

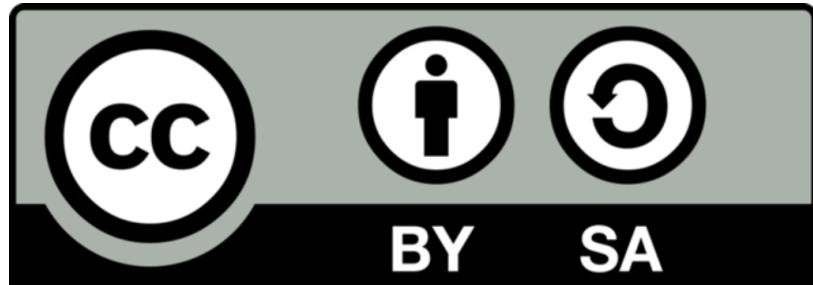
Arduino: Soluciones en Robótica y Domótica

ETSIIT

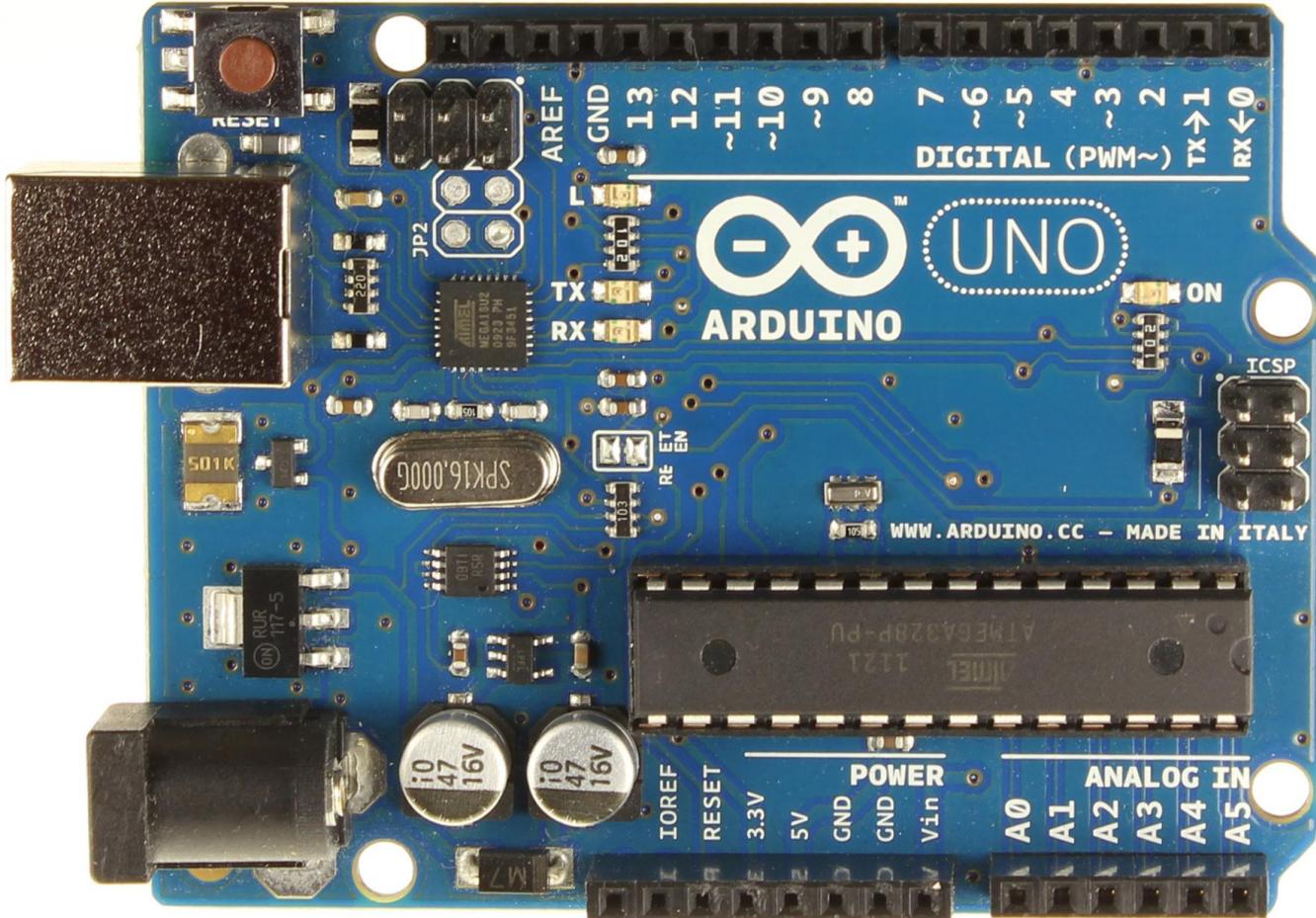
15.04.2015



elcacharreo.com



Introducción a Arduino: Presente



Introducción a Arduino: Presente



José Antonio Vacas Martínez



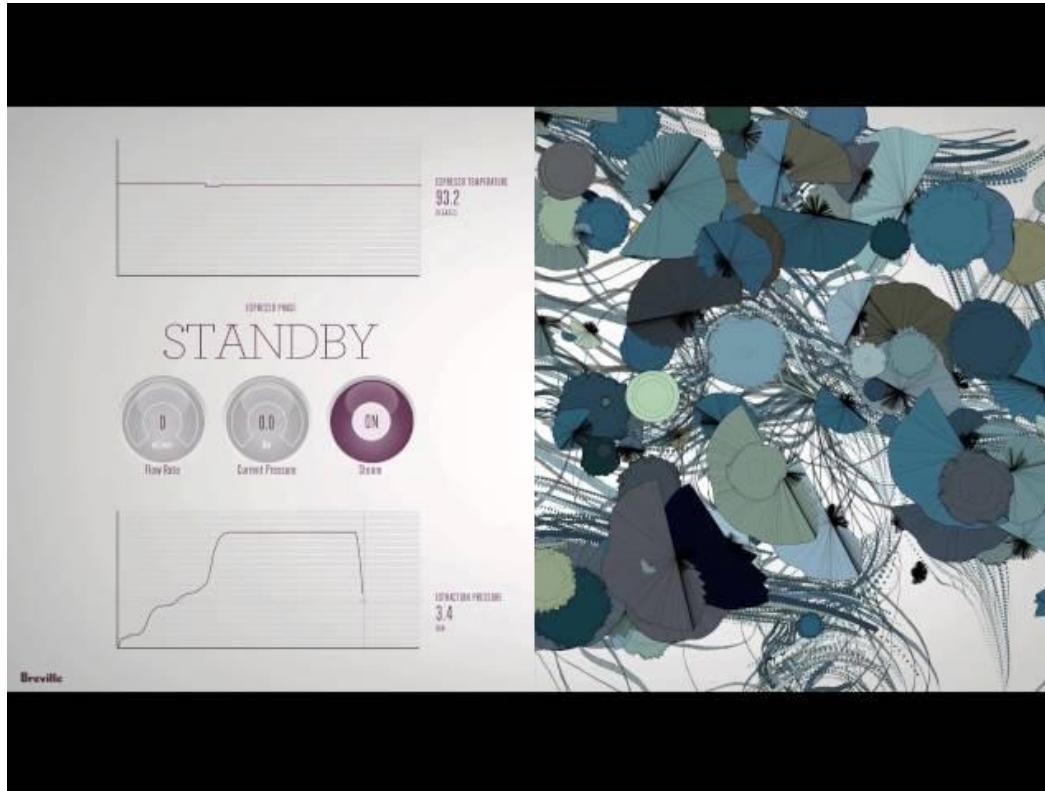
Introducción a Arduino: Proyectos

Hockey



Introducción a Arduino: Proyectos

Física del café



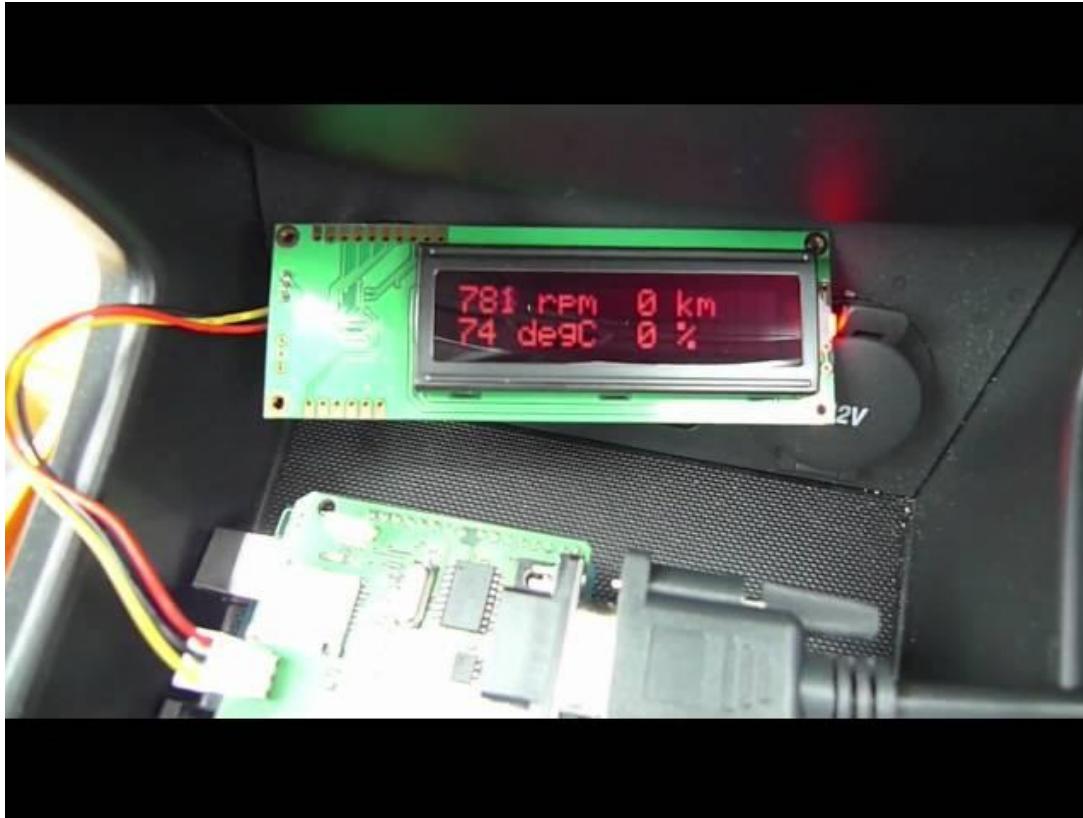
Introducción a Arduino: Proyectos

Alimentador de mascotas activado por twitter



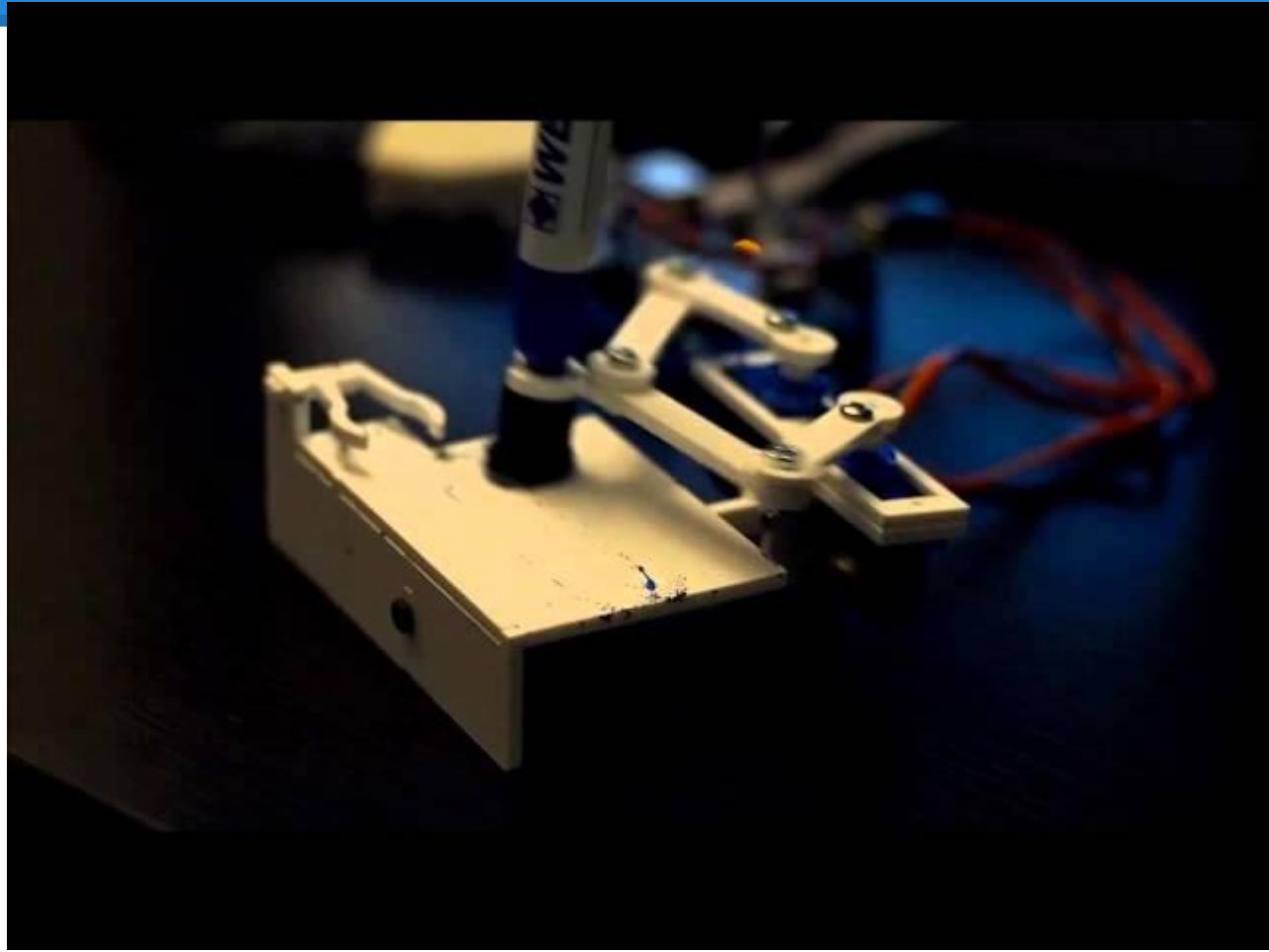
Introducción a Arduino: Proyectos

Can Bus: Centralita de un coche



Introducción a Arduino: Proyectos

Plot clock



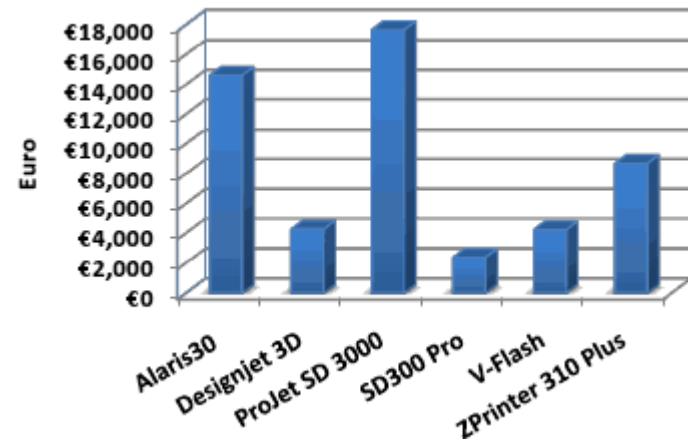
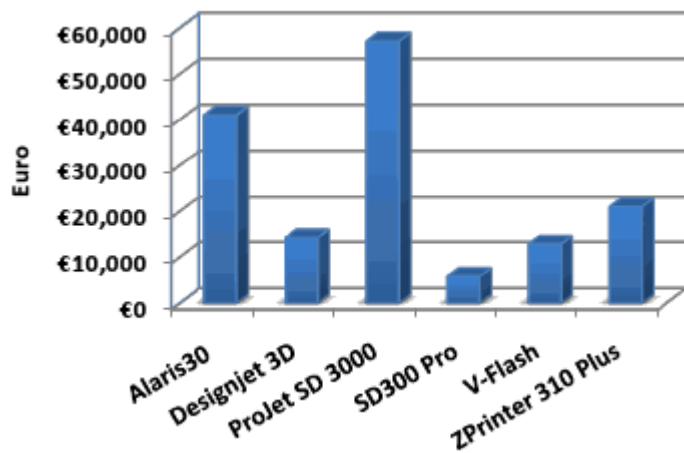
Introducción a Arduino: Proyectos

Impresoras 3D: PrintrBot, RepRap, ...

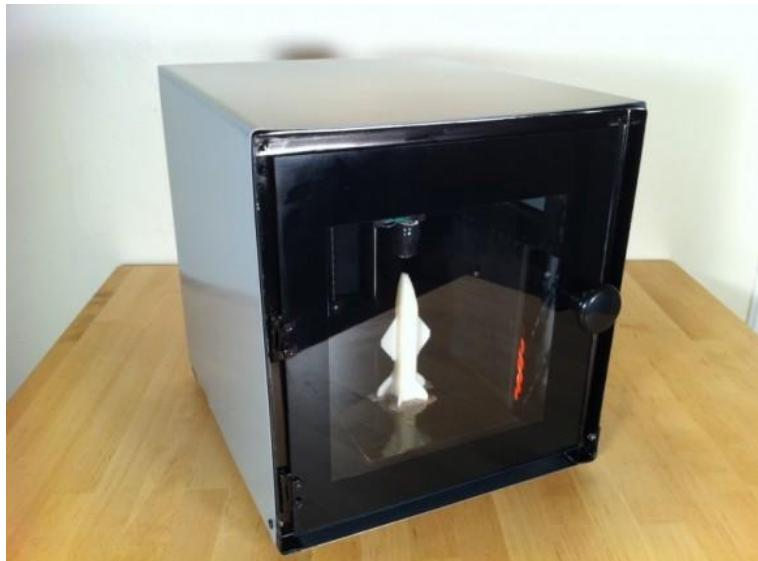


Introducción a Arduino: Proyectos

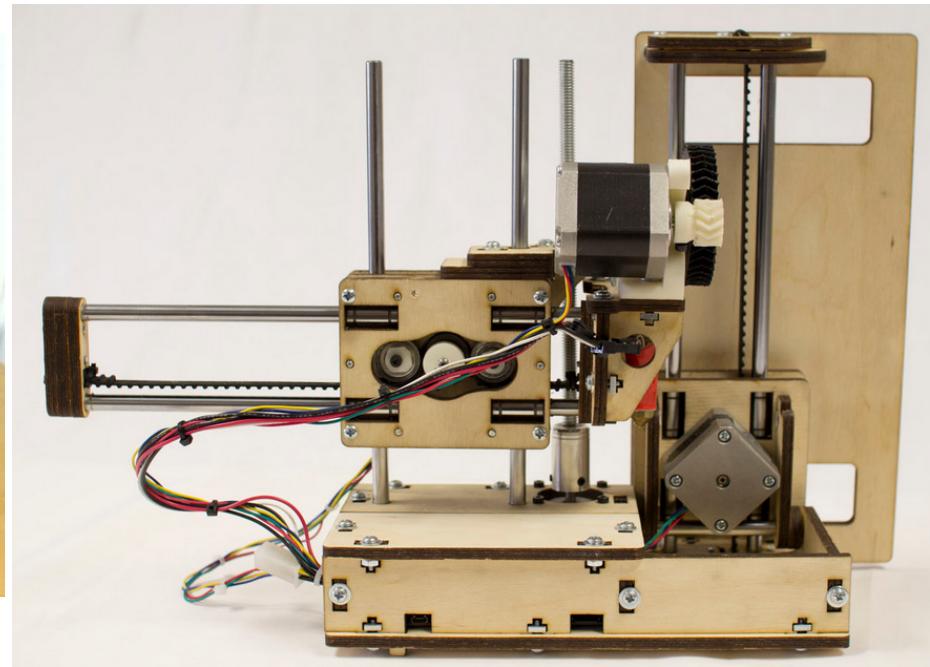
2010



Introducción a Arduino: Proyectos



599€



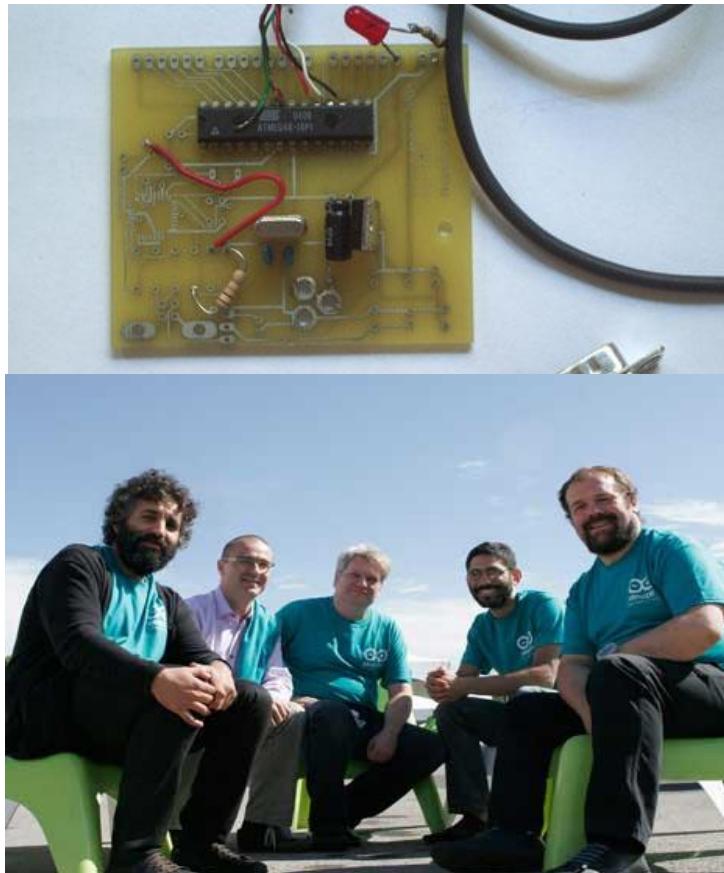
399€



elcacharreo.com

ETSIIT

Introducción a Arduino: Historia



- Maximo Banzi 2005
- Un bar le da nombre
- Made in Italy
- Computación física
- Precio objetivo 30\$
- 100% free source

David Cuartielles @dcuartielles

Gianluca Martino

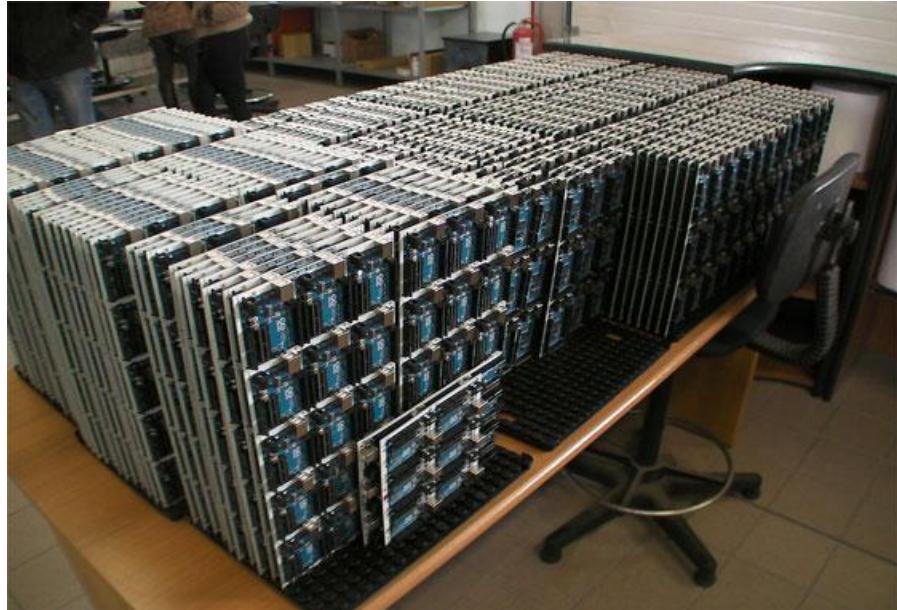
Tom Igoe @tigoe

David Mellis @mellis

Massimo Banzi @mbanzi



Introducción a Arduino: Presente



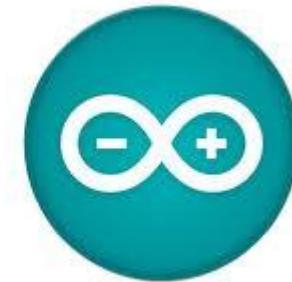
300.000 en Mayo de 2011
¿cuantos se venden ahora?

Due
Uno
Leo
Ethernet
Mega
Mini
Pro
Lilypad
Bluetooth
Yun



Introducción a Arduino: ~~Futuro~~

- Galileo
- Tree
- Wifi (BBB)
- Android ADK
- ...



Introducción a Arduino: Donde

Makers/DIY



MakeProjects.com

Instructables.com



elcacharreo.com

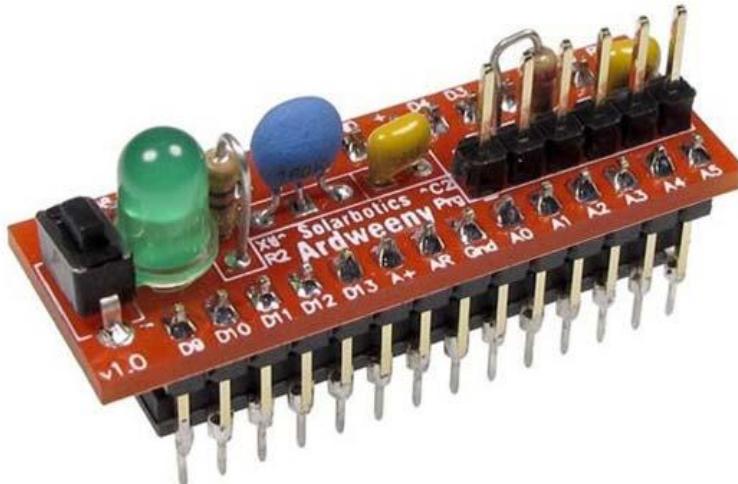
ETSIIT

Introducción a Arduino: Licencias

¿Gratis como la cerveza?

software libre

hardware libre



ejemplo

como montarlo



Introducción a Arduino: Trabajo

Hablando de Hardware

Hablando de Colaboración

The screenshot shows a Google Sheets document with a blue header bar. The title 'Personal Monthly Budget' is at the top left. The menu bar includes File, Edit, View, Insert, Format, Form, Tools, Help, and a 'Share' button. The main content is a budget spreadsheet with several sections:

- INCOME:** Total Income is \$2,000.00.
- HOME EXPENSES:** Total is \$1,381.00.
- DAILY LIVING:** Total is \$619.00.
- ENTERTAINMENT:** Total is \$0.00.

Each section has columns for Projected, Actual, and Difference. The spreadsheet is autosaved on 12/2/09.

Trabajo colaborativo



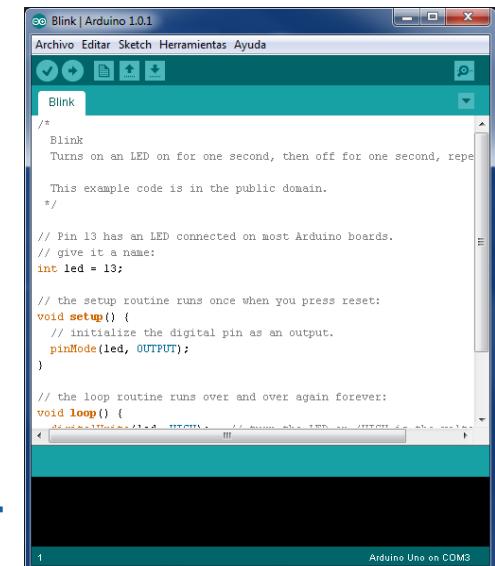
IDE: instalación

Descargamos el IDE de la página de descargas de Arduino



The screenshot shows the Arduino Software download page. At the top is the Arduino logo. Below it is a search bar with the placeholder "Search". A large blue button labeled "Descargar" is prominently displayed. To its left, there's a section titled "Download the Arduino Software" with a brief description and a "Get Started" link. On the right, there's a "Licencia" section with a "Read License" link. Below these are two download links: "Windows" and "Mac OS X". Under each link, there's a file size (38.05 MB) and a "Download" button. To the right of the download buttons is a "More setup" section with links to "Getting Started", "Environment", "Examples", "Foundations", and "FAQ". At the bottom, a note states "Downloads will be available from arduino.cc soon."

Descargar

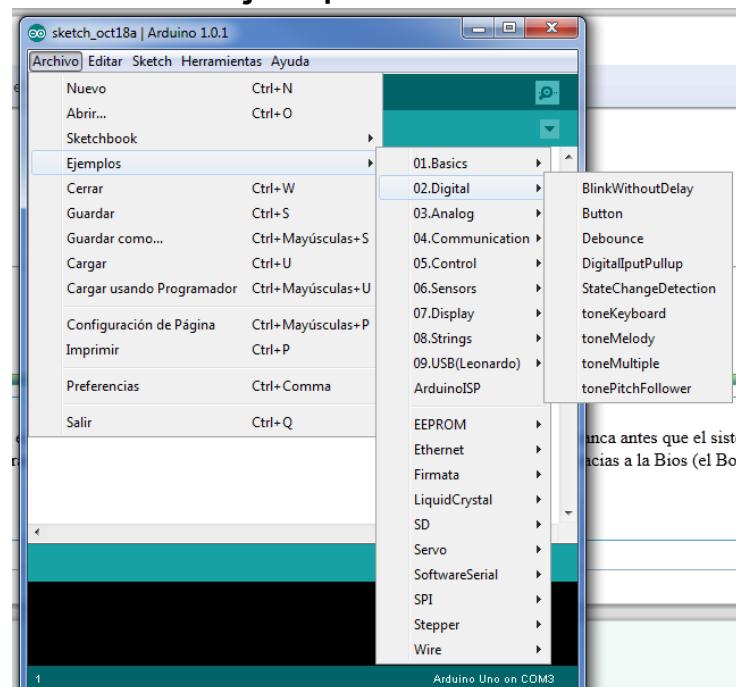


<http://arduino.cc/en/Main/Software>



IDE: instalación

Una vez descargado, lo descomprimimos en una carpeta
y a probar los ejemplos!!!



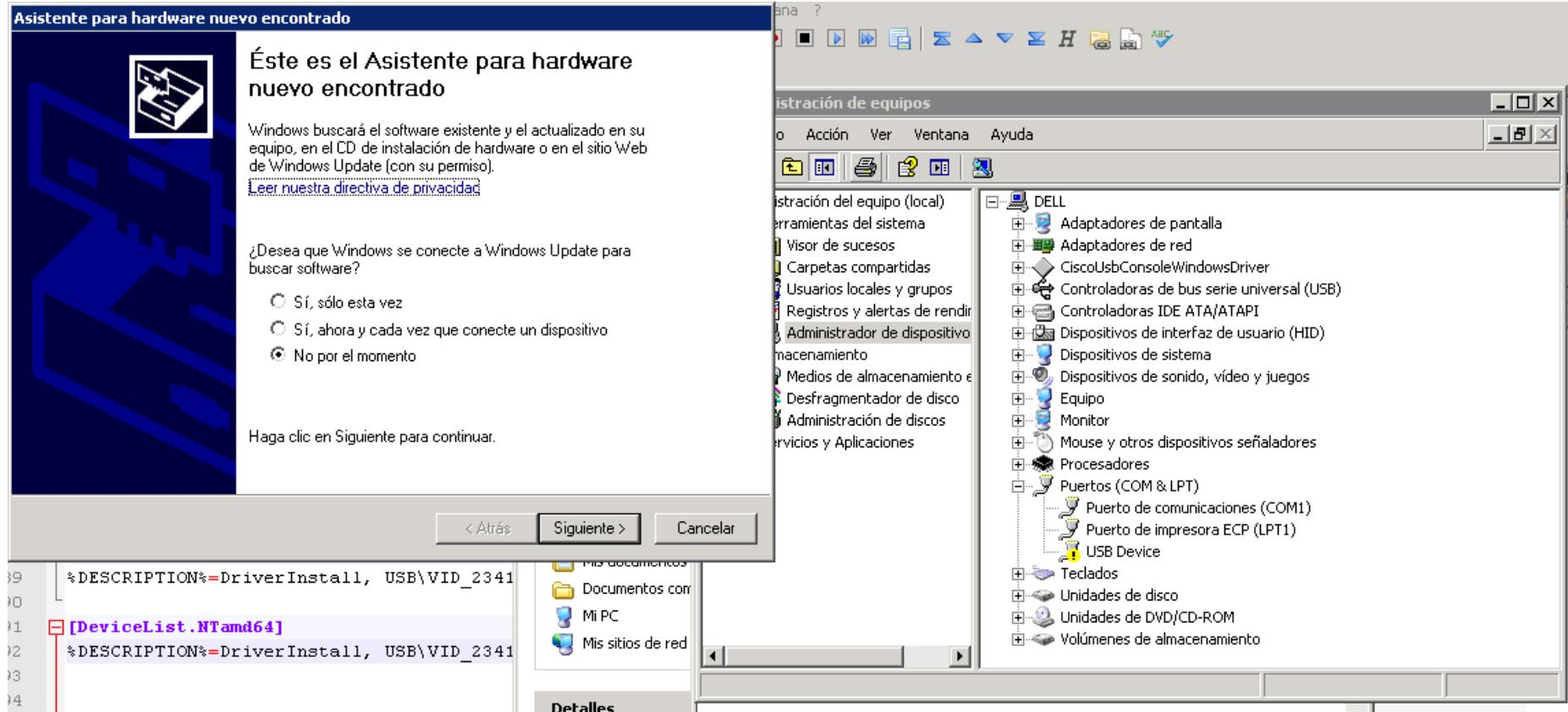
IDE: instalación Linux

```
sudo usermod -aG dialout <myuser>
```

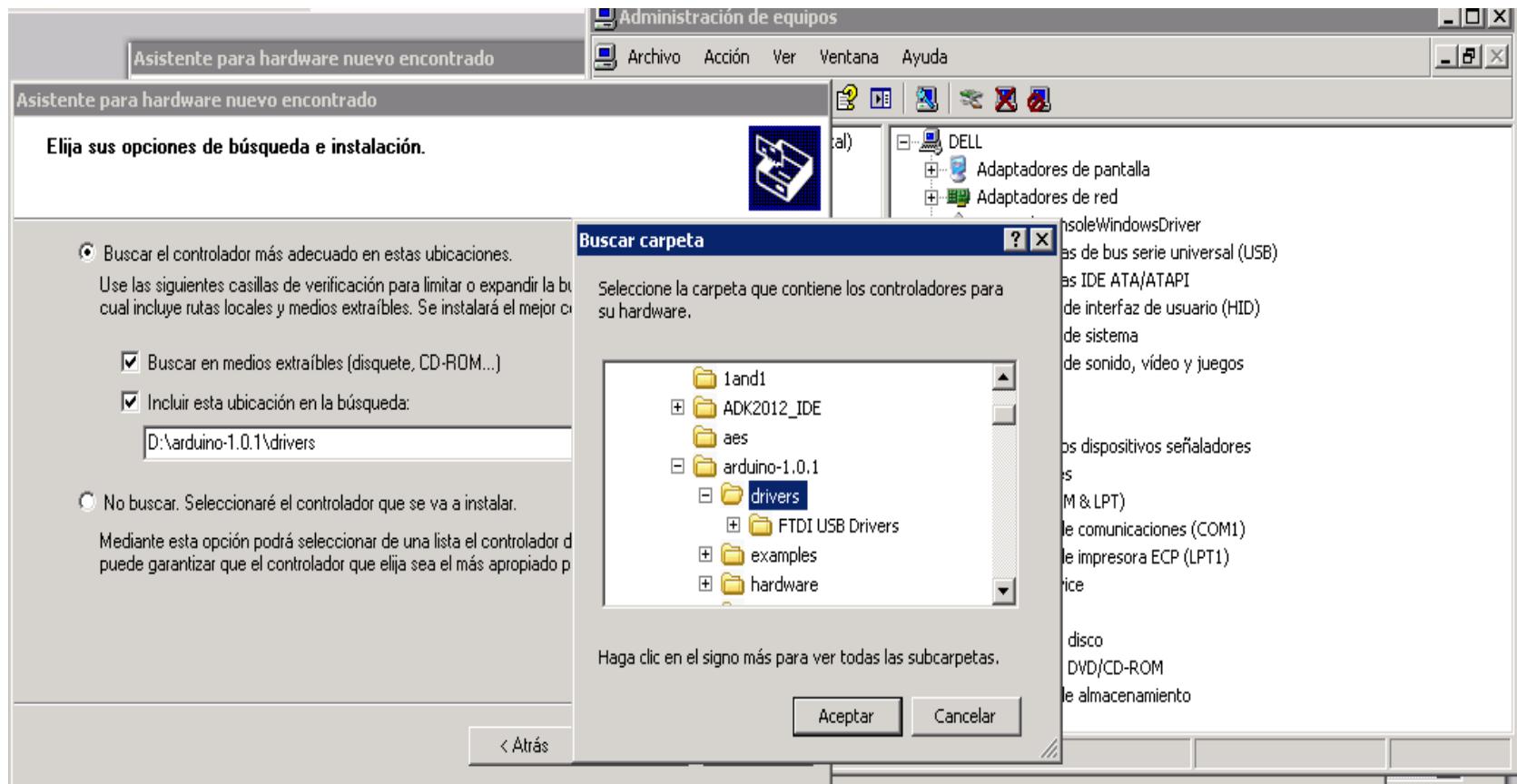


IDE: Drivers windows

El problema más habitual es el instalar el driver de arduino bajo Windows



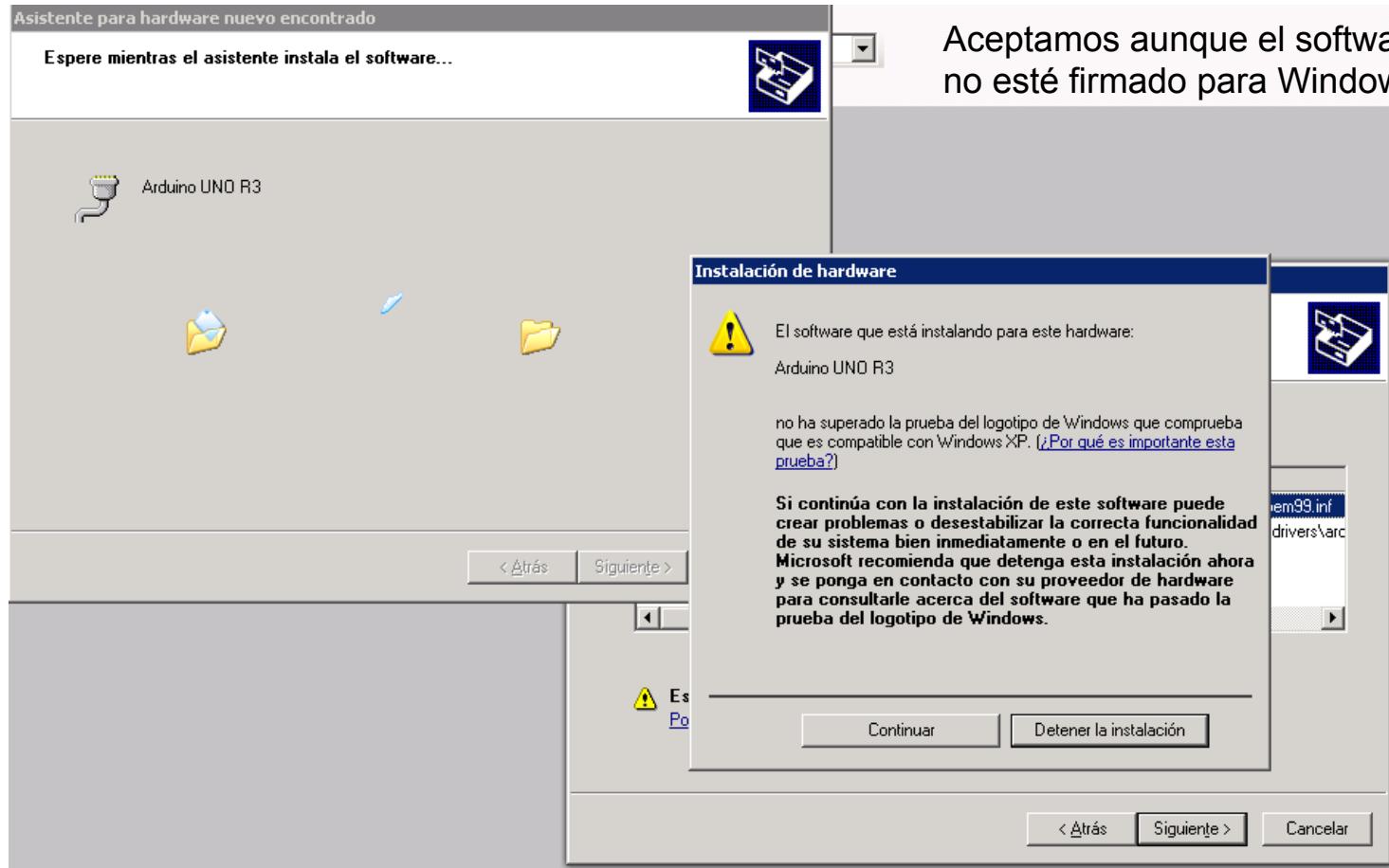
IDE: Drivers windows



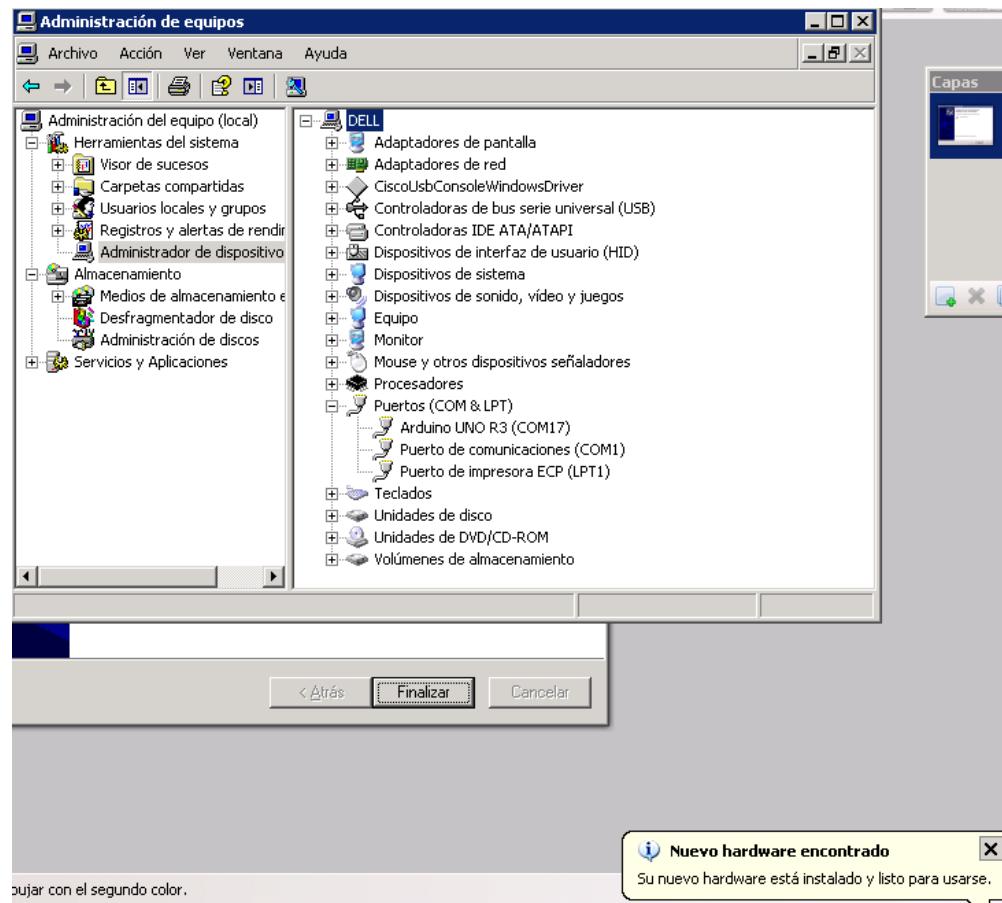
La clave es seleccionar el directorio "drivers" que contiene "FTDI USB Drivers"



IDE: Drivers windows



IDE: Drivers windows



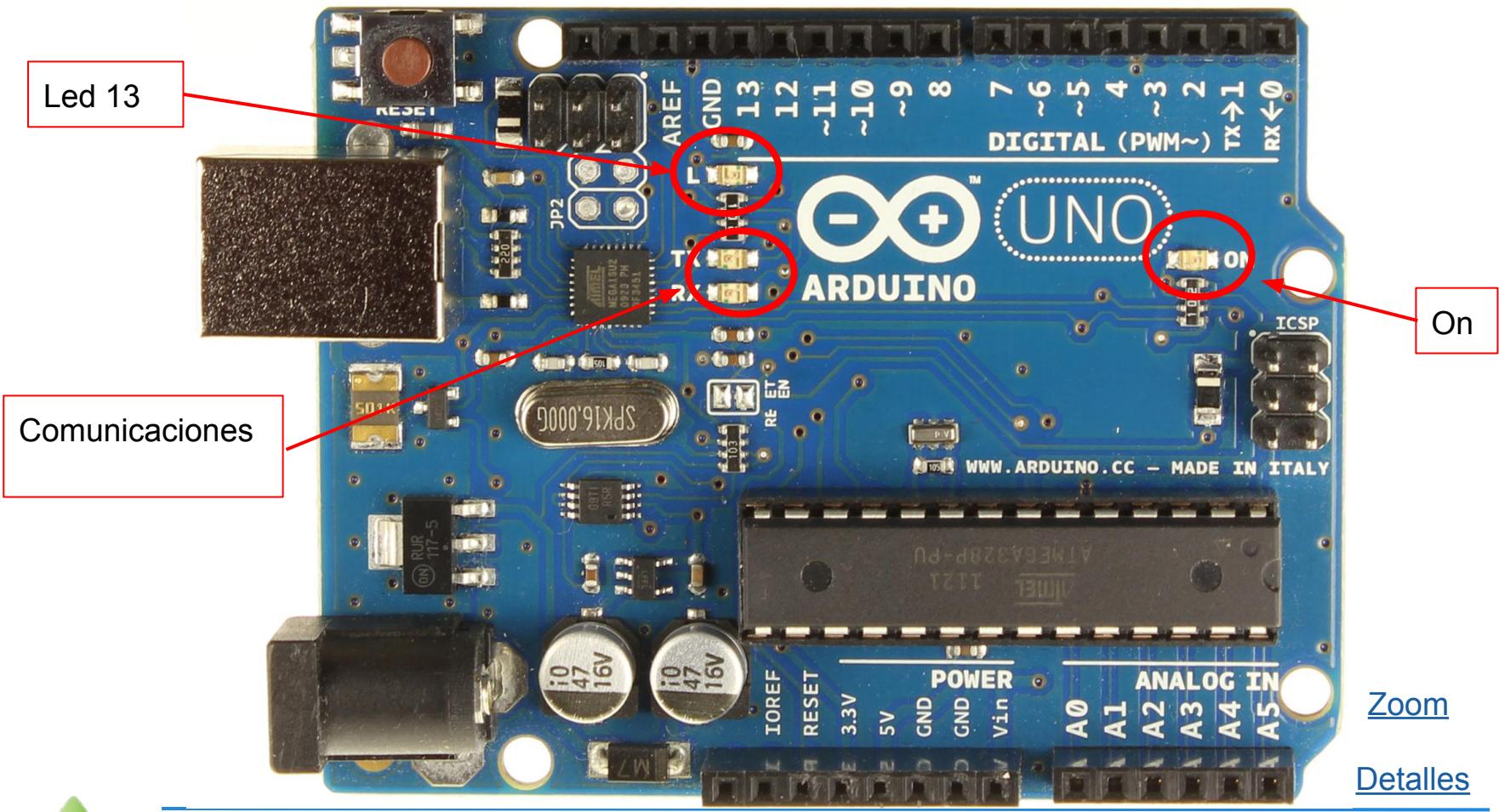
Hardware Arduino: Uno Rev 3

Uno Rev 3

- [Zoom](#)
- 5V
- aTMege328
- 14 digitales (6 PWM) + 6 analógicos
- 32Kb + 2Kb + 1Kb 16MHz
- 1 UART
- [Detalles](#)



Hardware Arduino: Bloques



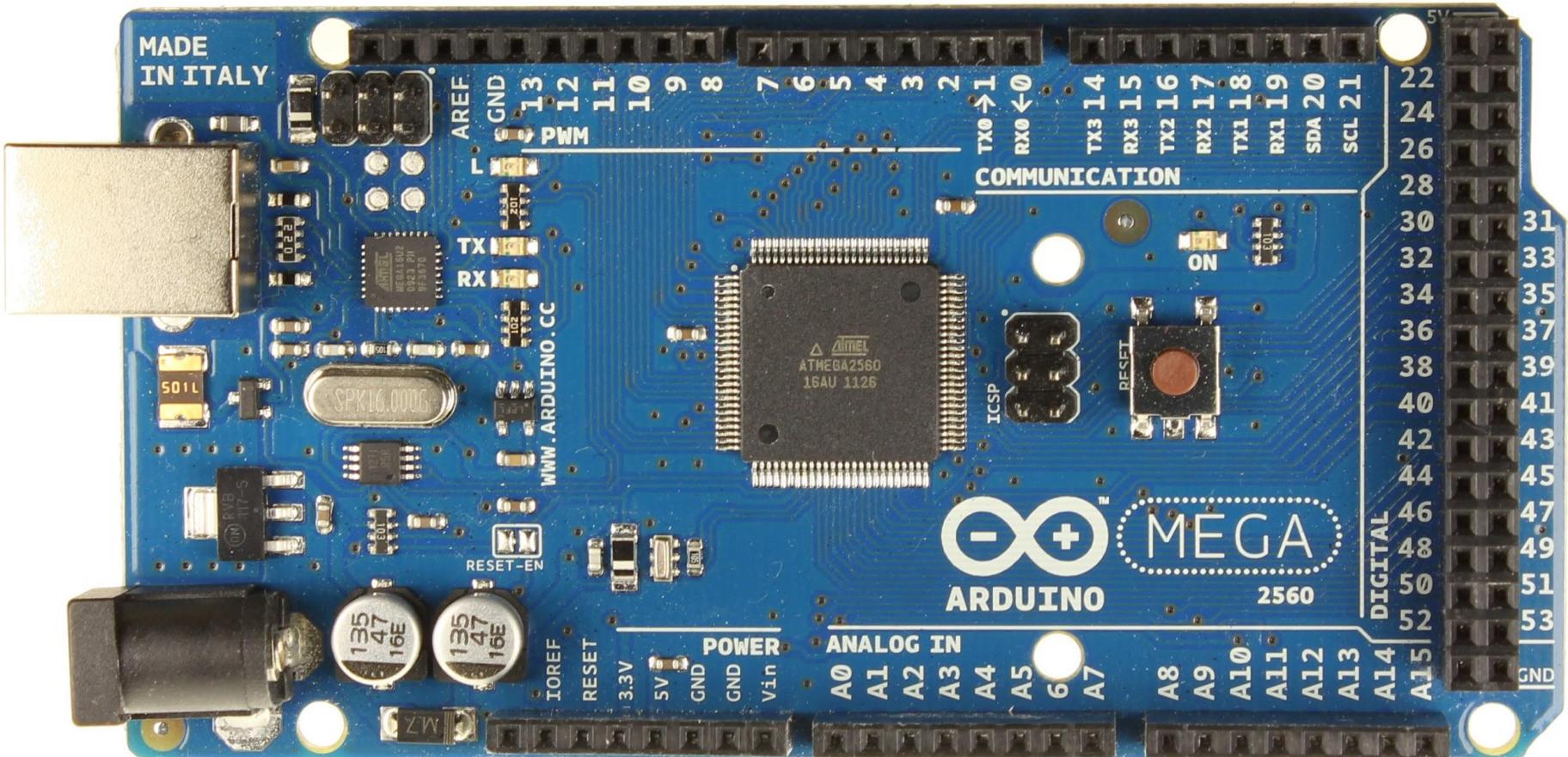
Hardware Arduino: Mega 2560

Mega 2560

- 5V
- atMega 2560
- [Zoom](#)
- 54 digitales (15PWM) + 16 analógicos
- 256Mb + 8Mb + 4Mb 16MHz
- 4 UART
- [Detalles](#)



Hardware Arduino: Mega 2560



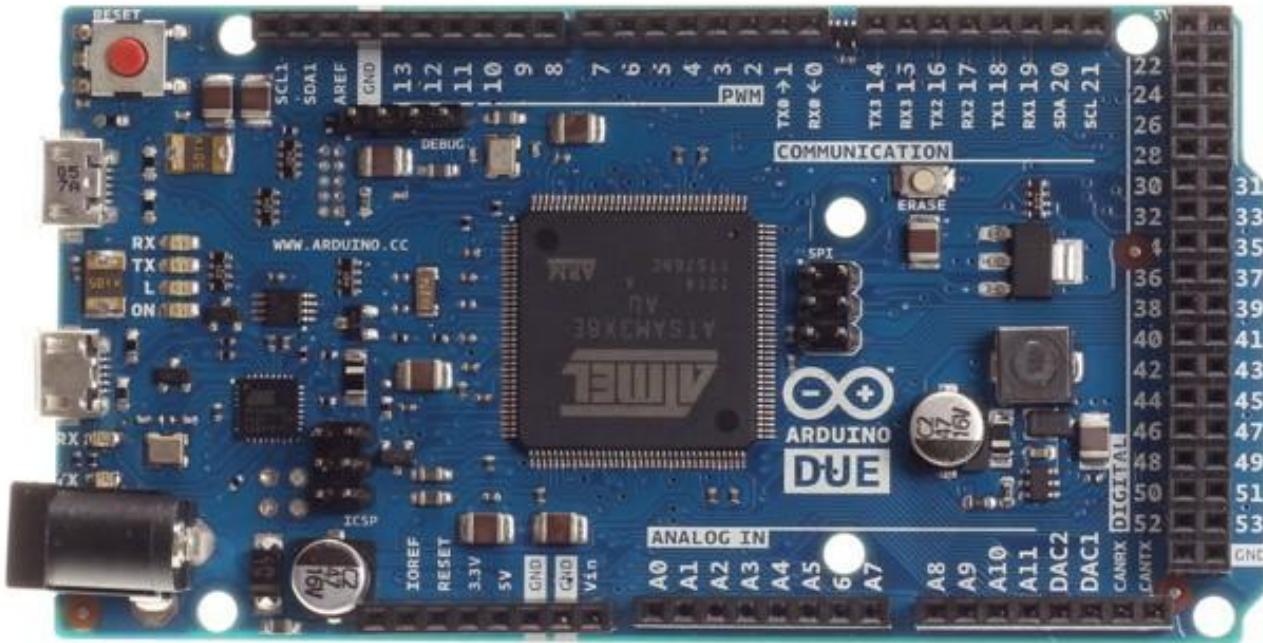
Hardware Arduino: DUE

Due

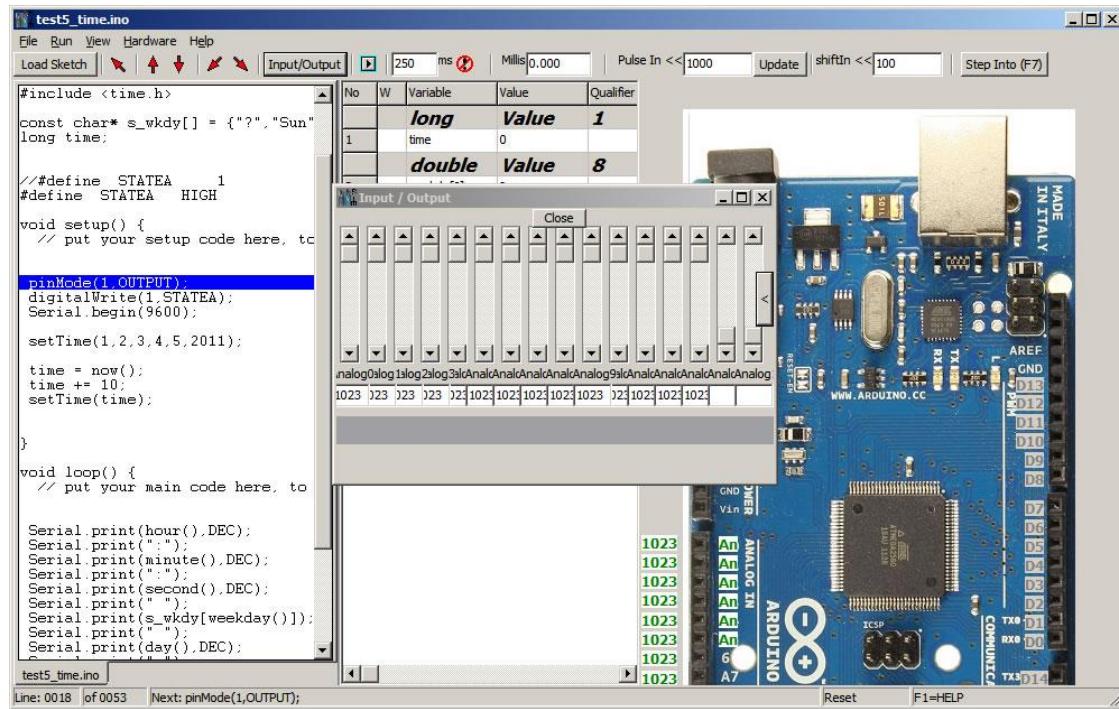
- 3.3V
- ARM SAM3X
- Zoom
- 54 digitales (8PWM) + 12 analógicos +
2 DAC
- 512Kb + 96Kb + 0Kb 84MHz
- DMA
- 4 UART
- Detalles



Hardware Arduino: DUE



Emuladores: ¿y si no tengo arduino?

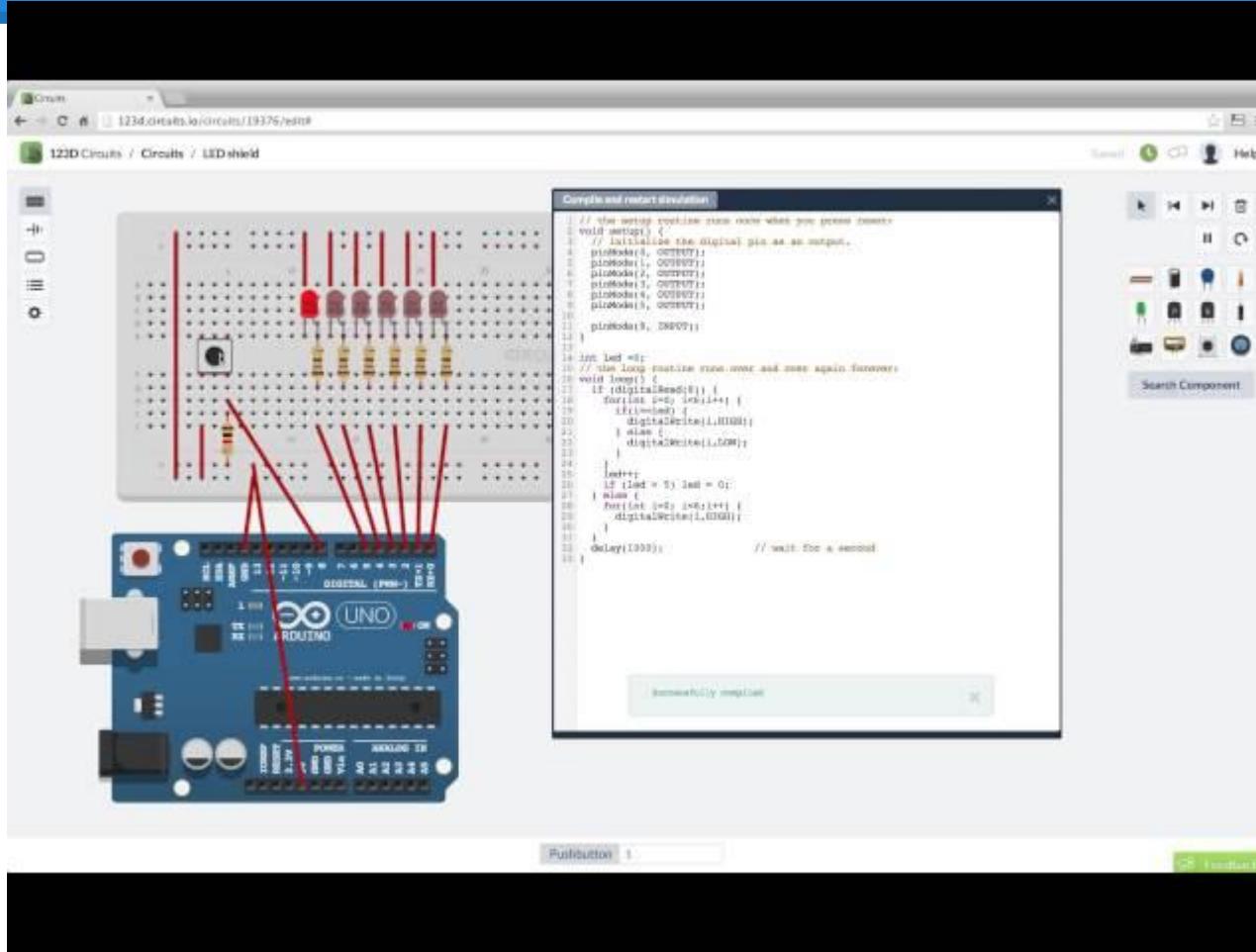


[Simulator for Arduino](#)

[Virtualbreadboard](#)



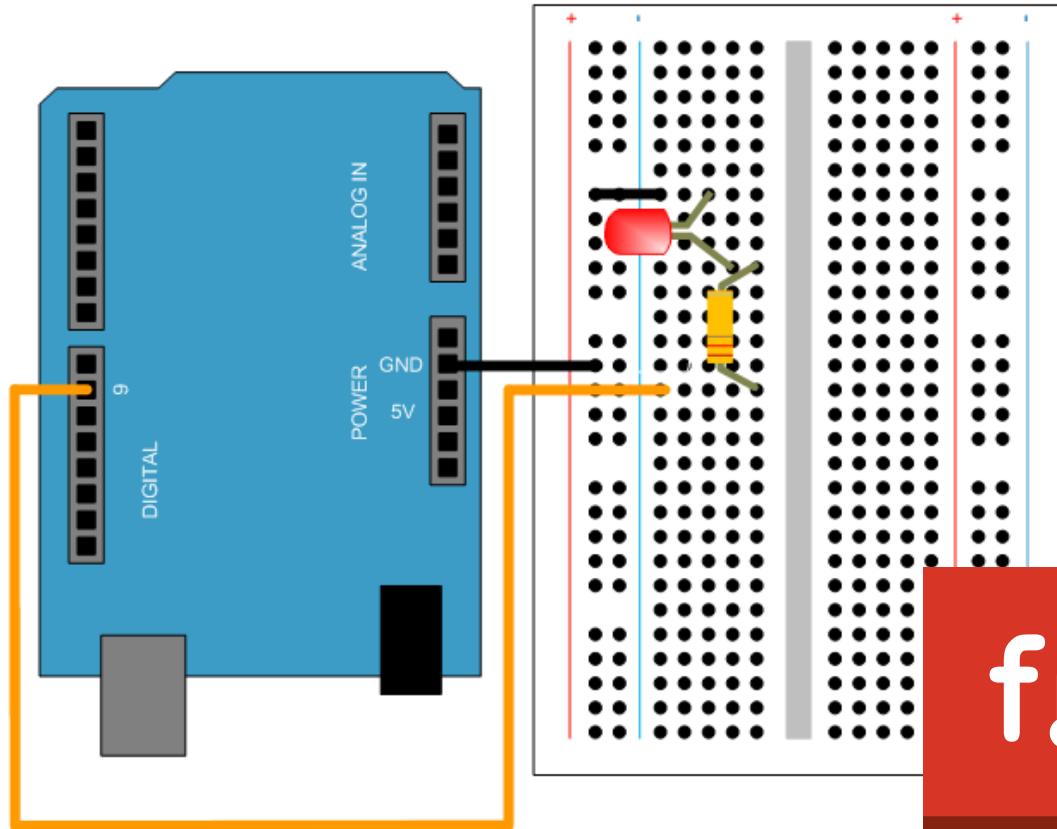
Emuladores: ¿y si no tengo arduino?



123d.circuits.io



Prototipos: LED en la Breadboard



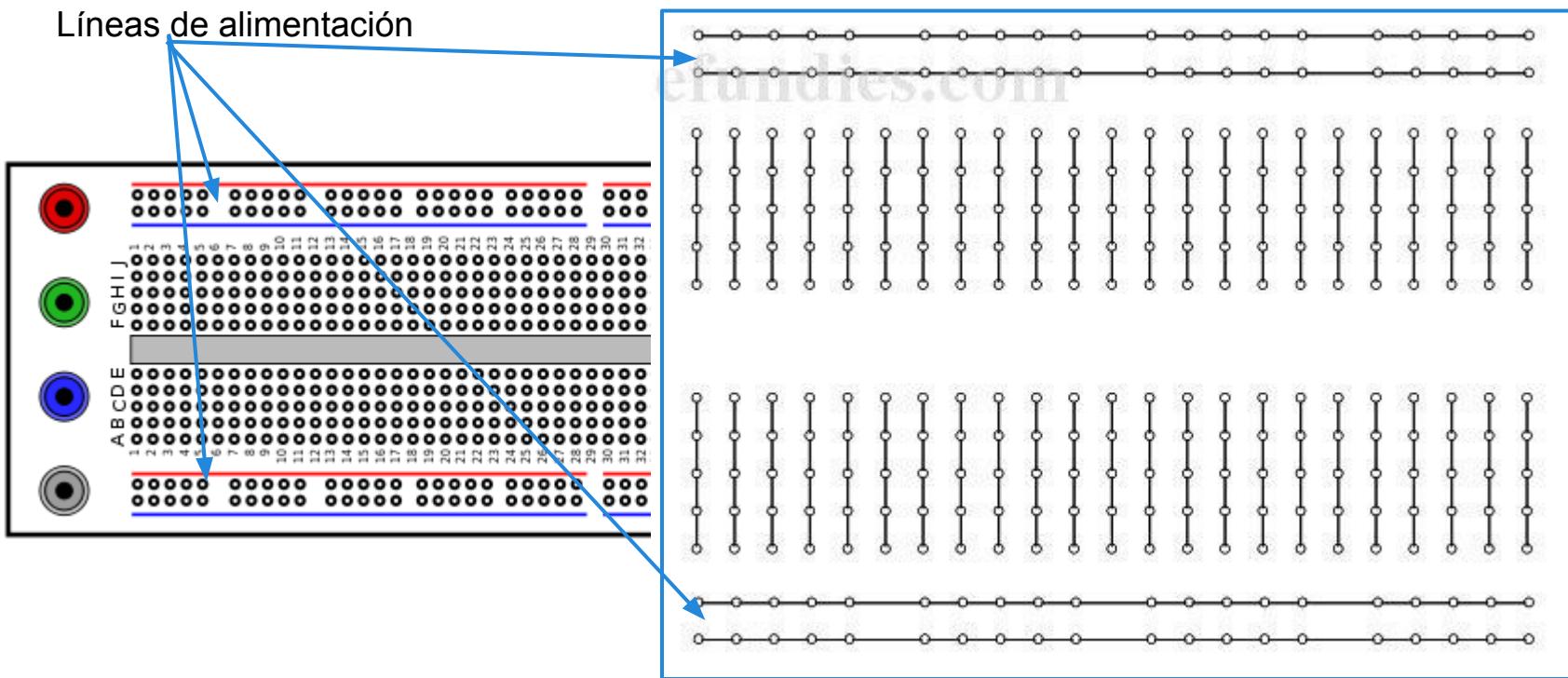
Veamos como sería
un montaje de un led
en una breadboard

<http://fritzing.org/home/>

fritzing



Prototipos: Breadboard



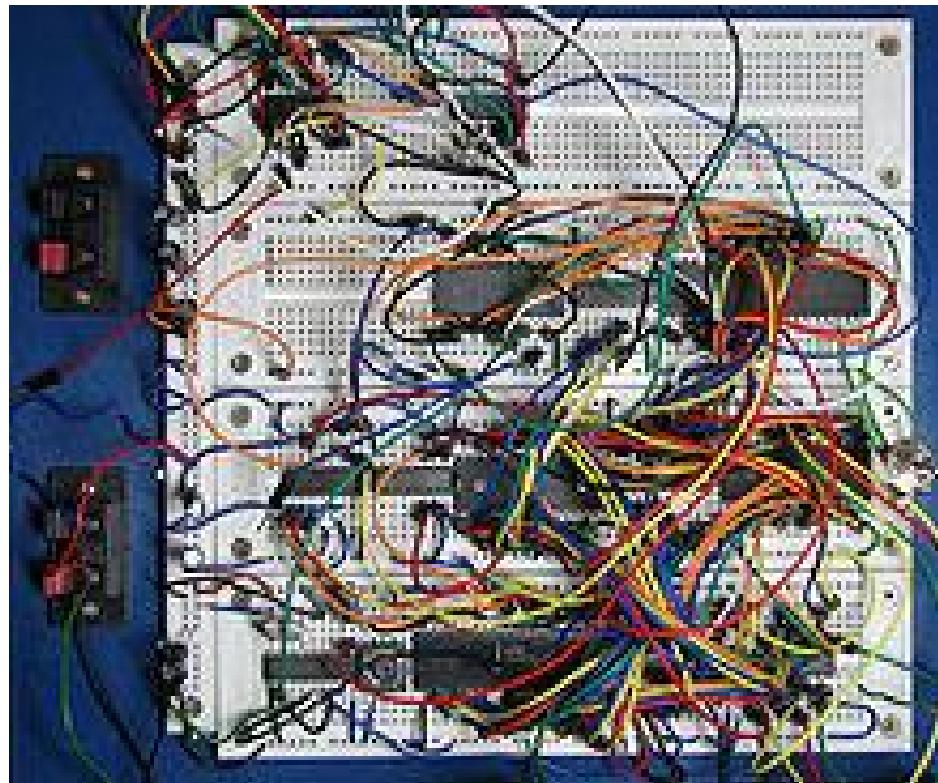
Estas son las conexiones internas que existen en una placa breadboard:

- Existen esas líneas largas que se utilizan para alimentación a lo largo de la placa.
- Las líneas perpendiculares unen 5 puntos (etiquetados como ABCDE) de cada fila

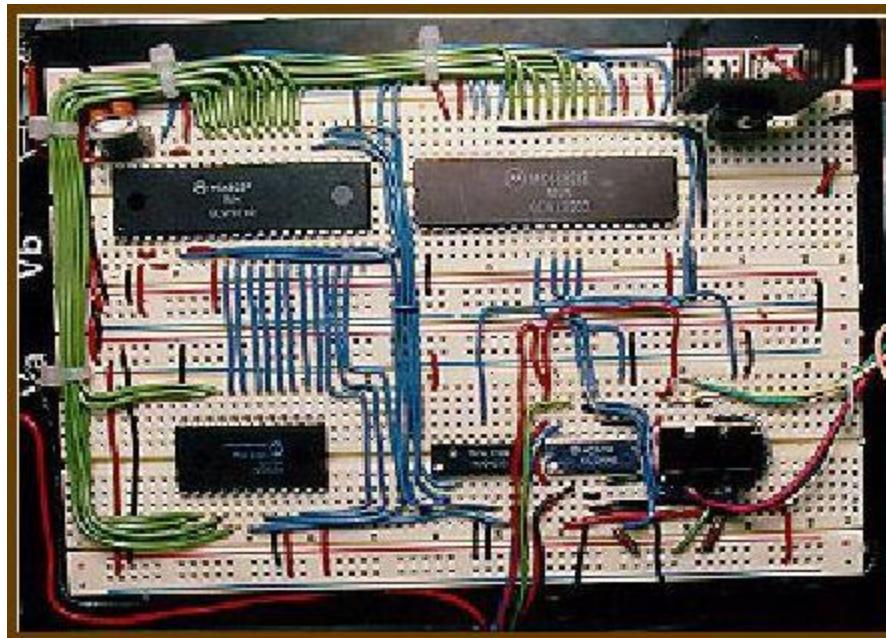
A la derecha se ven las conexiones que se pueden ver por la parte de abajo



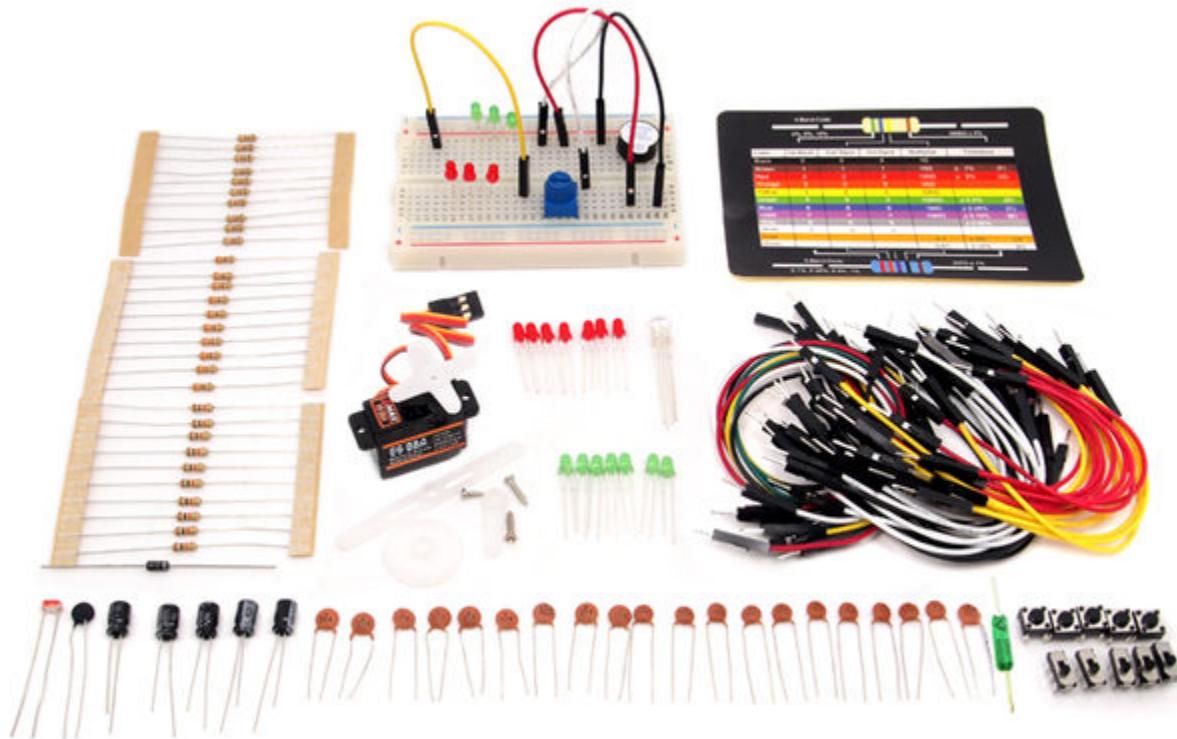
Prototipos: Ejemplo para NO repetir



Prototipos: Así SI



Componentes: Kit



http://www.seeedstudio.com/wiki/index.php?title=Arduino_Sidekick_Basic_Kit



Componentes: Motor DC



Motor CC 3 a 6 V

3 a 6 V.
Ø eje 2,3 mm.
Ø motor 22 mm.
L=27 mm.

<http://www.electan.com/motor-cc-3-a-6-v-p-1540.html>

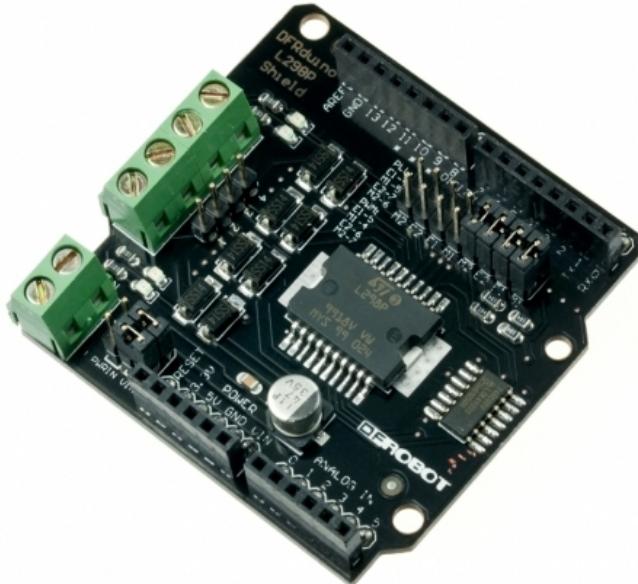


ElCacharreo.com

ETSIIT

Componentes: Controlador Motor

Arduino Shield Motor 2A DFRobot



Características:

- Soporta motores de 4.8 a 35V
- Hasta 2A cada motor
- Utiliza los pines 5,6,7,8 para controlar 2 motores DC
- Soporta control de velocidad PWM.
- Soporta PLL.

<http://www.electan.com/arduino-shield-motor-dfrobot-p-3158.html>

[http://www.dfrobot.com/wiki/index.php?title=Arduino_Motor_Shield_\(L298N\)_%28SKU:DRI0009%29](http://www.dfrobot.com/wiki/index.php?title=Arduino_Motor_Shield_(L298N)_%28SKU:DRI0009%29)



Componentes: Motor Paso a Paso



Small Stepper Motor

This is a **Bipolar** motor.

Features:

- Stride Angle (degrees) : 7.5
- 2-Phase
- Rated Voltage : 12V
- Rated Current : 400mA
- 3mm Diameter Drive Shaft
- 4-Wire Cable Attached
- In-traction Torque : 100 g/cm

<http://www.electan.com/motor-paso-paso-pequeno-p-3297.html>



ElCacharreo.com

ETSIIT

Componentes: LCD

6. Interface pin description



Pin no.	Symbol	External connection	Function
1	V _{ss}	Power supply	Signal ground for LCM
2	V _{DD}		Power supply for logic for LCM
3	V ₀		Contrast adjust
4	RS	MPU	Register select signal
5	R/W	MPU	Read/write select signal
6	E	MPU	Operation (data read/write) enable signal
7~10	DB0~DB3	MPU	Four low order bi-directional three-state data bus lines. Used for data transfer between the MPU and the LCM. These four are not used during 4-bit operation.
11~14	DB4~DB7	MPU	Four high order bi-directional three-state data bus lines. Used for data transfer between the MPU
15	LED+	LED BKL power supply	Power supply for BKL
16	LED-		Power supply for BKL

<http://www.electan.com/basic-16x2-character-lcd-amber-black-33v-p-4144.html>



Programando Arduino: IDE

The screenshot shows the Arduino IDE interface with the title bar "Blink | Arduino 1.0". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for Save, Run, Upload, and others. The main area displays the "Blink" sketch code. The code is a classic example that turns an LED on for one second and off for one second, repeating indefinitely. It uses pins 13 and 12. The code is as follows:

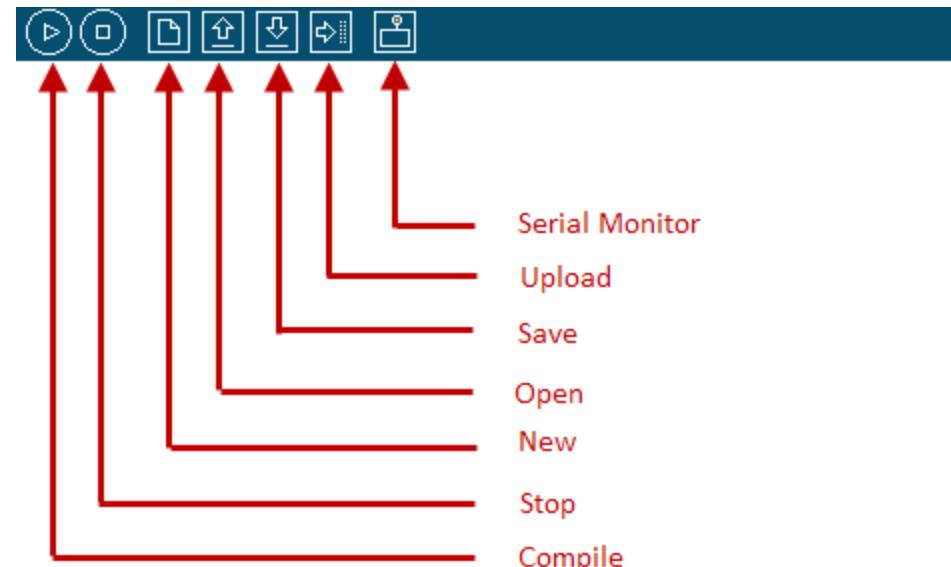
```
/*
Blink
Turns on an LED on for one second, then off for one second, repeats.

This example code is in the public domain.
*/

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH);    // set the LED on
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);     // set the LED off
  delay(1000);              // wait for a second
}
```

At the bottom, a progress bar indicates "Compiling sketch..." and a status bar shows "Arduino Uno on COM3".



Programando Arduino: Lenguaje

```
void setup()  
{ }
```

```
void loop()  
{ }
```

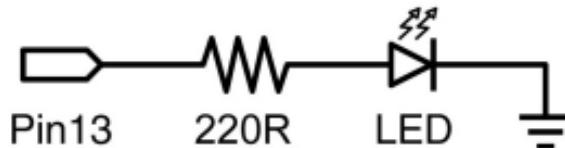
Guía de referencia de Arduino

Librerías



El mundo digital: salidas

digital output



This is the basic 'hello world' program used to simply turn something on or off. In this example, an LED is connected to pin13, and is blinked every second. The resistor may be omitted on this pin since the Arduino has one built in.

```
int ledPin = 13; // LED on digital pin 13

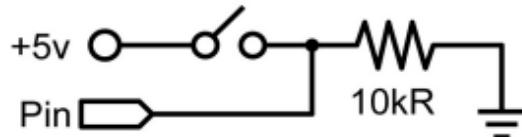
void setup() // run once
{
    pinMode(ledPin, OUTPUT); // sets pin 13 as output
}

void loop() // run over and over again
{
    digitalWrite(ledPin, HIGH); // turns the LED on
    delay(1000); // pauses for 1 second
    digitalWrite(ledPin, LOW); // turns the LED off
    delay(1000); // pauses for 1 second
}
```



El mundo digital: entradas

digital input



This is the simplest form of input with only two possible states: on or off. This example reads a simple switch or pushbutton connected to pin2. When the switch is closed the input pin will read HIGH and turn on an LED.

```
int ledPin = 13;           // output pin for the LED
int inPin = 2;             // input pin (for a switch)

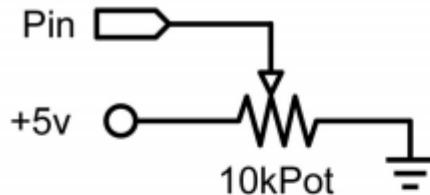
void setup()
{
    pinMode(ledPin, OUTPUT); // declare LED as output
    pinMode(inPin, INPUT);  // declare switch as input
}

void loop()
{
    if (digitalRead(inPin) == HIGH) // check if input is HIGH
    {
        digitalWrite(ledPin, HIGH); // turns the LED on
        delay(1000);             // pause for 1 second
        digitalWrite(ledPin, LOW); // turns the LED off
        delay(1000);             // pause for 1 second
    }
}
```



El mundo analógico: entradas

potentiometer input



Using a potentiometer and one of the Arduino's analog-to-digital conversion (ADC) pins it is possible to read analog values from 0-1024. The following example uses a potentiometer to control an LED's rate of blinking.

```
int potPin = 0;      // input pin for the potentiometer
int ledPin = 13;     // output pin for the LED

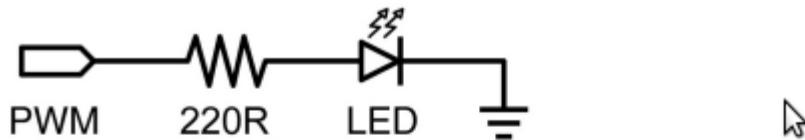
void setup()
{
  pinMode(ledPin, OUTPUT); // declare ledPin as OUTPUT
}

void loop()
{
  digitalWrite(ledPin, HIGH);    // turns ledPin on
  delay(analogRead(potPin));    // pause program
  digitalWrite(ledPin, LOW);     // turns ledPin off
  delay(analogRead(potPin));    // pause program
}
```



El mundo analógico: salidas

pwm output



Pulsewidth Modulation (PWM) is a way to fake an analog output by pulsing the output. This could be used to dim and brighten an LED or later to control a servo motor. The following example slowly brightens and dims an LED using for loops.

```
int ledPin = 9; // PWM pin for the LED

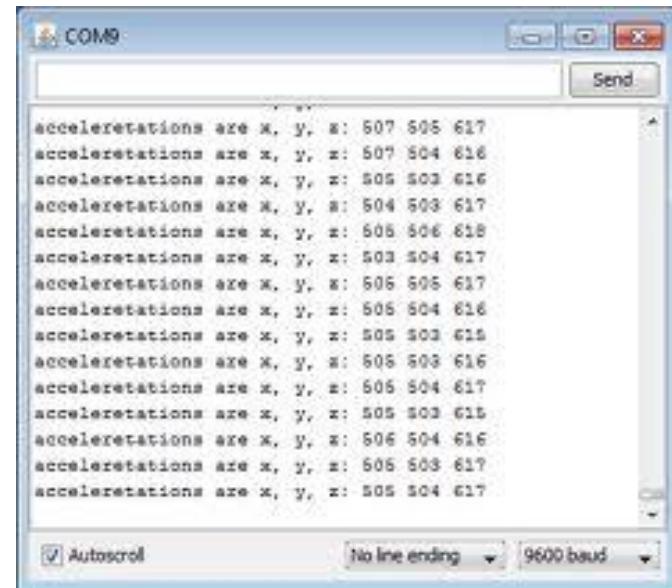
void setup(){}
void loop()
{
    for (int i=0; i<=255; i++) // ascending value for i
    {
        analogWrite(ledPin, i); // sets brightness level to i
        delay(100); // pauses for 100ms
    }
    for (int i=255; i>=0; i--) // descending value for i
    {
        analogWrite(ledPin, i); // sets brightness level to i
        delay(100); // pauses for 100ms
    }
}
```



Comunicaciones: introducción

Comunicando con el pc:

```
void setup () {  
    Serial.begin (9600); }  
  
int i=0;  
void loop () {  
    Serial.print ("hola ");  
    Serial.println (i);  
}
```



Sensores

Existem muitas de sensores:
Cada uno diseñado para medir una magnitud física distinta.
A partir de esa medida generará un voltaje que mediremos con nuestro Arduino conectándolos habitualmente a una entrada analógica



Sensores: Referencia externa

A veces nuestro montaje necesita medir con voltajes diferentes de los 5V habituales. Para ello podemos usar la función

```
analogReference(valor);
```

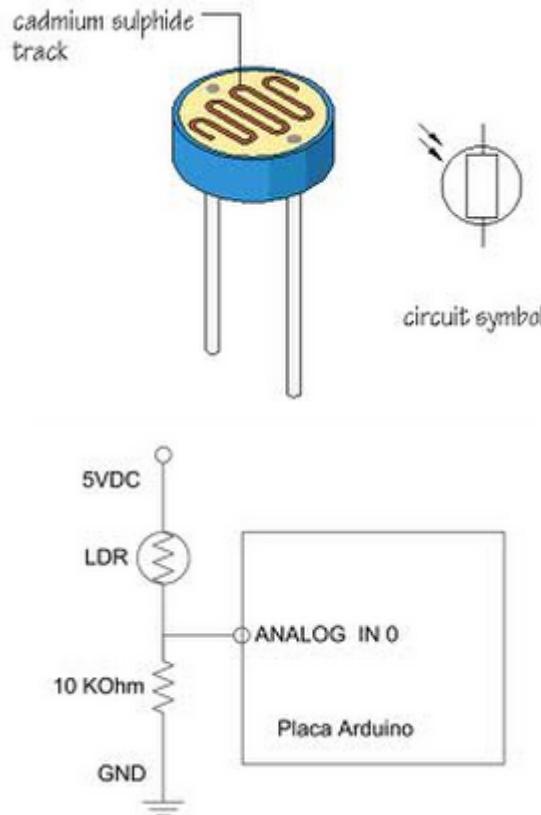
con valor pudiendo ser

- **DEFAULT:** el valor por defecto (5V en placas de 5 y 3.3V en las de 3.3)
- **INTERNAL:** 1.1V en ATmega168 or ATmega328 y 2.56V en ATmega8
- **INTERNAL1V1:** 1.1V (Sólo *Arduino Mega*)
- **INTERNAL2V56:** 2.56V (Sólo *Arduino Mega*)
- **EXTERNAL:** el voltaje de AREF pin (0 to 5V)

Ejemplo muy documentado en [Arduteka](#)



Sensores: LDR (Luz)



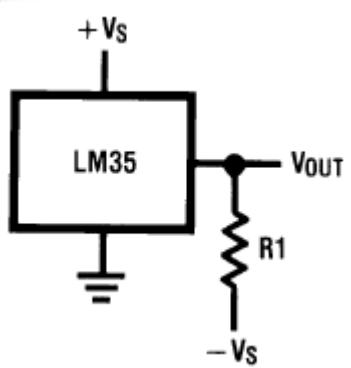
```
int analogPin = 0; // Pin analogico 0 de Arduino
void setup()
{
    Serial.begin(9600);
}
void loop ()
{
    int valorLuz = analogRead(analogPin);
    Serial.print(valorLuz);
    delay(5000);
}
```



Sensores: Temperatura

LM35

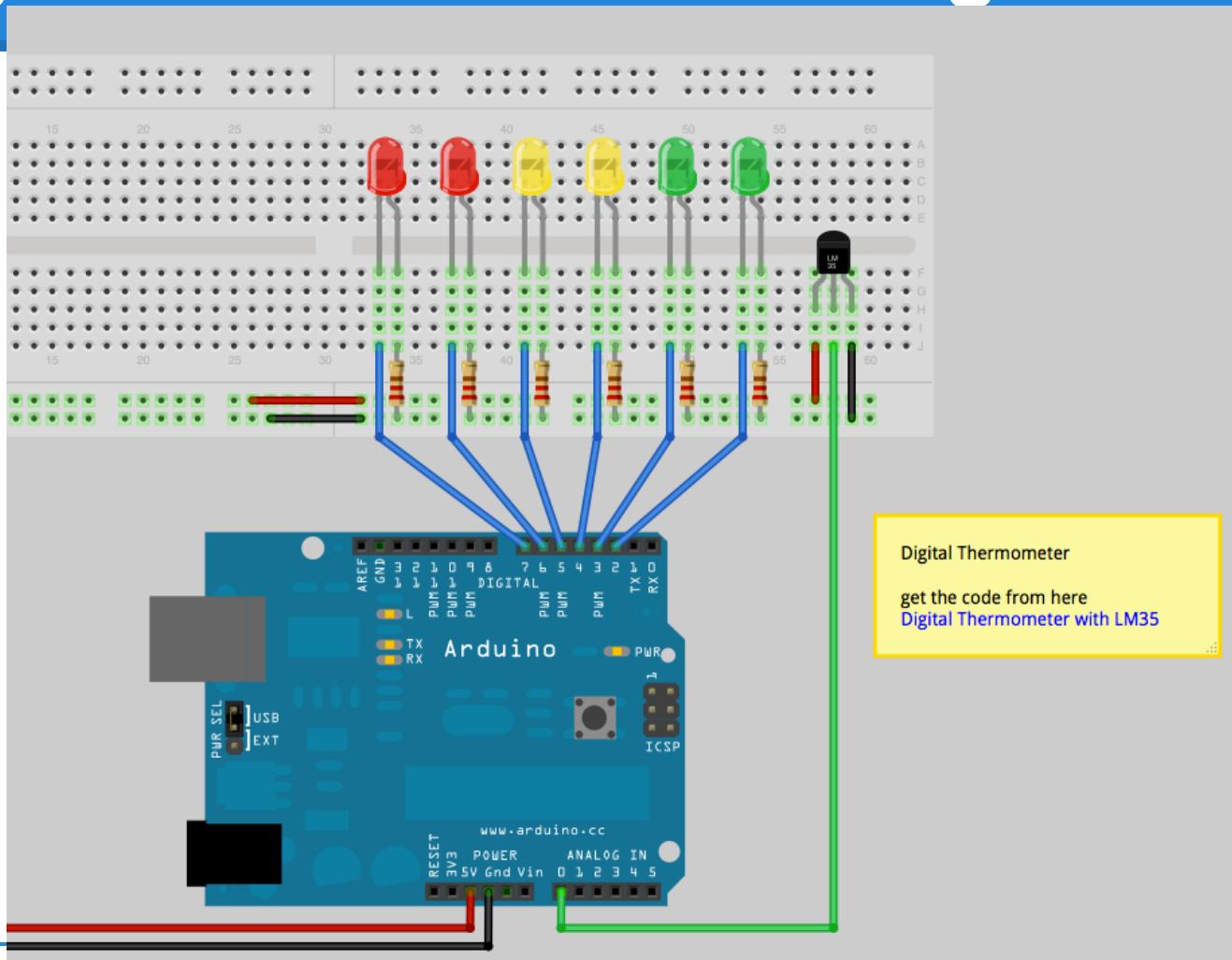
Especificación



```
// Ejemplo en http://fritzing.org/projects/digital-thermometer-with-Lm35/
int analogPin = 0; // Pin analogico 0 de Arduino
int temperature = 0;
int lm35read;
void setup()
{
    Serial.begin(9600);
}
void loop ()
{
    lm35read = analogRead(analogPin);
    temperature = (5.0 * lm35read * 100.0)/1024.0;
    Serial.print(temperature);
    Serial.println("C");
    delay(5000);
}
```



Proyectos: Termómetro digital



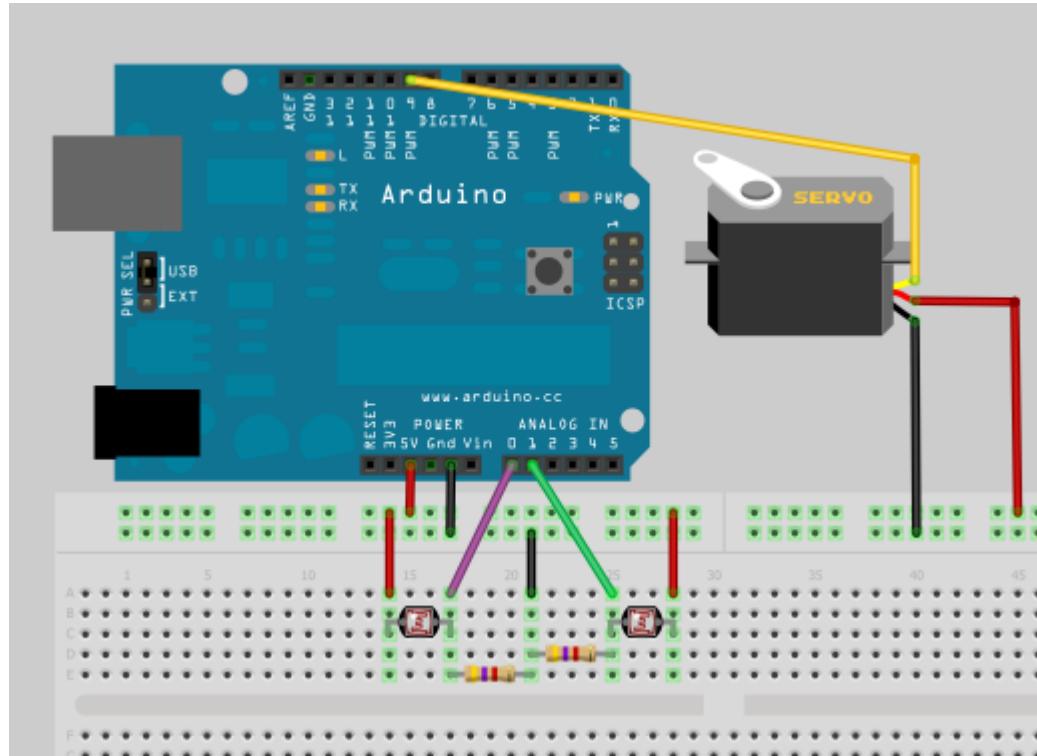
Proyectos: Termómetro digital

```
int startTemp=20;
void setup()
{ Serial.begin(9600);
  for (int i=2;i<8; i++)
    { pinMode(i,OUTPUT); }
}
void loop()
{ int lm35read = analogRead(A0);
  float temperature = (5.0 * lm35read * 100.0)/1024.0;
  temperature = temperature*0.488;
  for (int i=0;i<8; i++){
    if (temperature>((i*2)+startTemp))
      { digitalWrite(i,HIGH); }
    else
      { digitalWrite(i,LOW); }
  }

  Serial.print(temperature);
  Serial.println(" °C");
  delay(500);
}
```



Proyectos: Comparador luminoso



Reciclando hardware

	Impresoras	Videos	Lectores DVD	Coches RF
Motores	X	X	X	X
Led	X	X		
Laser			X	
Drivers	X	X	X	X
Componentes	X	X	X	X



Reciclando hardware: Samples

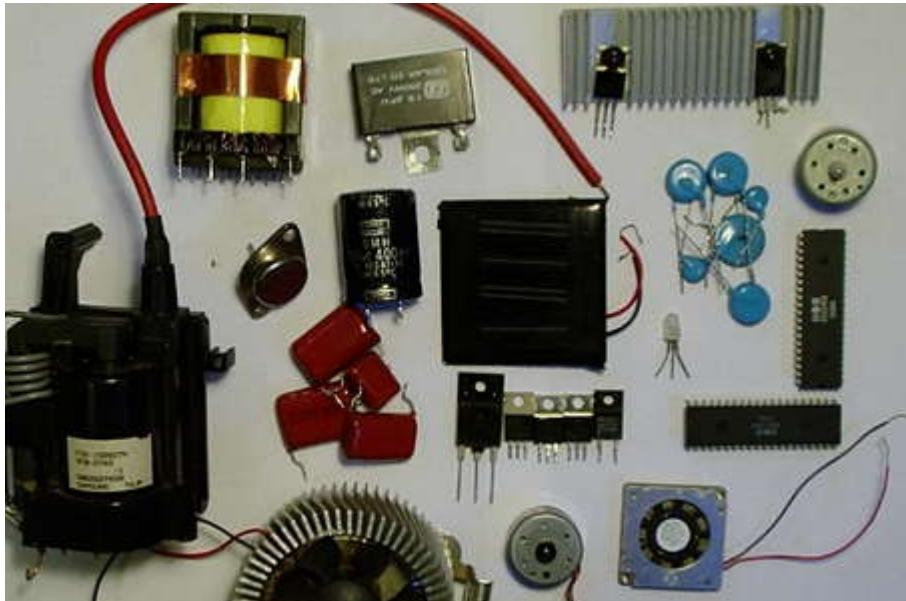


samples (muestras): Texas Instruments y Maxim

- Lista de **fabricantes** que proporcionan **samples** de **instructables**
- Fabricantes que proporcionan samples segun ladyada
- En **hack a day** han recopilado también fuentes



Reciclando hardware: Todo gratis



<http://www.instructables.com/id/How-To-Get-FREE-Electronic-parts/?ALLSTEPS>

<http://blog.elcacharreo.com/2012/04/13/de-donde-obtener-material-electronico-gratis/>



Fuentes

arduino

arduino programing notebook

freeduino



Conclusiones

Gracias por vuestra atención

