



# **Taller Arduino para Dummies**

**Hackers Week  
ETSI Informática  
Málaga, 4 de Abril 2017**

**Ana Cruz Martín  
@WeekendArchaeologist**

# Introducción a Arduino

## Índice

- ¿Qué es Arduino?
- Instalación del software. GUI
- Primer ejemplo: parpadeo de un LED
- Programación más avanzada
- Extracción de datos y representación gráfica
- Bibliografía
- Recursos complementarios
- Contacto

# Introducción a Arduino

## Índice

- ¿Qué es Arduino?
- Instalación del software. GUI
- Primer ejemplo: parpadeo de un LED
- Programación más avanzada
- Extracción de datos y representación gráfica
- Bibliografía
- Recursos complementarios
- Contacto

# Introducción a Arduino

¿Qué es Arduino?

- “Arduino is an open-source electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects.”
- Diferentes modelos, nosotros vamos a usar el Arduino UNO
  - Extensibles mediante shields (ethernet, wifi, pantallas...)
  - Kits (iniciación, wearable...)



# Introducción a Arduino

## ¿Qué es Arduino?



**Arduino UNO Rev3**

The board everybody gets started with, based on the ATmega328.

€20.00 + VAT

[ADD TO CART](#)



**Arduino Leonardo with Headers**

Similar to an Arduino UNO, can be recognized by computer as a mouse or keyboard.

€18.00 + VAT

[ADD TO CART](#)



**Arduino YÚN**

Arduino with onboard Wi-Fi connectivity and a Linux computer. Great for IoT projects.

€52.00 + VAT

[ADD TO CART](#)



**Arduino YÚN with PoE**

Arduino YÚN that can be powered through the ethernet cable.

€61.00 + VAT

[ADD TO CART](#)



**Arduino DUE**

The evolution of the Arduino Mega, with more memory and a more powerful processor (ARM Cortex-M3).

€36.00 + VAT

[OUT OF STOCK](#)



**Arduino Mega 2560 Rev3**

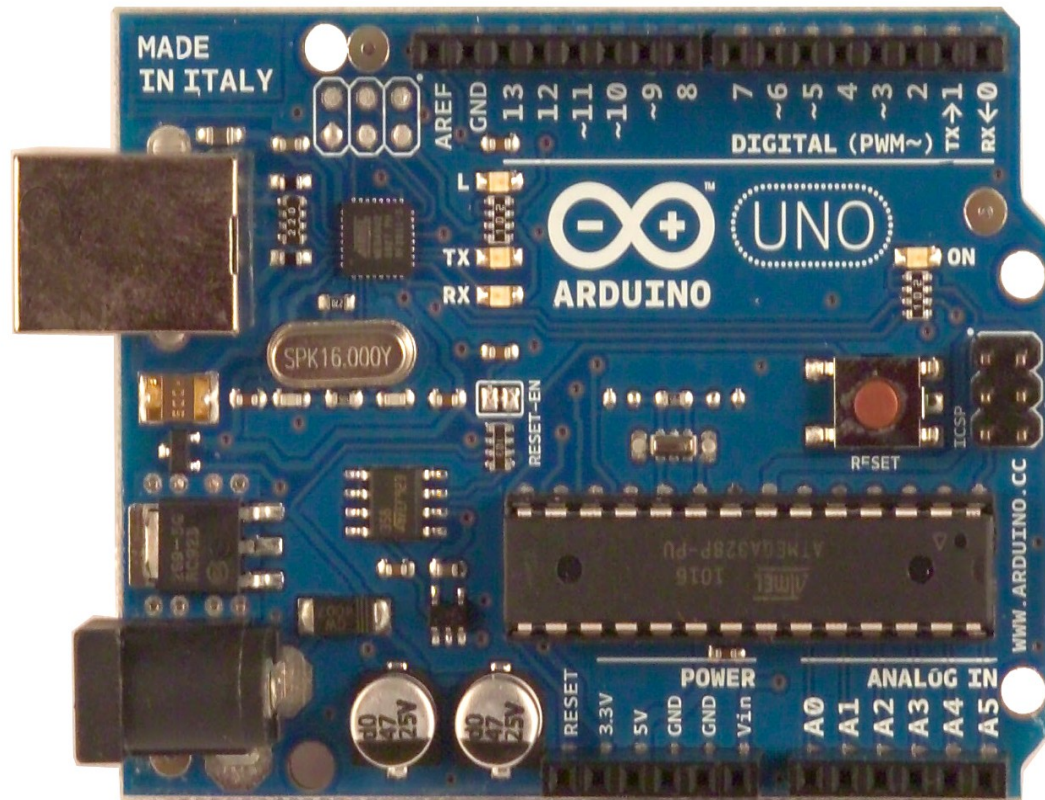
Arduino able to control many components, designed for complex projects, based on the ATmega2560.

€35.00 + VAT

[ADD TO CART](#)

# Introducción a Arduino

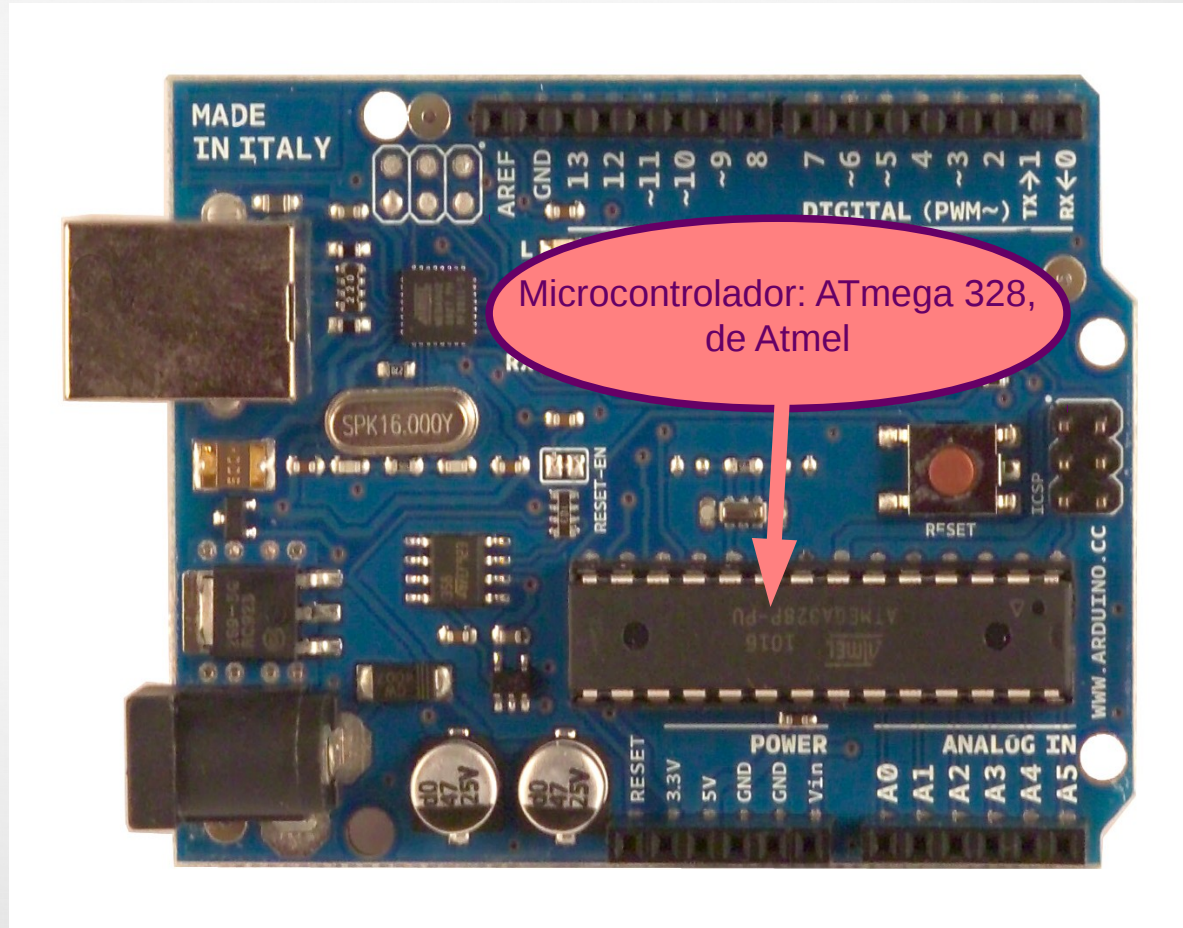
¿Qué es Arduino?





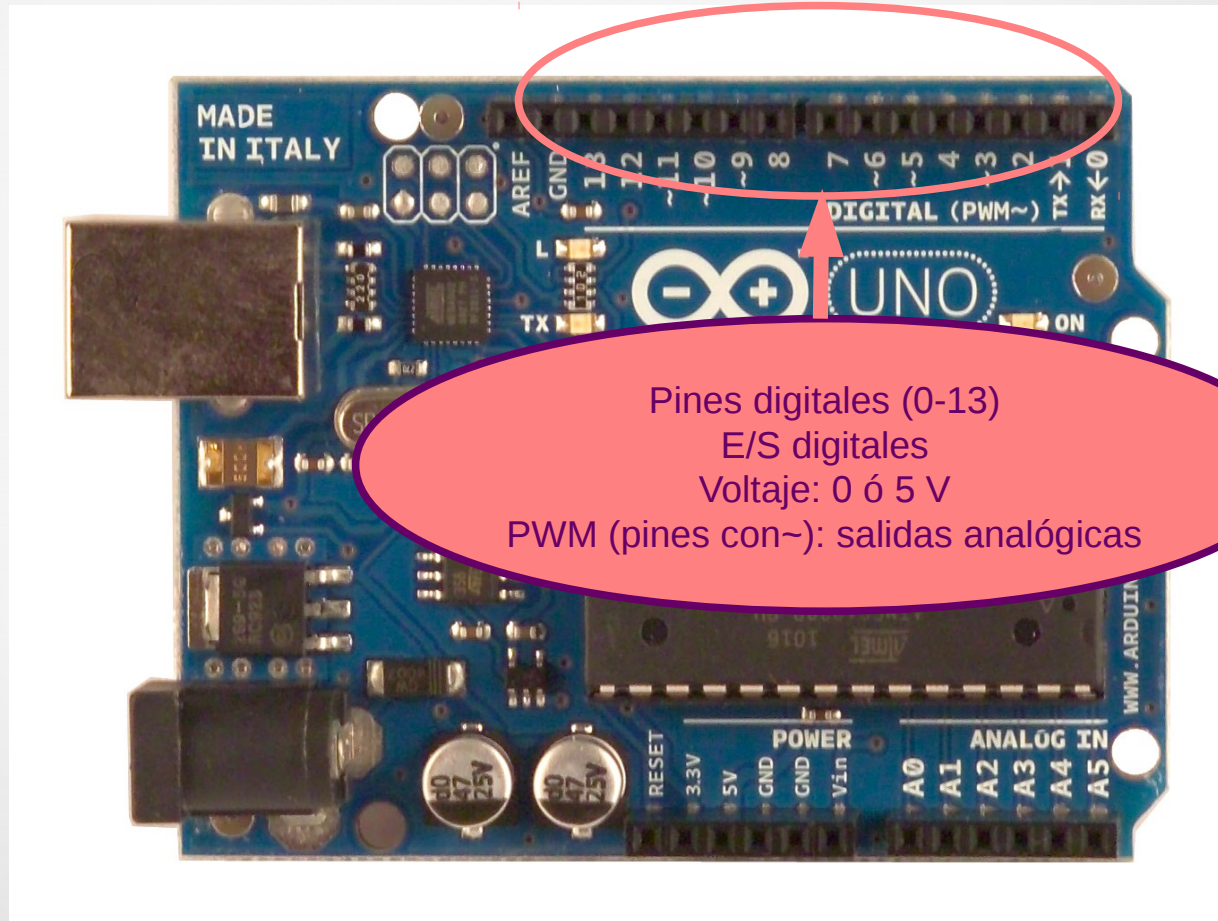
# Introducción a Arduino

¿Qué es Arduino?



# Introducción a Arduino

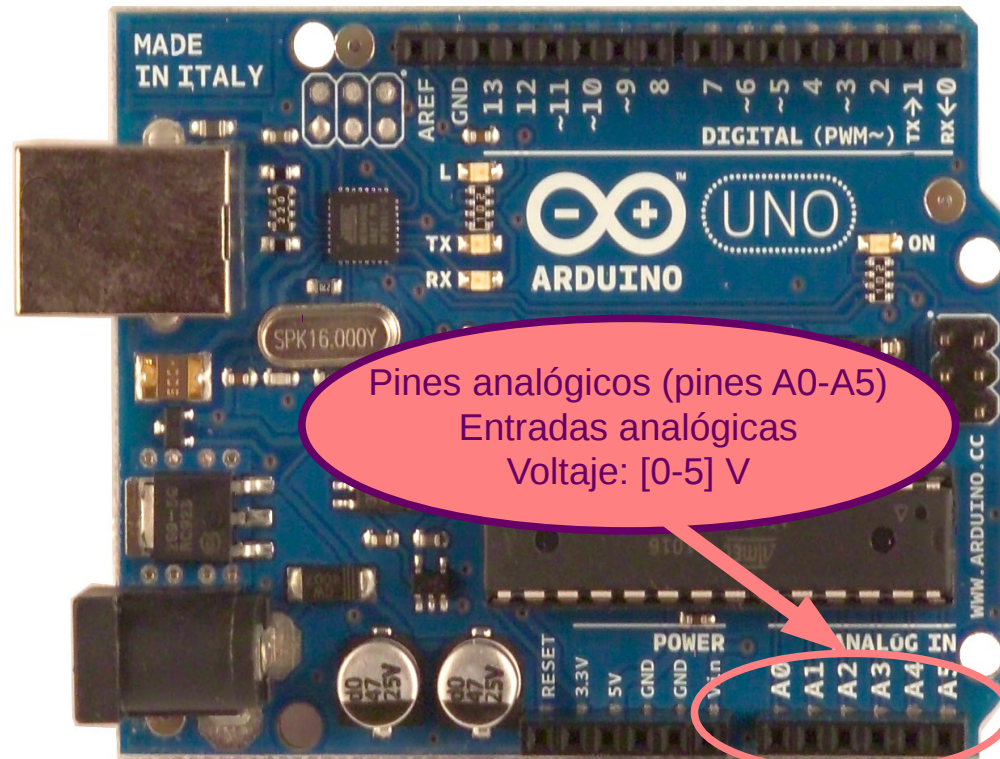
¿Qué es Arduino?





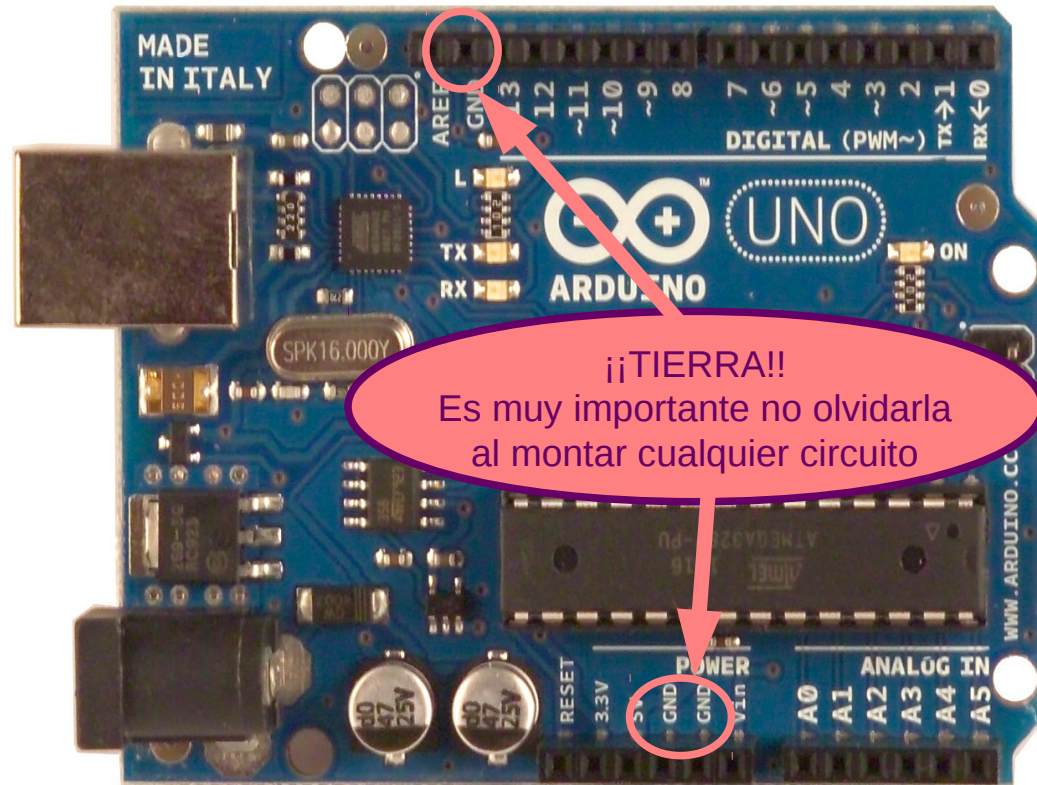
# Introducción a Arduino

¿Qué es Arduino?



# Introducción a Arduino

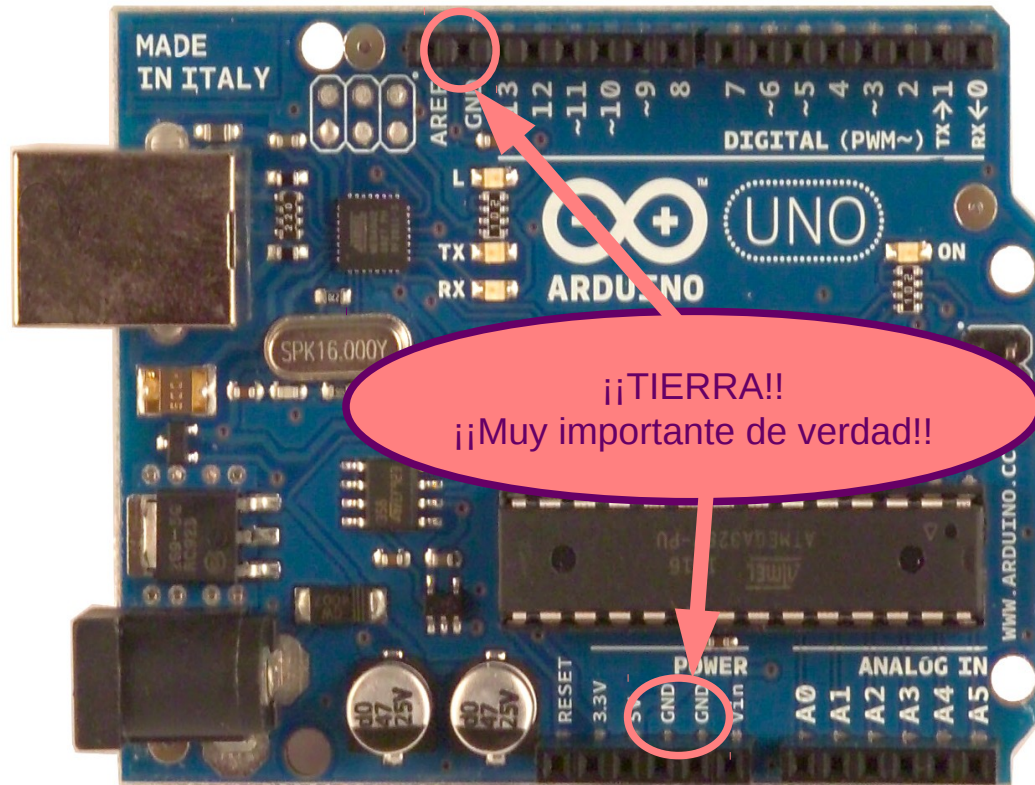
¿Qué es Arduino?





# Introducción a Arduino

¿Qué es Arduino?





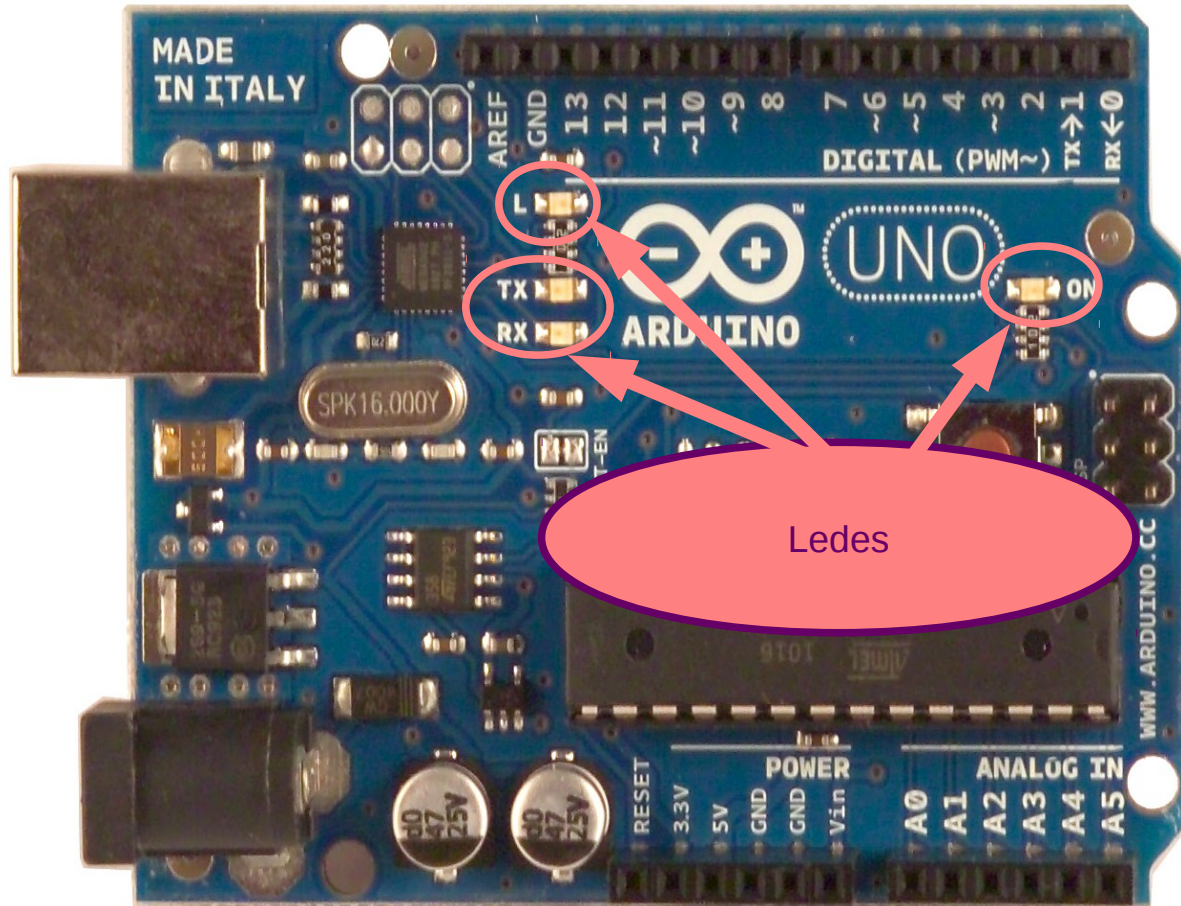
# Introducción a Arduino

¿Qué es Arduino?



# Introducción a Arduino

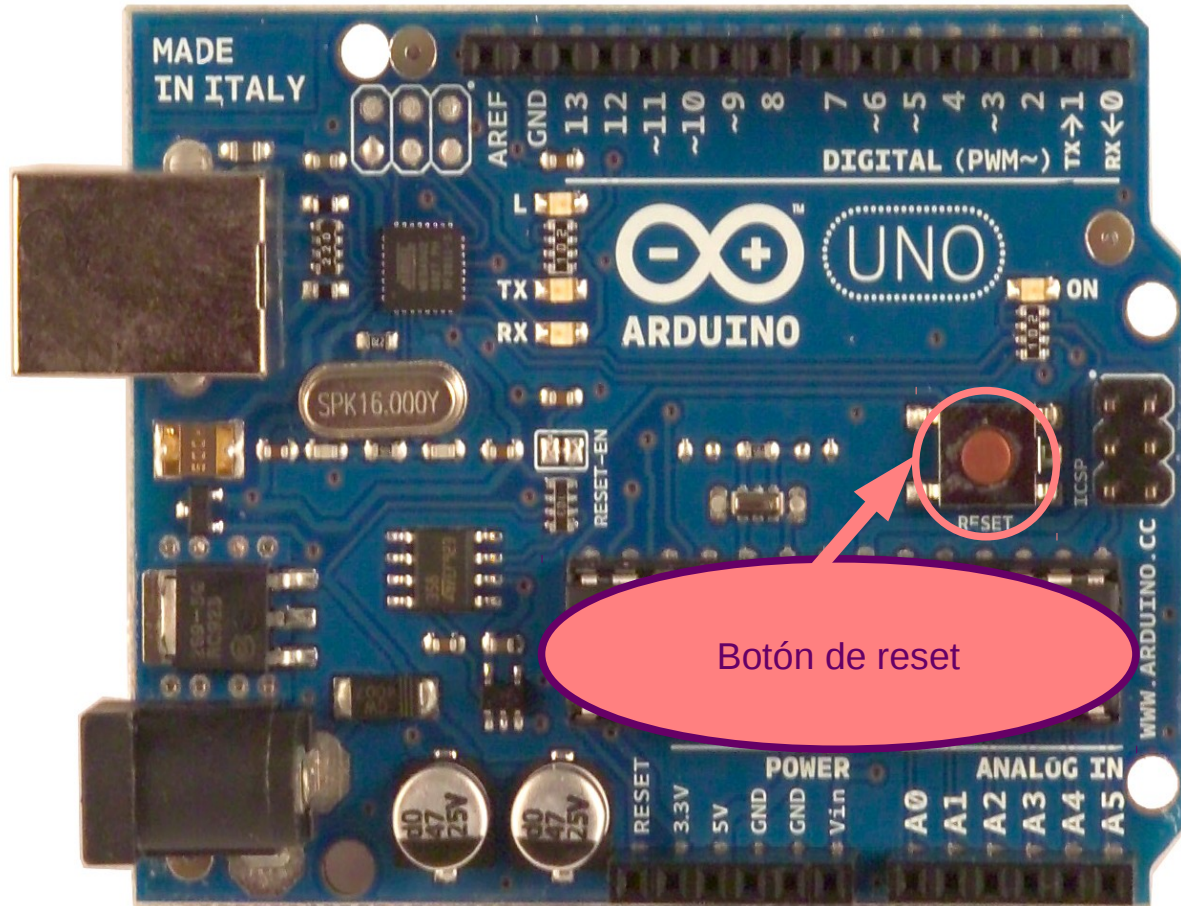
Arduino





# Introducción a Arduino

Arduino



Botón de reset



# Introducción a Arduino

## Índice

- ¿Qué es Arduino?
- [Instalación del software. GUI](#)
- Primer ejemplo: parpadeo de un LED
- Programación más avanzada
- Extracción de datos y representación gráfica
- Bibliografía
- Recursos complementarios
- Contacto

# Introducción a Arduino

## Instalación del software. GUI

- Página oficial de Arduino: [www.arduino.cc](http://www.arduino.cc)
  - Menú *Learning*
- Windows y Mac OS: *Arduino for Dummies*
  - Part I: Getting to know Arduino, pág. 34
- Xubuntu 14.04
  - Instalar desde Synaptic *arduino* y *arduino-core*
  - Add al *dialout* group

# Introducción a Arduino

## Instalación del software. GUI

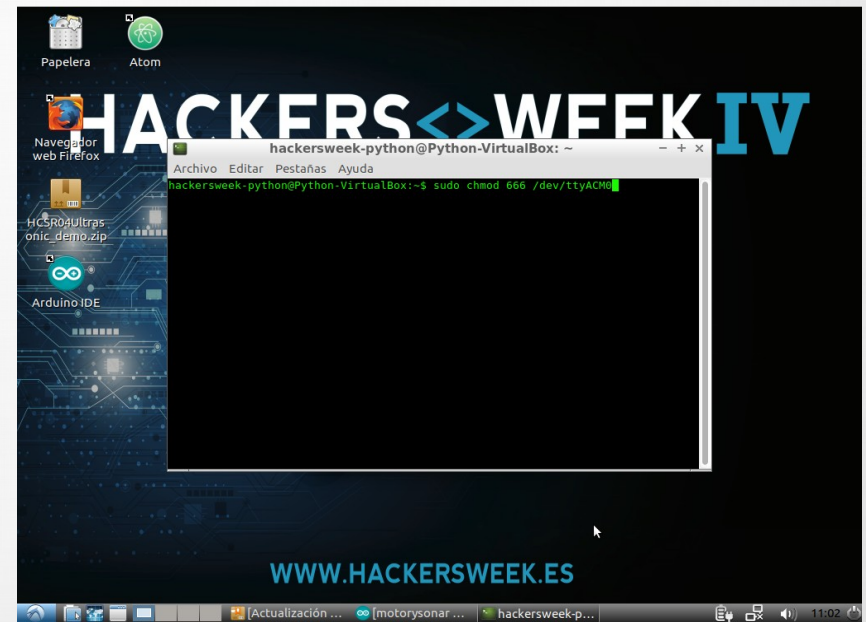
- Pasos para trabajar con la máquina virtual del Taller
  - Comprobar que el Arduino está conectado
    - Arrancar la máquina virtual
    - Menú Dispositivos → Dispositivos USB → Activar Arduino que aparece
  - Dar permisos al terminal serie
    - Herramientas del sistema → LX Terminal
    - `sudo chmod 666 /dev/ttyACM0`



# Introducción a Arduino

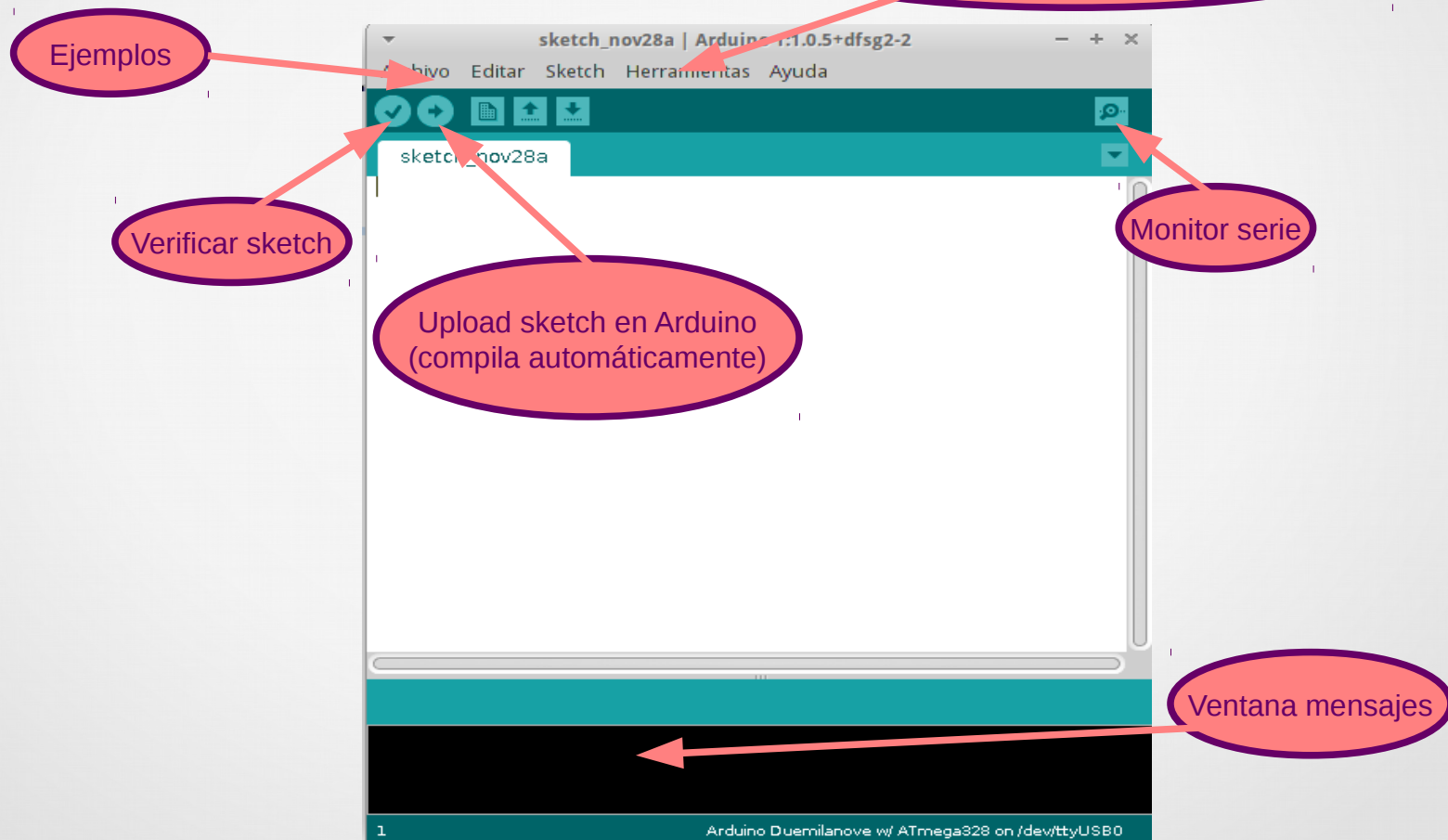
## Instalación del software. GUI

- Pasos para trabajar con la máquina virtual del Taller
  - Dar permisos al terminal serie
    - Herramientas del sistema → LX Terminal
    - `sudo chmod 666 /dev/ttyACM0`



# Introducción a Arduino

## Instalación del software. GUI



# Introducción a Arduino

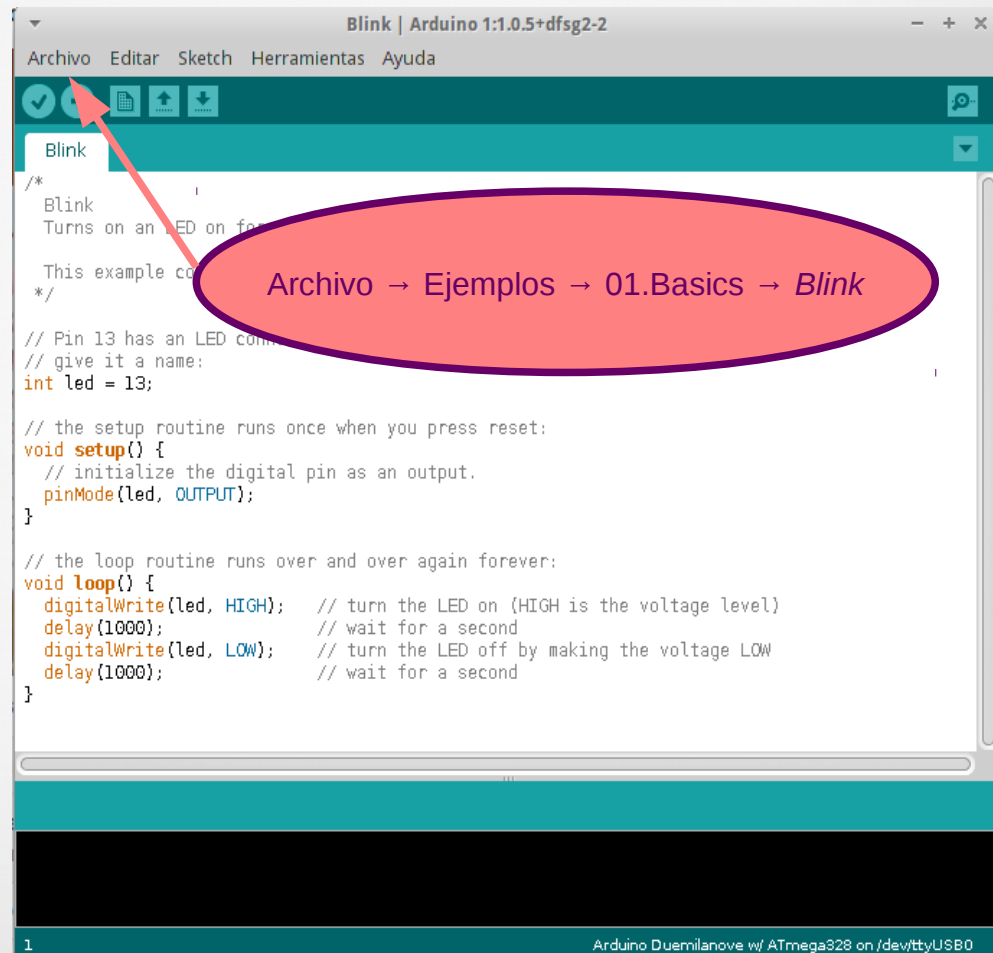
## Índice

- ¿Qué es Arduino?
- Instalación del software. GUI
- **Primer ejemplo: parpadeo de un LED**
- Programación más avanzada
- Extracción de datos y representación gráfica
- Bibliografía
- Recursos complementarios
- Contacto



# Introducción a Arduino

Primer ejemplo: parpadeo de un LED



# Introducción a Arduino

## Primer ejemplo: parpadeo de un LED

Usamos el pin digital 13

### loop

Se ejecuta continuamente  
Ahí se incluye el código de nuestro programa

```
Blink | Arduino 1:1.0.5+dfsg2-2
Archivo  Editar  Sketch  Herramientas  Ayuda

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

  This example code is in the public domain.
  */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

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

### setup

Se ejecuta al principio, una sólo vez  
En este caso, configura el pin 13 como salida

### Programa

Envía 5V al pin 13  
Espera un segundo  
Envía 0V al pin 13  
Espera un segundo

# Introducción a Arduino

Primer ejemplo: parpadeo de un LED

¡¡Atención!!

- El GUI “ensucia” mucho el código que se pasa al Arduino
  - Para crear algunas de las “ayudas” (por ejemplo, la función *setup*) utiliza temporizadores, etc. que dejan de estar disponibles para el programador.
  - AVR Studio: <http://www.avr-tutorials.com/avr-studio-6/avr-studio-6>
- ¡¡Nunca conectar directamente un LED al pin 13 del Arduino!!
  - Ejemplo pág. 58 en *Arduino for Dummies*
  - Explicación:  
<http://electronics.stackexchange.com/questions/66992/pin-13-do-i-need-a-resistor>
  - Cualquier LED debe ir **SIEMPRE** protegido por una resistencia



# Introducción a Arduino

## Índice

- ¿Qué es Arduino?
- Instalación del software. GUI
- Primer ejemplo: parpadeo de un LED
- [Programación más avanzada](#)
- Extracción de datos y representación gráfica
- Bibliografía
- Recursos complementarios
- Contacto

# Introducción a Arduino

Programación más avanzada

## 1) Señal SOS

- Ejercicio: modificar el programa (y guardarlo aparte) para que el LED reproduzca la señal de SOS

# Introducción a Arduino

Programación más avanzada

## 2) Trabajando con sensores y actuadores:

- Sensores: elementos que nos permiten percibir el entorno
- Sensores para Arduino (<http://tienda.bricogeek.com/23-sensores>).
- Documento fundamental: hoja de características de los mismos (*datasheets*).
  - Características y limitaciones de conexionado y uso.
  - A veces, explicaciones detalladas de su funcionamiento.



# Introducción a Arduino

Programación más avanzada

## 2) Trabajando con sensores y actuadores:

### Sónar HC-SR04



- Los sónares nos permiten medir la distancia a la que se encuentra un objeto situado delante de él.
- Para realizar su tarea emite un sonido y mide el tiempo transcurrido hasta que recibe el eco.
- Es fácil derivar la distancia a la que se encuentra un objeto si se conoce dicho tiempo y la velocidad a la que se transmite el sonido.
- El factor de conversión es proporcionado típicamente por la hoja de especificaciones o *datasheet*.

# Introducción a Arduino

Programación más avanzada

2) Trabajando con sensores y actuadores:

Sónar HC-SR04:



Datasheet: <http://www.micropik.com/PDF/HCSR04.pdf>

# Introducción a Arduino

Programación más avanzada

## 2) Trabajando con sensores y actuadores:

### Sónar HC-SR04

- Información importante que podemos extraer de la hoja de especificaciones
  - Rango
  - Precisión
  - Voltaje
  - Corriente
  - Frecuencia
  - Temporización

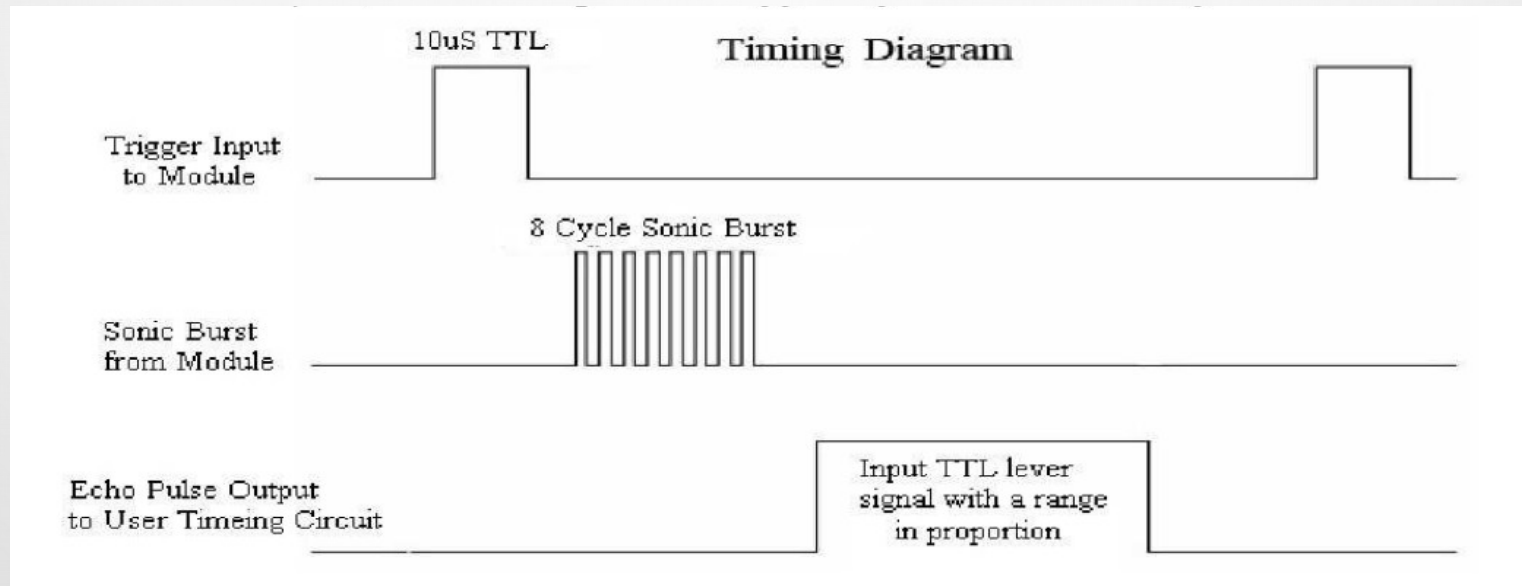


# Introducción a Arduino

Programación más avanzada

2) Trabajando con sensores y actuadores:

Sónar HC-SR04:



Datasheet: <http://www.micropik.com/PDF/HCSR04.pdf>

# Introducción a Arduino

## Programación más avanzada

### 2) Trabajando con sensores y actuadores:



```
ejemplo_sonar Arduino 1.8.1
Archivo Editar Programa Herramientas Ayuda

ejemplo_sonar

/*
  HC-SR04 Ping distance sensor

  VCC al pin 5v de arduino
  GND al pin GND de arduino
  Echo al pin 13 de arduino
  Trig al pin 12 de arduino

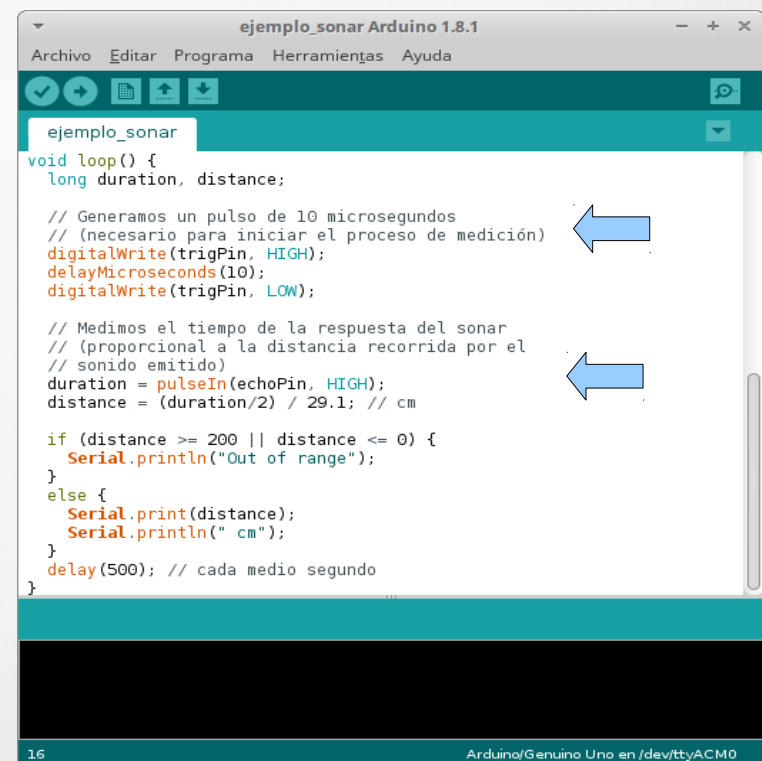
  */

#define trigPin 13
#define echoPin 12

void setup() {
  // Inicializamos el monitor serie para mostrar la distancia (cm)
  Serial.begin (9600);

  // Configuramos:
  // el pin 13 como salida (disparo) y
  // el pin 12 como entrada (echo)
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}
```

37 Arduino/Genuino Uno en /dev/ttyACM0



```
ejemplo_sonar Arduino 1.8.1
Archivo Editar Programa Herramientas Ayuda

ejemplo_sonar

void loop() {
  long duration, distance;

  // Generamos un pulso de 10 microsegundos
  // (necesario para iniciar el proceso de medición)
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Medimos el tiempo de la respuesta del sonar
  // (proporcional a la distancia recorrida por el
  // sonido emitido)
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1; // cm

  if (distance >= 200 || distance <= 0) {
    Serial.println("Out of range");
  }
  else {
    Serial.print(distance);
    Serial.println(" cm");
  }
  delay(500); // cada medio segundo
}
```

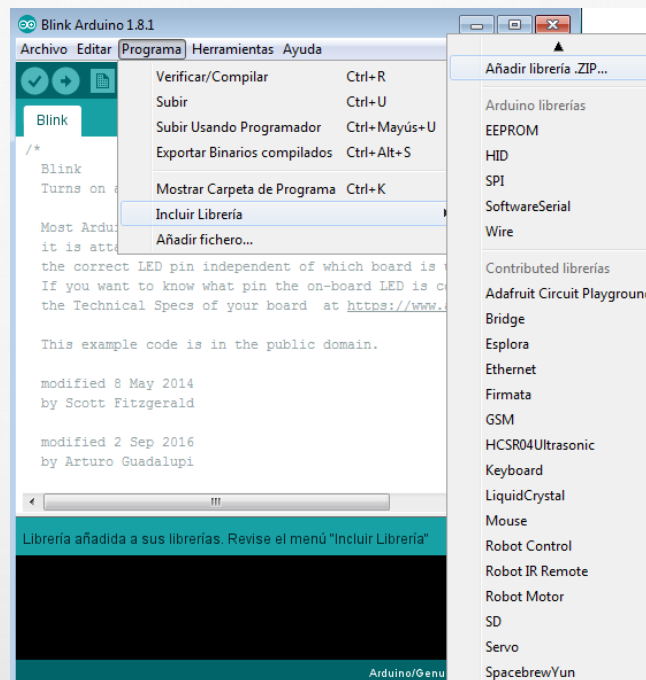
16 Arduino/Genuino Uno en /dev/ttyACM0

# Introducción a Arduino

Programación más avanzada

2) Trabajando con sensores y actuadores:

b) Otra forma de trabajar con el sónar HC-SR04: usar su librería



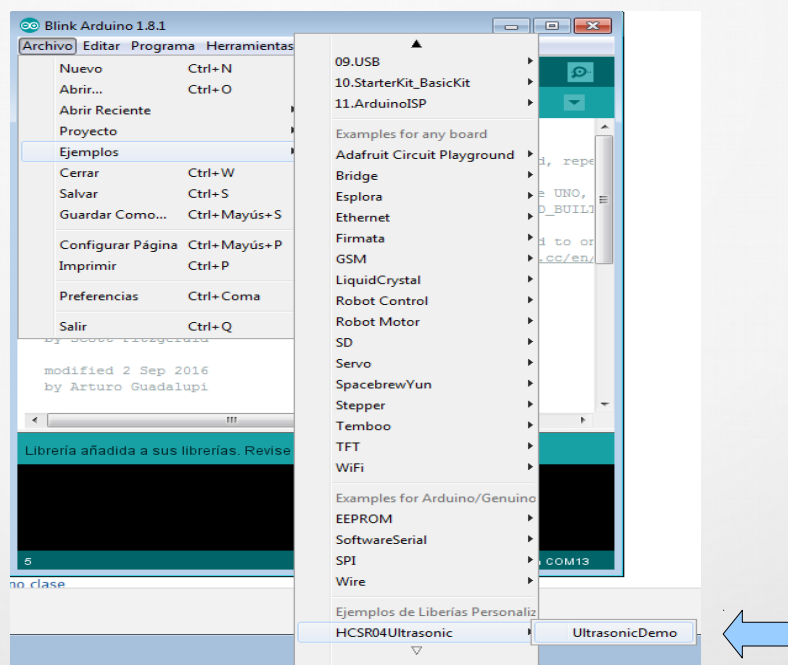


# Introducción a Arduino

Programación más avanzada

2) Trabajando con sensores y actuadores:

b) Otra forma de trabajar con el sónar HC-SR04: usar su librería



# Introducción a Arduino

Programación más avanzada

## 2) Trabajando con sensores y actuadores:

b) Otra forma de trabajar con el sónar HC-SR04: usar su librería

```
/* UltrasonicDemo.pde - Ultrasonic sensor Ranging measure Demo
 * @Author: dragon
 * @DATA: 2013-8-7
 * Company website: www.electronicsfreaks.com
 */
#include <Ultrasonic.h>

int trigpin = 2; // appoint trigger pin
int echopin = 3; // appoint echo pin

Ultrasonic ultrasonic(trigpin, echopin);

void setup()
{
    Serial.begin(9600); // set Serial Baud rate
    Serial.println("Ultrasonic sensor starting!!!!");
}

void loop()
{
    float cmdistance, indistance;
    long microsec = ultrasonic.timing();
    Serial.print("microsec: ");
    Serial.print(microsec);
    cmdistance = ultrasonic.CalculateDistance(microsec, Ultrasonic::CM); // this result unit is cm
    indistance = ultrasonic.CalculateDistance(microsec, Ultrasonic::IN); // this result unit is in
    Serial.print("cmdistance: ");
    Serial.print(cmdistance);
    Serial.print(" indistance: ");
    Serial.println(indistance);
    delay(1000);
}

1

Arduino Uno on /dev/ttyACM0
```

# Introducción a Arduino

Programación más avanzada

## 2) Trabajando con sensores y actuadores:

### b) Servomotor SG90 9g:

- Un servomotor es un dispositivo capaz de situarse en cualquier posición dentro de rango de operación y de mantenerse estable en dicha posición.
- La posición del servomotor es controlada con una señal PWM (Pulse Width Modulated). Más concretamente, por la anchura del ciclo de carga (duty cycle) de dicha señal.
- El rango de operación, par, precisión, etc. viene dado por el *datasheet* del dispositivo.
- Para trabajar con servomotores Arduino cuenta con una librería C que facilita mucho el trabajo (Servo.h).



# Introducción a Arduino

Programación más avanzada

2) Trabajando con sensores y actuadores:

b) Servomotor SG90 9g:



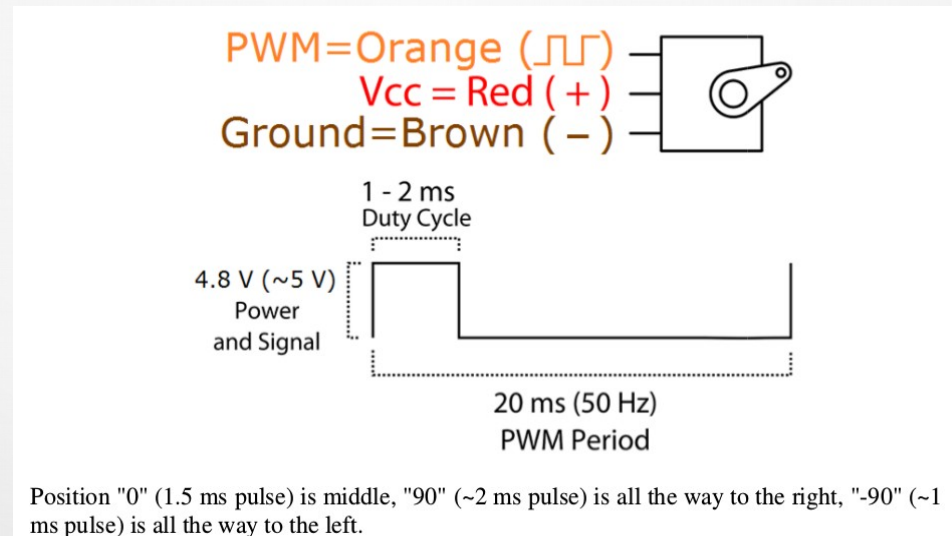
Datasheet: <http://www.micropik.com/PDF/SG90Servo.pdf>

# Introducción a Arduino

Programación más avanzada

## 2) Trabajando con sensores y actuadores:

### b) Servomotor SG90 9g:



*Datasheet:* <http://www.micropik.com/PDF/SG90Servo.pdf>

# Introducción a Arduino

## Programación más avanzada

### 2) Trabajando con sensores y actuadores:

#### b) Servomotor SG90 9g:



```
ejemplo_servo Arduino 1.8.1
Archivo Editar Programa Herramientas Ayuda

ejemplo_servo

/*
  Microservo SG90

  VCC (red) al pin 5v de arduino
  GND (brown) al pin GND de arduino
  PWM (orange) al pin 9 de arduino
*/

// Incluimos la librería para poder controlar el servo
#include <Servo.h>

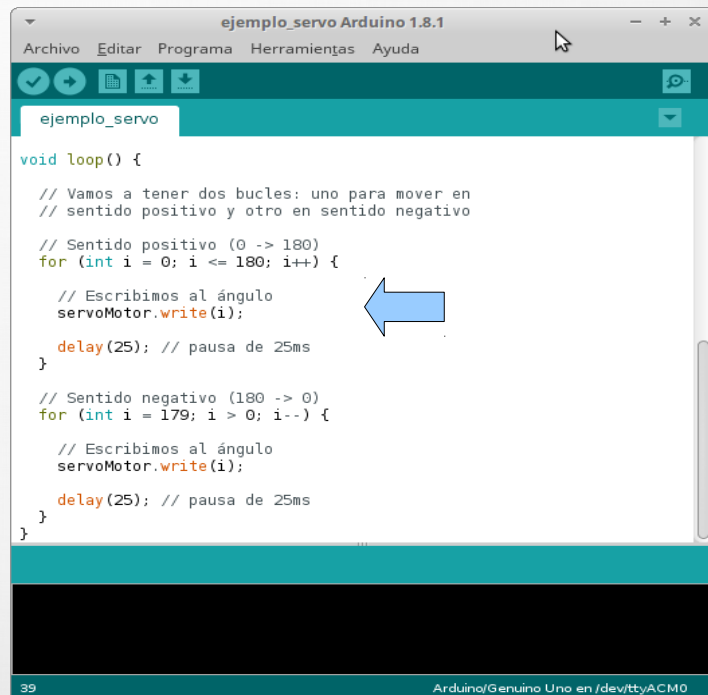
#define pwmPin 10

// Declaramos la variable (global) para controlar el servo
Servo servoMotor;

void setup() {
  // Asociamos el pin 10 (PWM) al servo
  servoMotor.attach(pwmPin);

  // Fijamos el ángulo del servomotor en 0 grados
  servoMotor.write(0);
}
```

Arduíno/Genuíno Uno en /dev/ttyACM0



```
ejemplo_servo Arduino 1.8.1
Archivo Editar Programa Herramientas Ayuda

ejemplo_servo

void loop() {
  // Vamos a tener dos bucles: uno para mover en
  // sentido positivo y otro en sentido negativo

  // Sentido positivo (0 -> 180)
  for (int i = 0; i <= 180; i++) {
    // Escribimos al ángulo
    servoMotor.write(i);
    delay(25); // pausa de 25ms
  }

  // Sentido negativo (180 -> 0)
  for (int i = 179; i > 0; i--) {
    // Escribimos al ángulo
    servoMotor.write(i);
    delay(25); // pausa de 25ms
  }
}
```

Arduíno/Genuíno Uno en /dev/ttyACM0



# Introducción a Arduino

Programación más avanzada

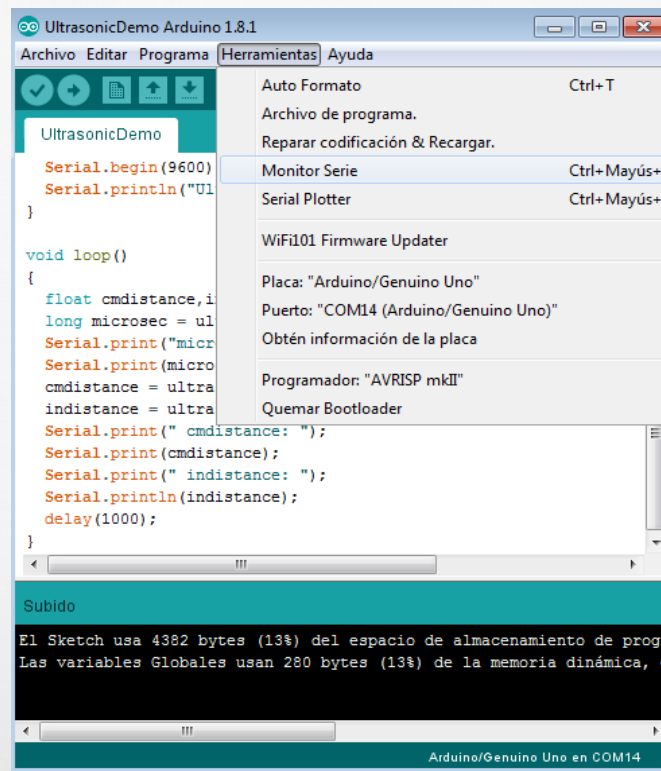
## 3) Combinar sónar y motor

- Crear un programa que, en función de las medidas del sónar, mueva el motor a un rango de posiciones (en grados) concretas.
  - Usar para el cálculo de la distancia la expresión basada en la longitud del tiempo de eco (consultar la hoja de especificaciones).
  - El máximo de distancia del sónar que vamos a considerar serán 200 cm.
  - En función de la distancia (se pueden crear, por ejemplo, 4 rangos de distancia), se mostrará por pantalla un mensaje de alarma, y el led de Arduino parpadeará más o menos rápido.
- Ejemplo similar:
  - <https://www.youtube.com/watch?v=7AMZmJNDnKs>

# Introducción a Arduino

Programación más avanzada

## 4) Recuperación sencilla de datos y representación gráfica en Matlab



**Abrir Monitor Serie**

# Introducción a Arduino

Programación más avanzada

## 4) Recuperación sencilla de datos y representación gráfica en Matlab

The image shows a Windows desktop with two windows open. The top window is the Arduino IDE's serial monitor, titled 'COM14 (Arduino/Genuino Uno)'. It displays a list of data points received from the Arduino, such as '3510.77 193958', '3511.04 193958', etc. A blue arrow points to the text '1) Monitor serie: copiar datos con Ctrl-C' overlaid on the serial monitor. The bottom window is a 'Bloc de notas' (Notepad) titled 'datossonar: Bloc de notas'. It contains the same data points copied from the serial monitor. A blue arrow points to the text '2) Pegar datos en fichero de texto y guardar en Y:\' overlaid on the Notepad window. The desktop background shows icons for AutoCAD 2011, MATLAB R2010b, AutoCAD 2013, and Measurement & Automation.

1) Monitor serie:  
copiar datos  
con Ctrl-C

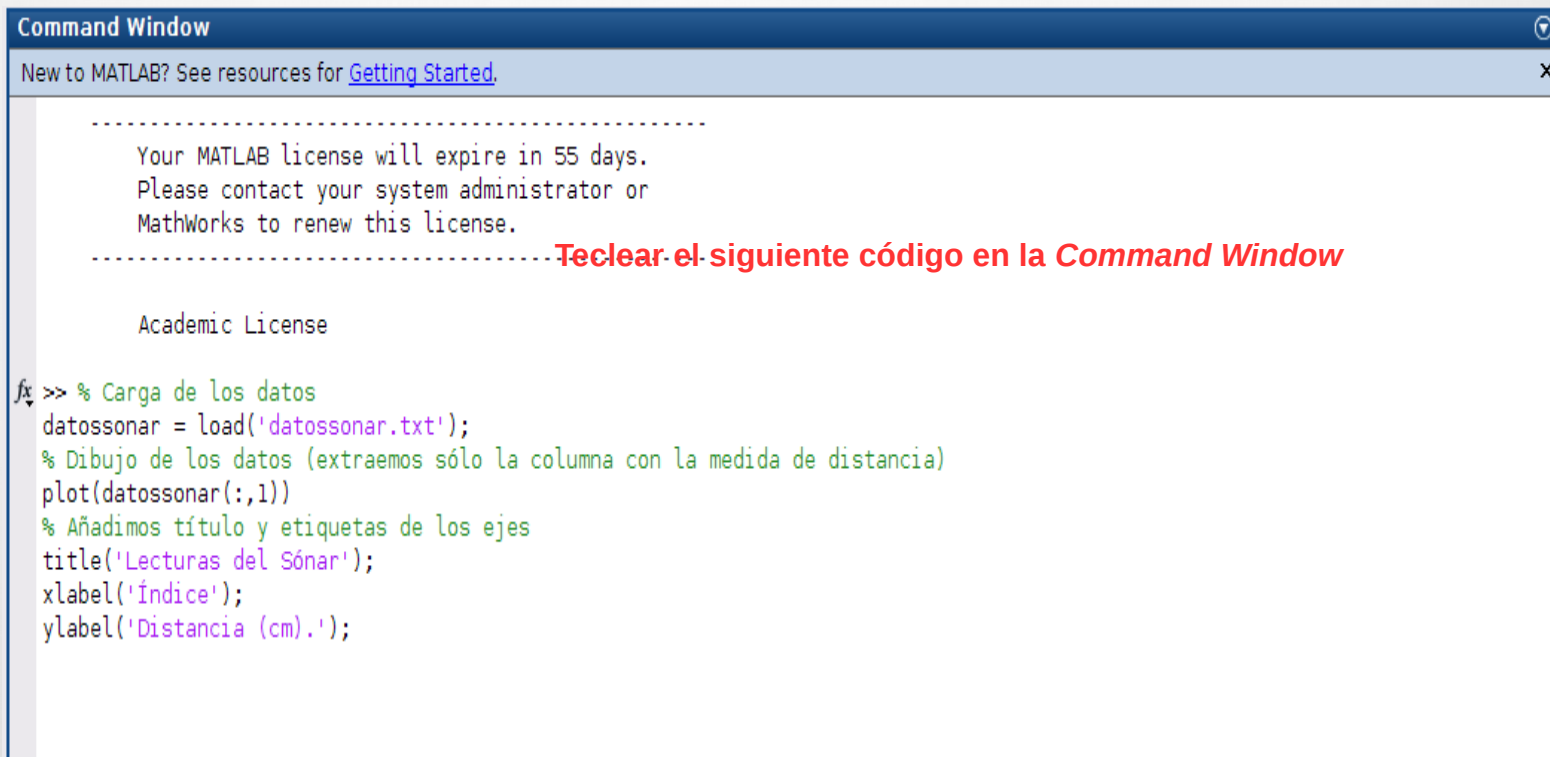
2) Pegar datos en fichero de  
texto y guardar en Y:\



# Introducción a Arduino

Programación más avanzada

## 4) Recuperación sencilla de datos y representación gráfica en Matlab



```
Command Window
New to MATLAB? See resources for Getting Started.

-----
Your MATLAB license will expire in 55 days.
Please contact your system administrator or
MathWorks to renew this license.
-----

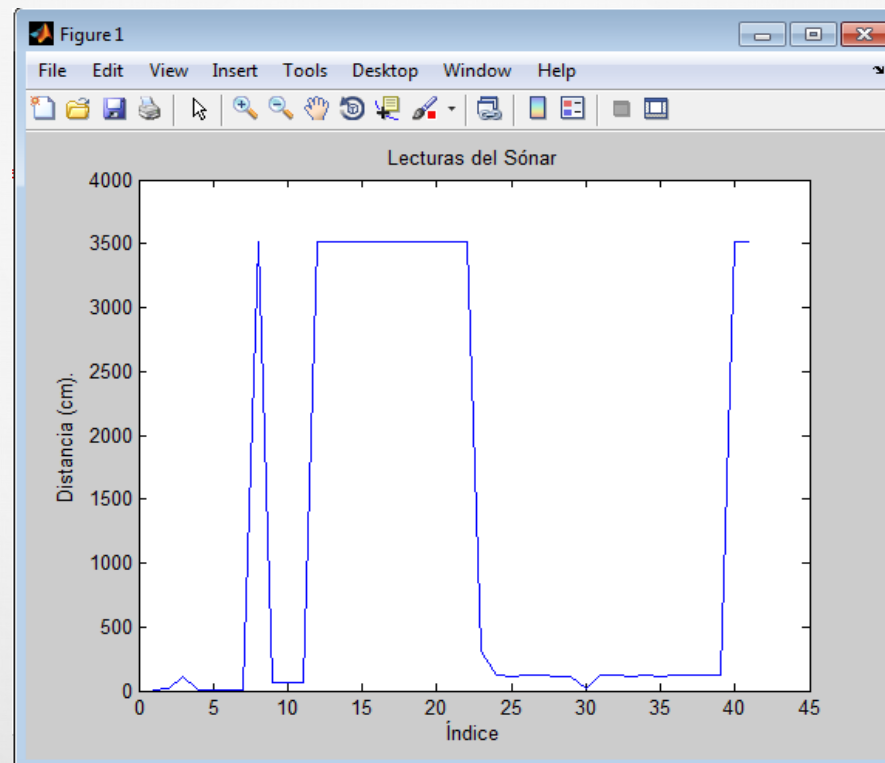
Academic License

fx >> % Carga de los datos
datossonar = load('datossonar.txt');
% Dibujo de los datos (extraemos sólo la columna con la medida de distancia)
plot(datossonar(:,1))
% Añadimos título y etiquetas de los ejes
title('Lecturas del Sónar');
xlabel('Índice');
ylabel('Distancia (cm).');
```

# Introducción a Arduino

Programación más avanzada

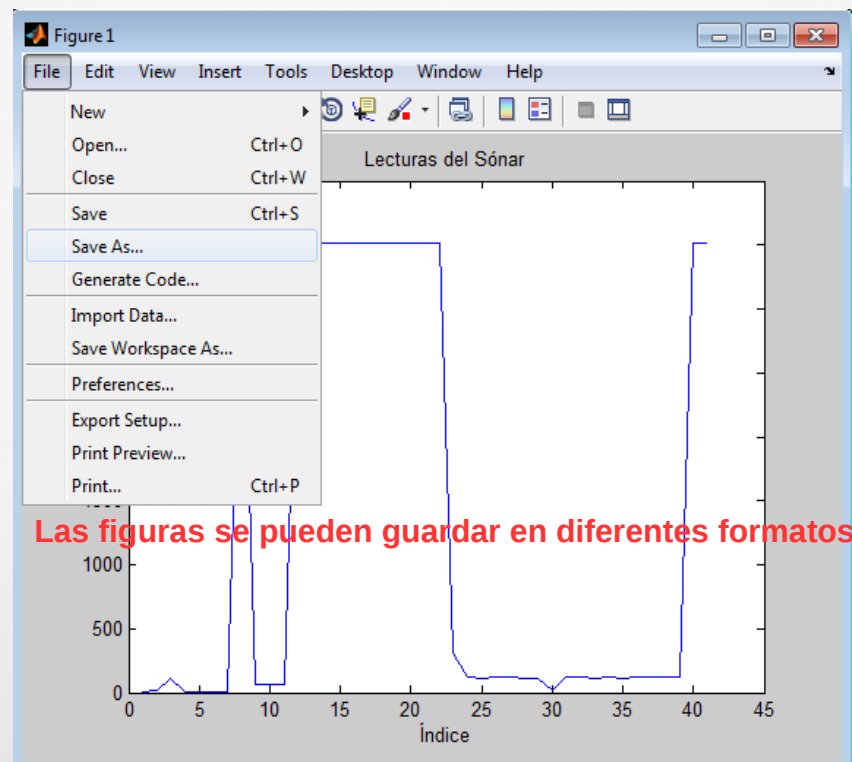
## 4) Recuperación sencilla de datos y representación gráfica en Matlab



# Introducción a Arduino

Programación más avanzada

## 4) Recuperación sencilla de datos y representación gráfica en Matlab



# Introducción a Arduino

## Bibliografía

### Básica

- Página web de Arduino (2017) <http://www.arduino.cc/>
- Torrente Artero, O. (2013) *Arduino. Curso práctico de formación*, RC Libros.



# Introducción a Arduino

## Bibliografía

### Avanzada

- Anderson A., Cervo D. (2013) *Pro Arduino*, Apress.
- Blum J. (2013) *Exploring Arduino: tools and techniques for engineering wizardry*, Wiley & Sons.
- Dennis A. K. (2013) *Raspberry Pi Home Automation with Arduino*, Packt Publishing.
- Goransson A., David Cuartielles Ruiz D. (2013) *Professional Android Open Accessory Programming with Arduino*, Wrox.
- Nussey J. (2013) *Arduino for Dummies*, Wiley & Sons.
- Olsson T. (2012) *Arduino Wearables*, Apress.
- Ramos Melgar E., Castro Díez C., Jaworski P. (2012) *Arduino and Kinect Projects. Design, Build, Blow Their Minds*, Apress.
- Wilcher D. (2012) *Learn Electronics with Arduino*, Apress.

# Introducción a Arduino

## Recursos complementarios

- Octave: software matemático de código abierto
  - [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/)
  - [es.wikipedia.org/wiki/GNU\\_Octave](http://es.wikipedia.org/wiki/GNU_Octave)
- Autodesk circuits: simulador de circuitos
  - [circuits.io/](http://circuits.io/)
  - Ejemplo S.O.S: [circuits.io/circuits/895430-s-o-s-signal](http://circuits.io/circuits/895430-s-o-s-signal)

# Introducción a Arduino

## Contacto

Ana Cruz Martín

[anacm@ctima.uma.es](mailto:anacm@ctima.uma.es)

[@WeekendArchaeologist](https://twitter.com/WeekendArchaeologist)

[twa.anacruzmartin.com](http://twa.anacruzmartin.com)

# Introducción a Arduino

¡¡MUCHAS GRACIAS POR VUESTRA ATENCIÓN :)!!

¿ALGUNA PREGUNTA?

[anacm@ctima.uma.es](mailto:anacm@ctima.uma.es)  
[@WeekendArchaeologist](#)  
[twa.anacruzmartin.com](http://twa.anacruzmartin.com)