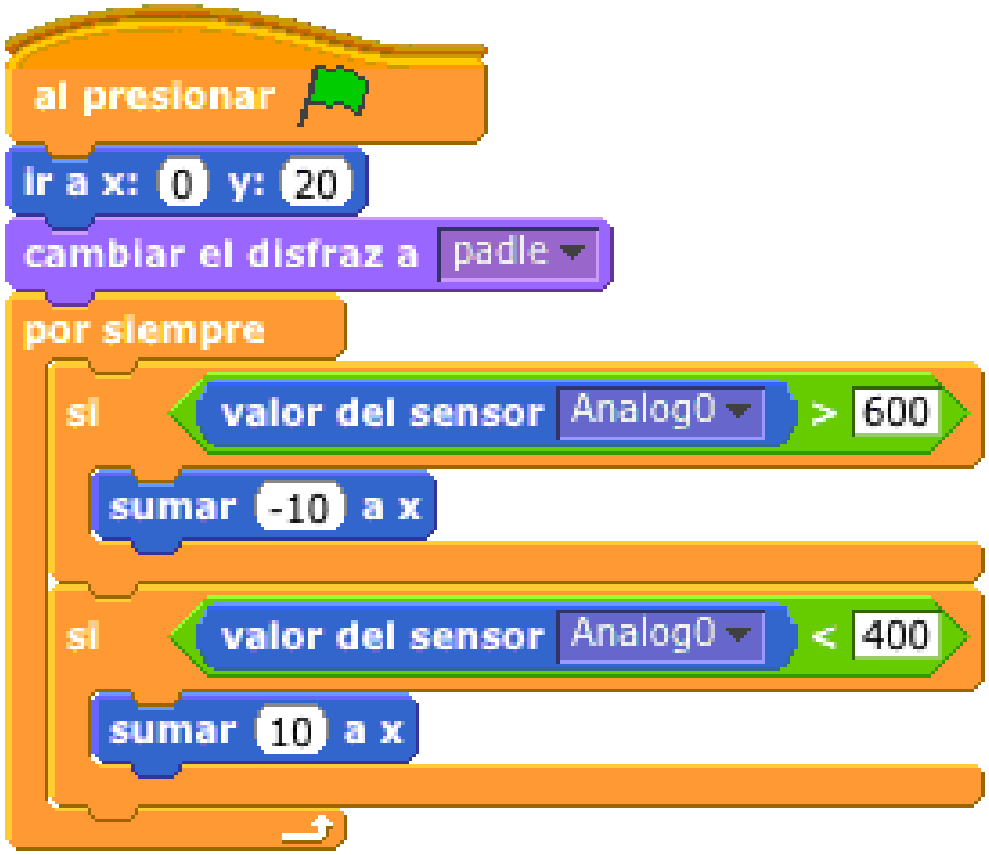


Podemos usar un potenciómetro o un joystick

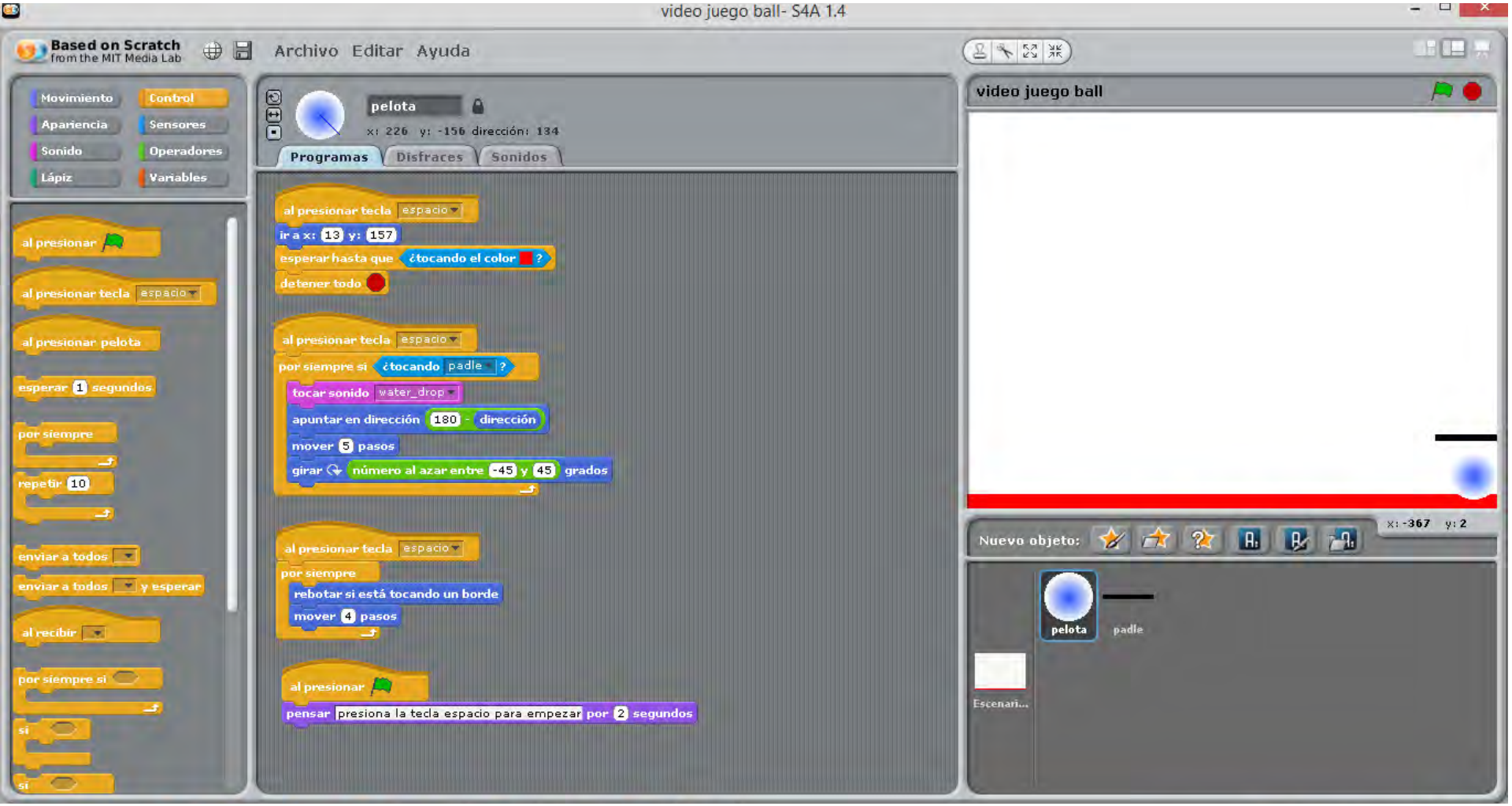

Analog Joystick
VCC
VRx
VRy
SW
GND
ANALOG IN
POWER
Arduino
UNO
ON
TX
RX
L
ICSP
ICSP2
RESET
DIGITAL (PWM~)

fritzing

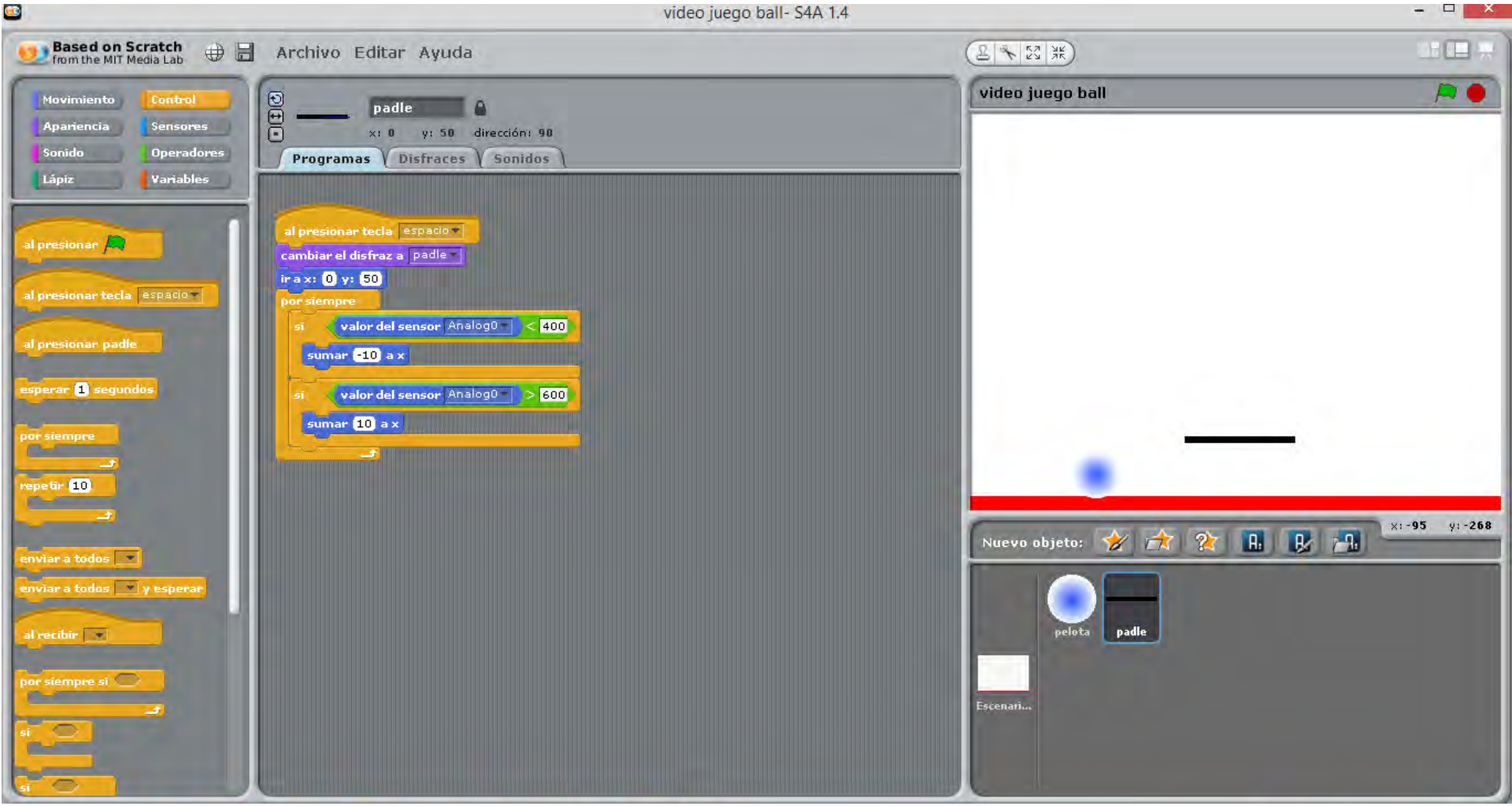

GND
N/C
D13/SCK
D12/MISO
D11 PWM/MOSI
D10 PWM/SS
D9 PWM
D8
D7
D6 PWM
D5 PWM
D4
D3 PWM
D2
D1/TX
D0/RX
A5/SCL
A4/SDA
A3
A2
A1
A0
ioref
AREF
RESET2
RESET
VIN
5V
3V3
(Rev3)
Uno
Arduino
GND
VERT
HORZ
Joystick
Thumb
SEL
VCC
Parte2
Parte1
fritzing


al presionar
ir a x: 0 y: 20
cambiar el disfraz a padle
por siempre
si valor del sensor Analog0 > 600
sumar -10 a x
si valor del sensor Analog0 < 400
sumar 10 a x

Programa de la pelota



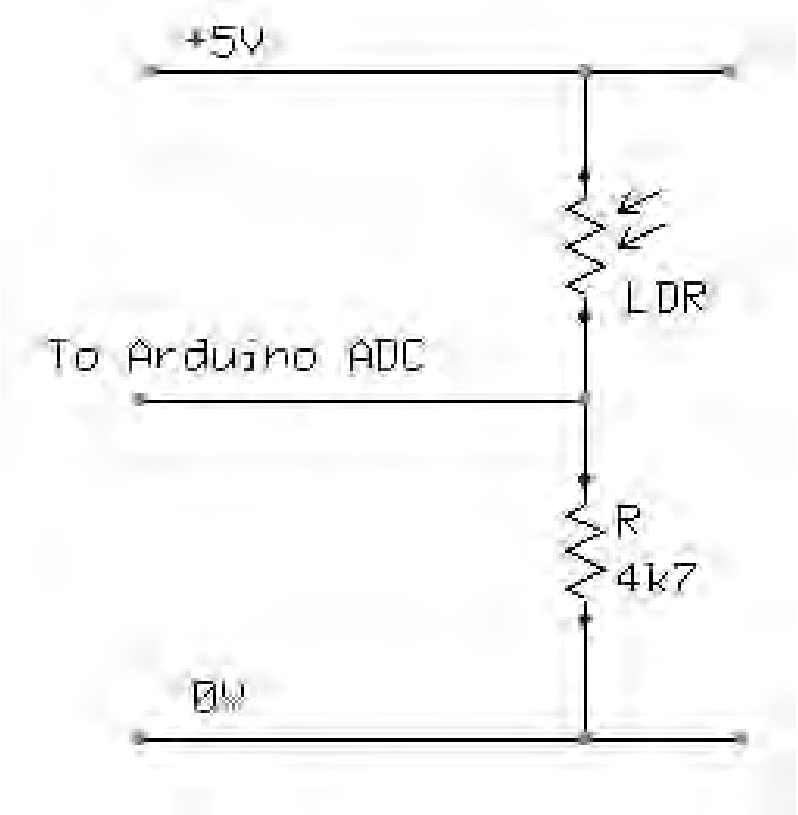
Programa del padle

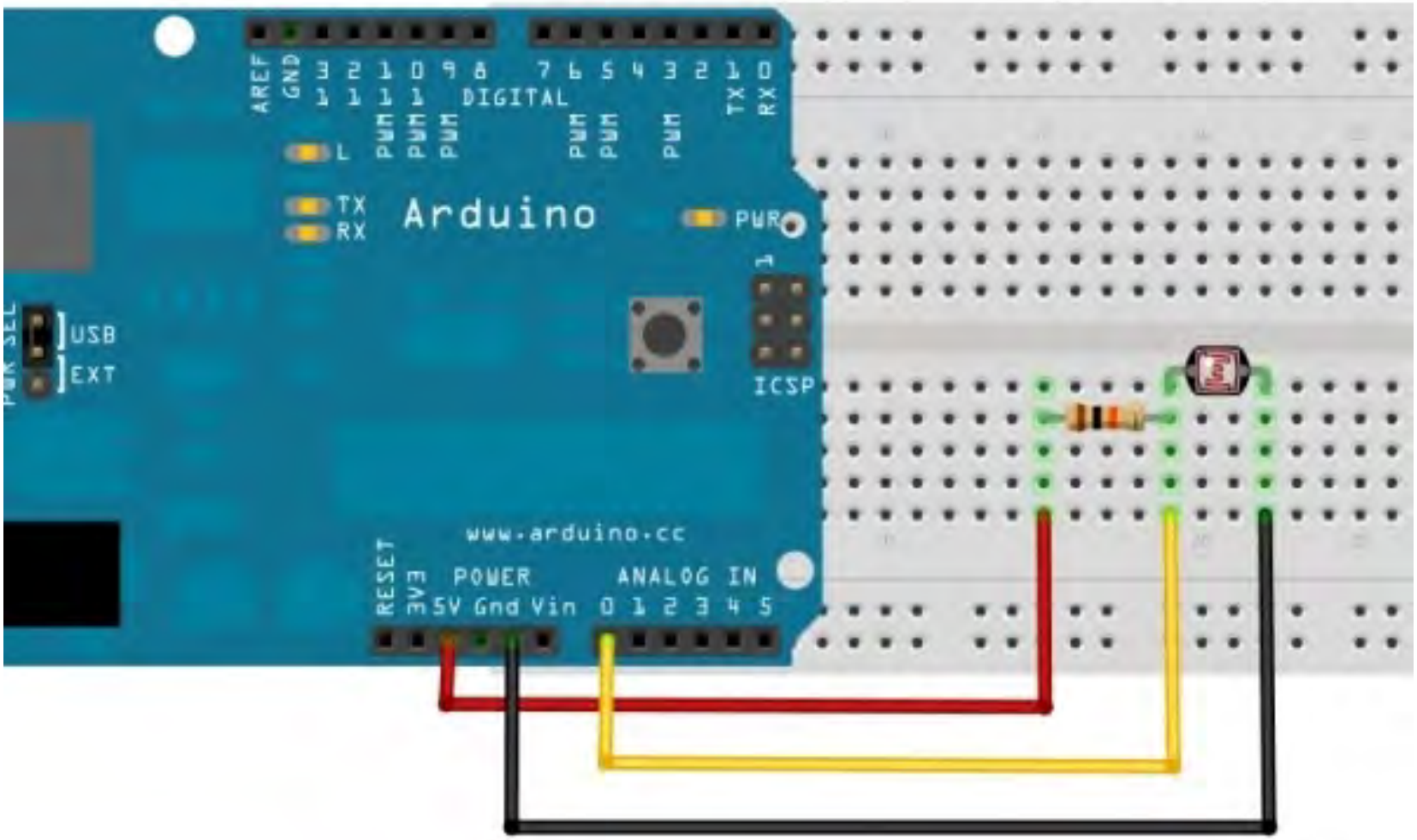


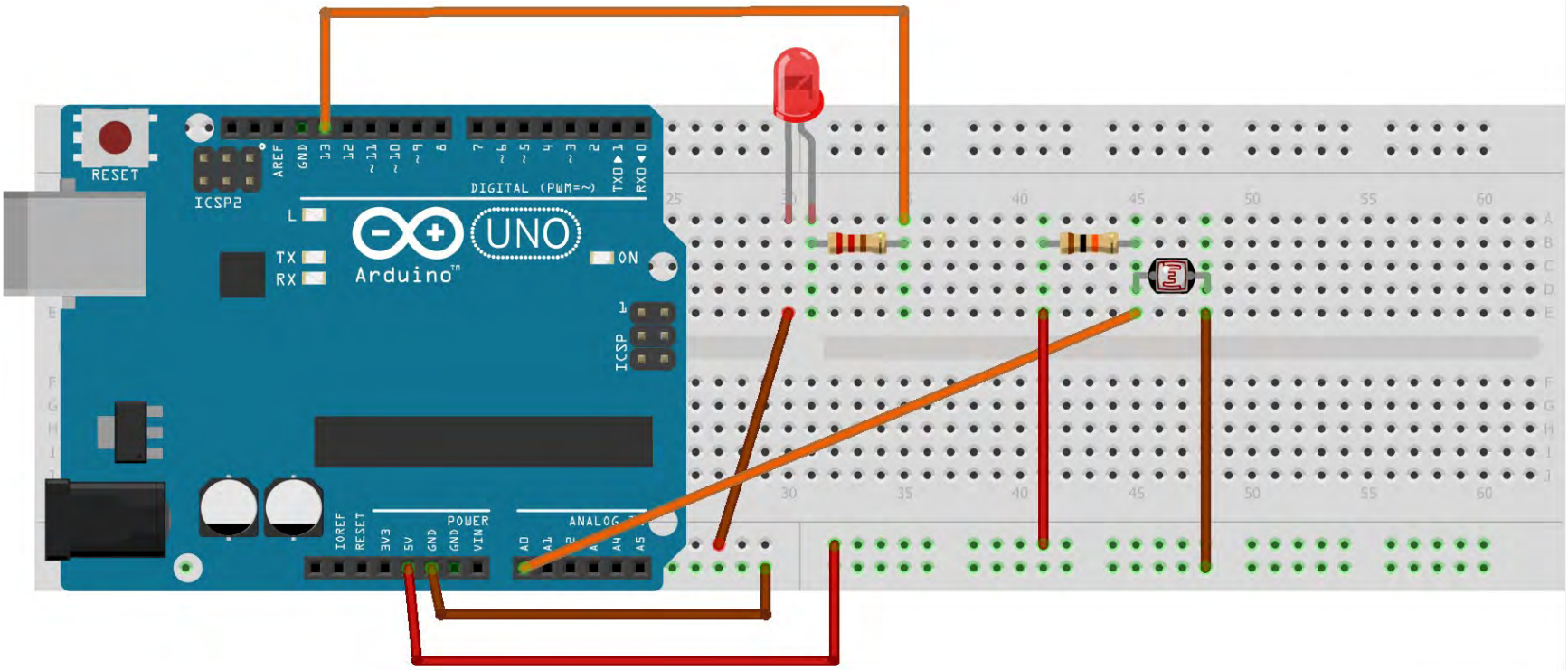
Arduino y LDR

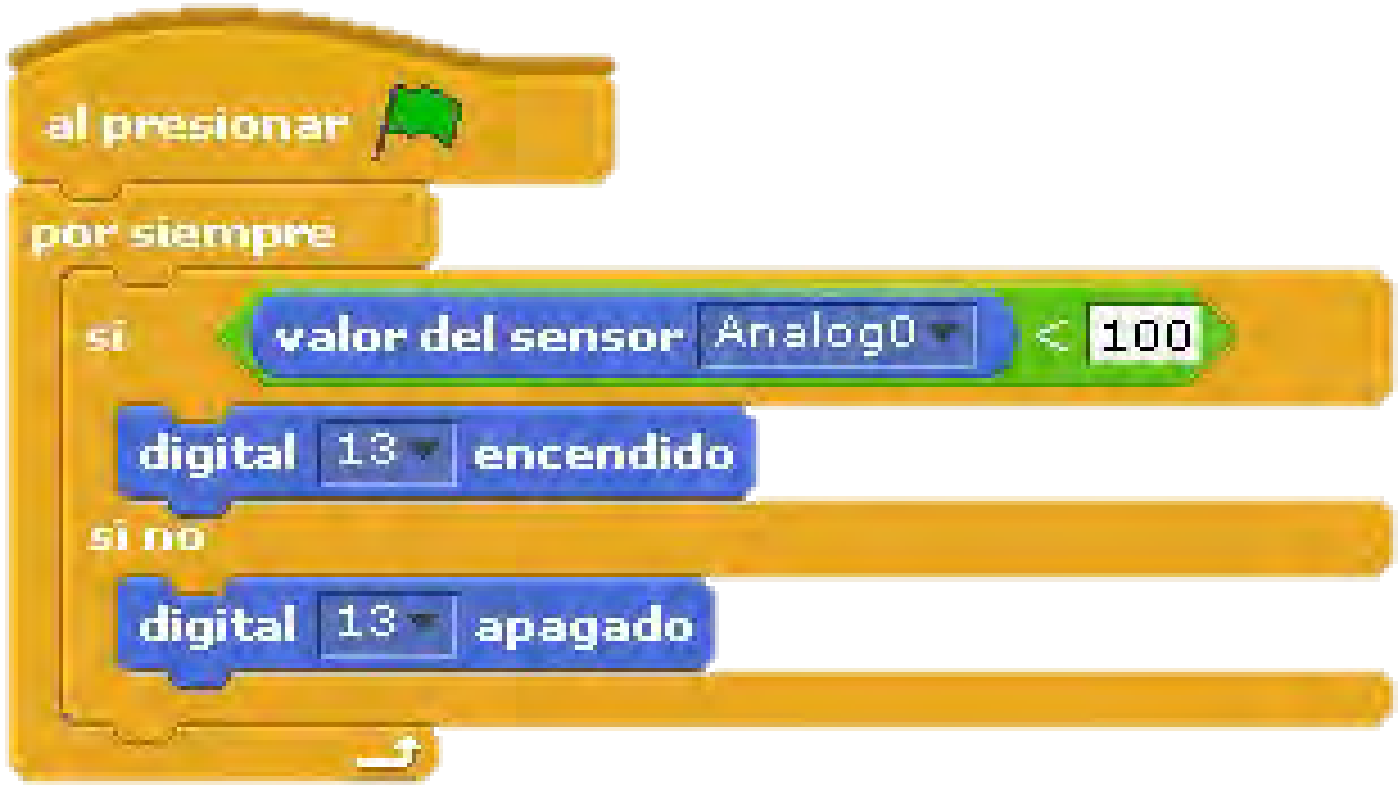


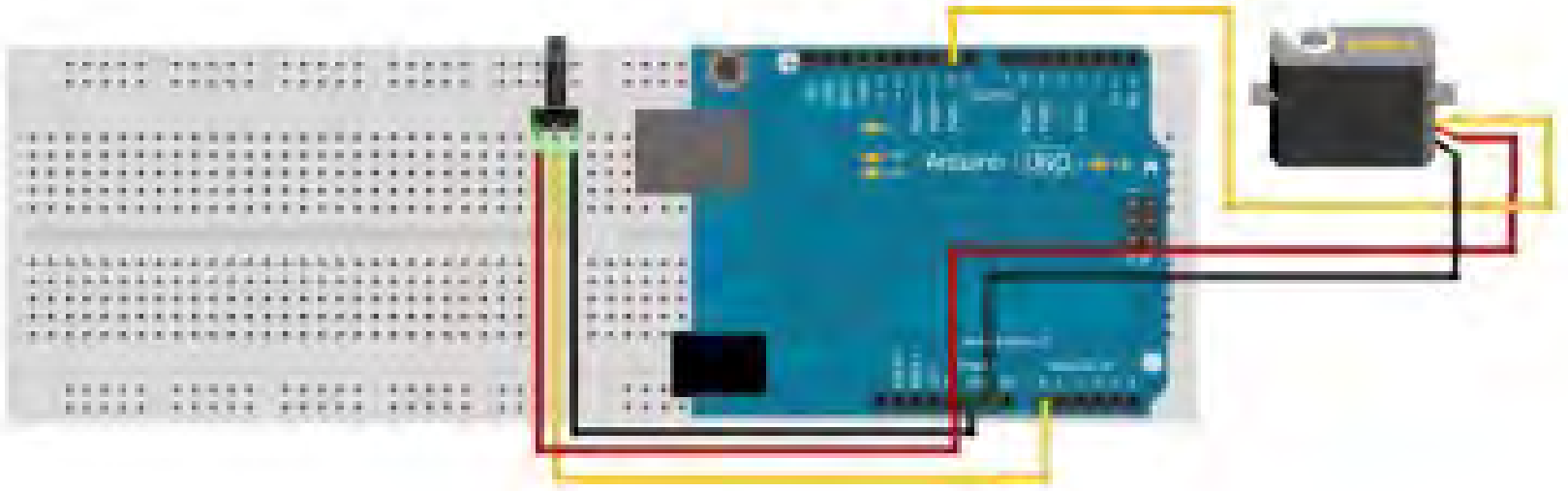
Arduino y LDR




AREF
GND
DIGITAL
TX
RX
L
TX
RX
Arduino
PWR
USB
EXT
ICSP
www.arduino.cc
RESET
3V3
POWER
5V Gnd Vin
ANALOG IN
0 1 2 3 4 5


RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX→1
RX←0
DIGITAL (PWM=~)
L
TX
RX
UNO
Arduino™
ON
ICSP
POWER
IOREF
RESET
3V3
5V
GND
GND
VIN
ANALOG
A0
A1
A2
A3
A4
A5


al presionar
por siempre
si
valor del sensor
Analog0
< 100
digital
13
encendido
si no
digital
13
apagado



Servo con potenciómetro


al presionar
por siempre
fijar señal a valor del sensor Analog0
fijar giro a redondear señal * 180 / 1023
motor 8 ángulo giro

Archivo / Ejemplo / servo / knob

```
#include <Servo.h>

Servo myservo;  // create servo object to control a servo

int potpin = A0;  // analog pin used to connect the potentiometer
int val;    // variable to read the value from the analog pin

void setup() {
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}

void loop() {
  val = analogRead(potpin);            // reads the value of the potentiometer (value
between 0 and 1023)
  val = map(val, 0, 1023, 0, 180);     // scale it for use with the servo (value
between 0 and 180)
  myservo.write(val);                  // sets the servo position according to the scaled
value
  delay(15);                           // waits for the servo to get there
}
```

Reto 6

Mínimo 2 personas y máximo 3.



En un parqueadero de un centro comercial. El brazo de la puerta realiza un giro de 90 grados para abrir y se devuelve para cerrar. Al abrir la puerta se enciende una bombilla verde y al cerrar una roja. Y en la noche deben prender las luces dentro del parqueadero.

Makey makey con arduino



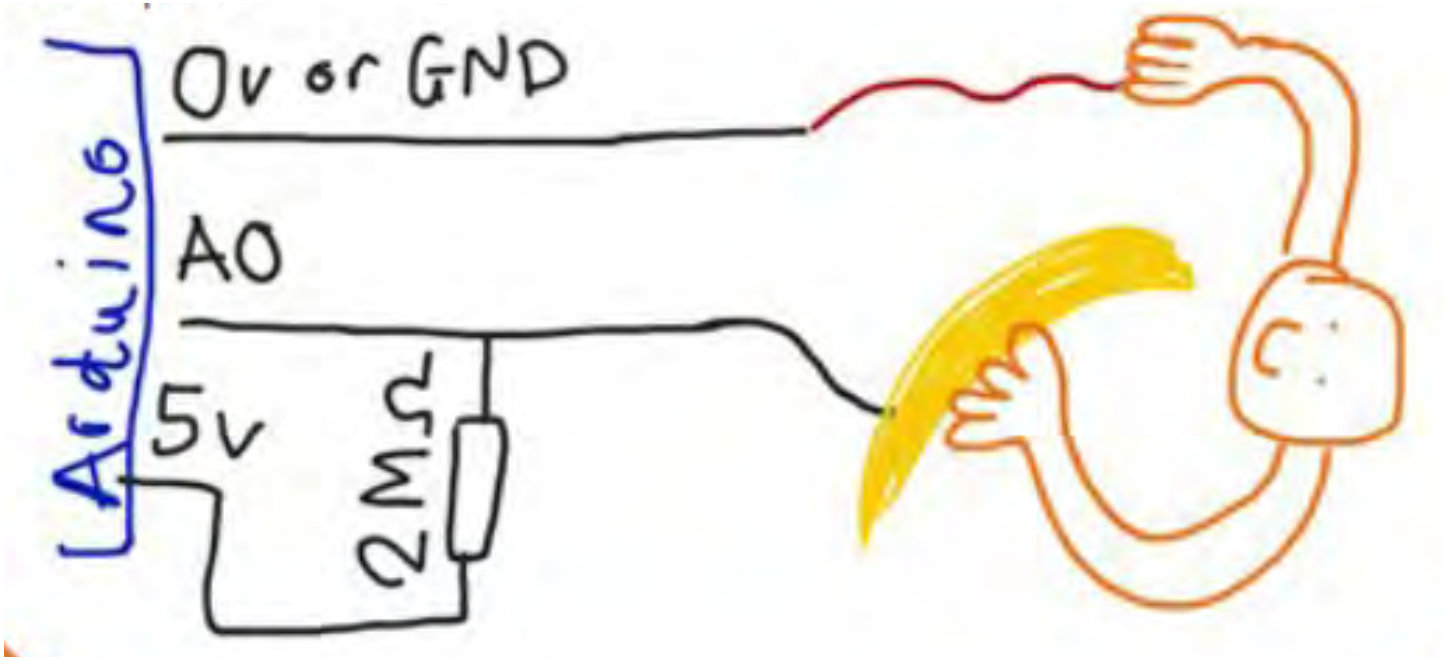
¿Haz visto un makey makey?

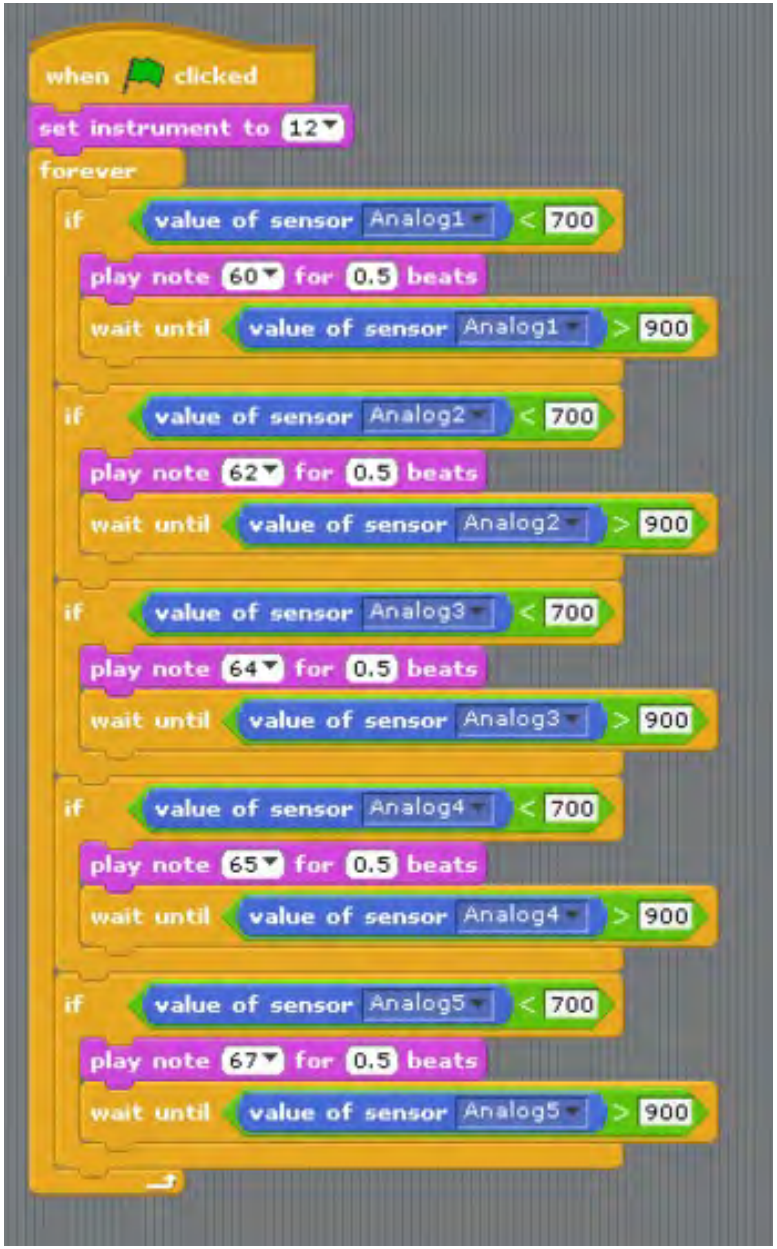
http://makeymakey.com/

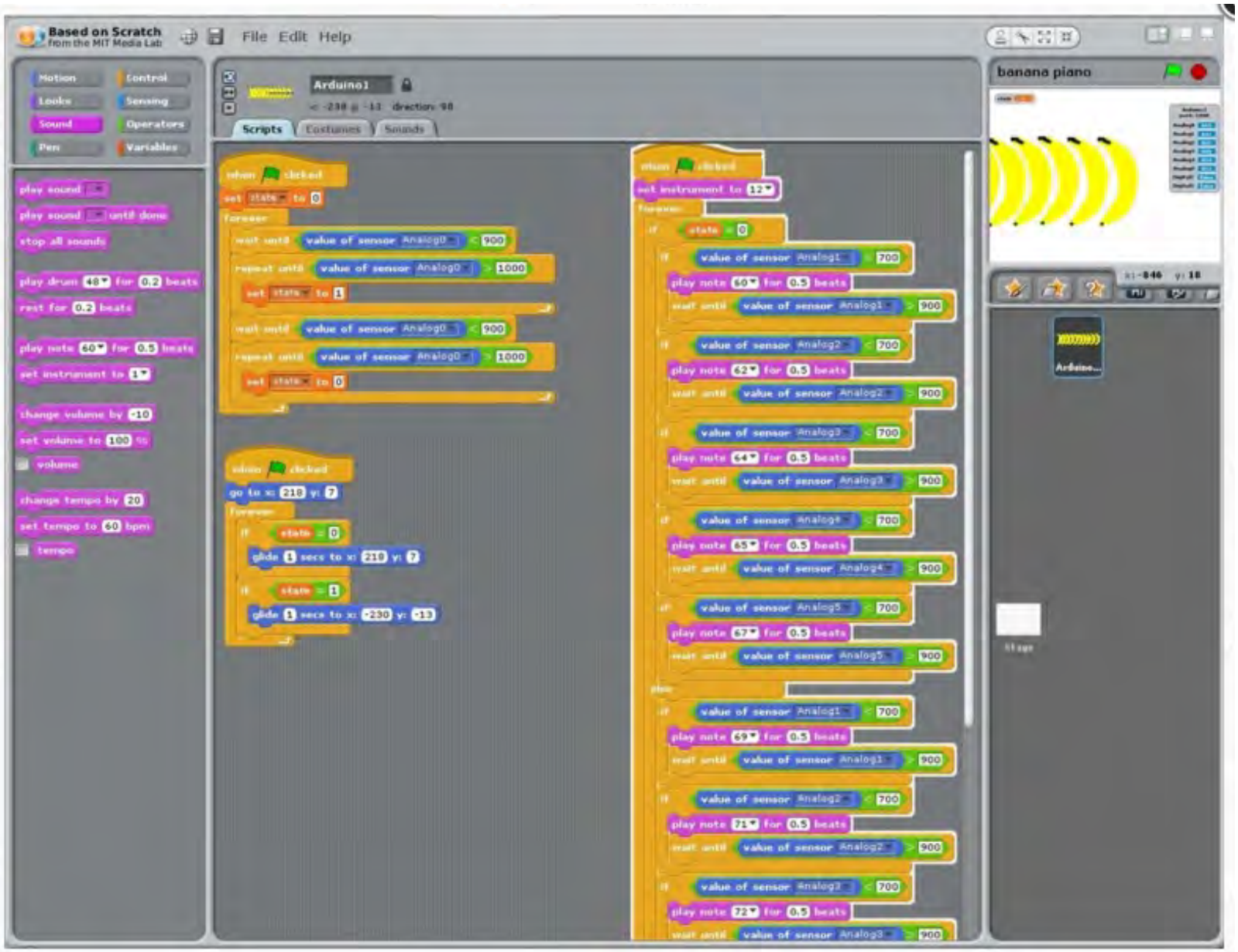


Makey makey y frutas para interactuar con el videojuego

Vamos a realizarlo con Arduino


Arduino
0v or GND
A0
5v
2MΩ


when clicked
set instrument to 12
forever
if value of sensor Analog1 < 700
play note 60 for 0.5 beats
wait until value of sensor Analog1 > 900
if value of sensor Analog2 < 700
play note 62 for 0.5 beats
wait until value of sensor Analog2 > 900
if value of sensor Analog3 < 700
play note 64 for 0.5 beats
wait until value of sensor Analog3 > 900
if value of sensor Analog4 < 700
play note 65 for 0.5 beats
wait until value of sensor Analog4 > 900
if value of sensor Analog5 < 700
play note 67 for 0.5 beats
wait until value of sensor Analog5 > 900


Based on Scratch
From the MIT Media Lab
File Edit Help
Motion
Control
Looks
Sensing
Sound
Operators
Pen
Variables
Arduino1
Scripts
Costumes
Sounds
banana piano
when clicked
set state to 0
forever
wait until value of sensor Analog0 < 900
repeat until value of sensor Analog0 > 1000
set state to 1
wait until value of sensor Analog0 < 900
repeat until value of sensor Analog0 > 1000
set state to 0
when clicked
go to x: 218 y: 7
forever
if state = 0
glide 1 secs to x: 218 y: 7
if state = 1
glide 1 secs to x: -230 y: -13
when clicked
set instrument to 12
forever
if state = 0
if value of sensor Analog1 < 700
play note 60 for 0.5 beats
wait until value of sensor Analog1 > 900
if value of sensor Analog2 < 700
play note 62 for 0.5 beats
wait until value of sensor Analog2 > 900
if value of sensor Analog3 < 700
play note 64 for 0.5 beats
wait until value of sensor Analog3 > 900
if value of sensor Analog4 < 700
play note 65 for 0.5 beats
wait until value of sensor Analog4 > 900
if value of sensor Analog5 < 700
play note 67 for 0.5 beats
wait until value of sensor Analog5 > 900
else
if value of sensor Analog1 < 700
play note 69 for 0.5 beats
wait until value of sensor Analog1 > 900
if value of sensor Analog2 < 700
play note 71 for 0.5 beats
wait until value of sensor Analog2 > 900
if value of sensor Analog3 < 700
play note 72 for 0.5 beats
wait until value of sensor Analog3 > 900

Inspiración

Veamos el video

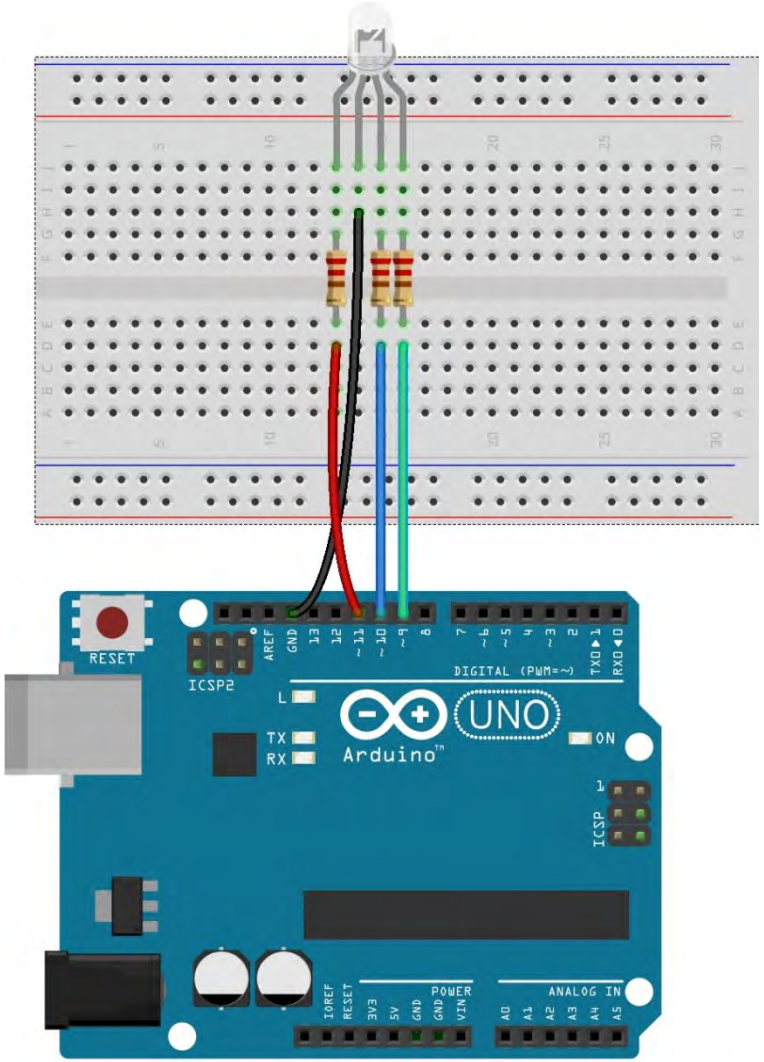
Y ¿Qué se te ocurre hacer?

https://www.youtube.com/watch?v=YO2rCrmur6g

LED RGB



LED RGB


RESET
ICSP2
AREF
GND
13
12
11
10
9
8
7
6
5
4
3
2
TXD
RXD
DIGITAL (PWM=~)
L
TX
RX
Arduino
UNO
ON
ICSP
POWER
IOREF
RESET
3V3
5V
GND
GND
VIN
ANALOG IN
A0
A1
A2
A3
A4
A5

Robótica Industrial



Conecta dos servomotores

