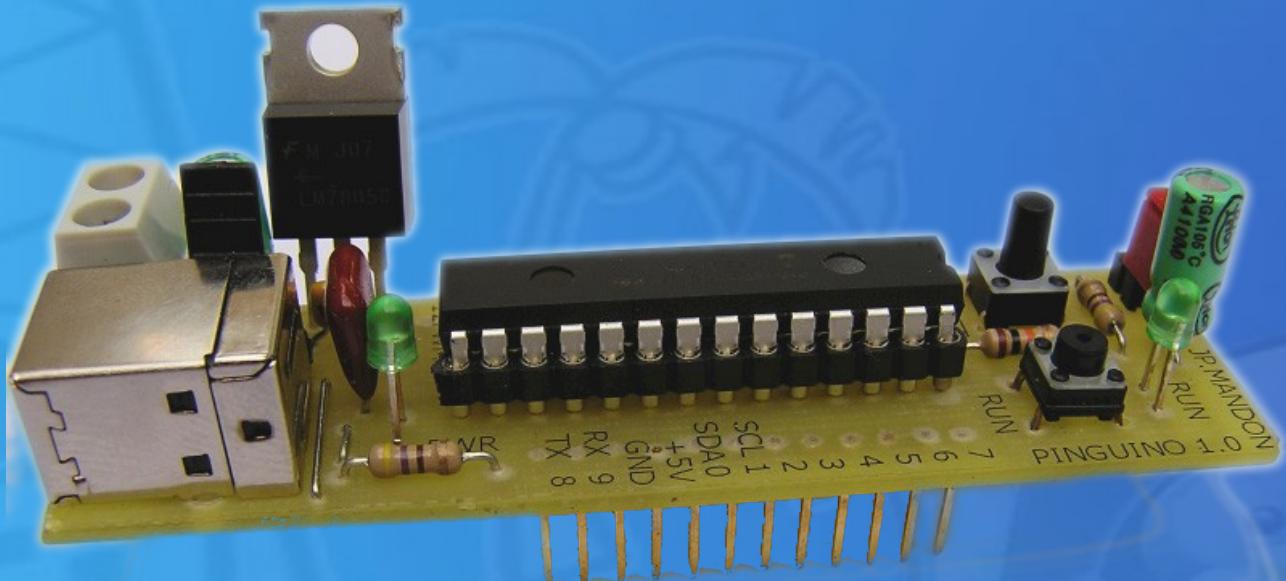
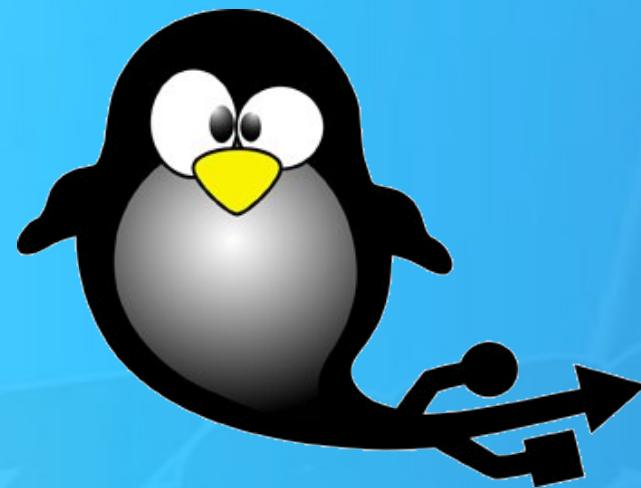




# PINGUINO

## Introducción al Hardware Libre

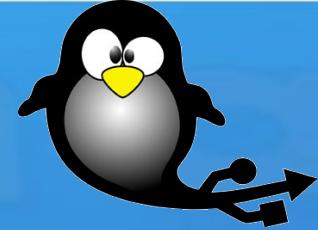


Presentado por:  
TSU Oswaldo Hernández / @cumanadigital



# PINGUINO

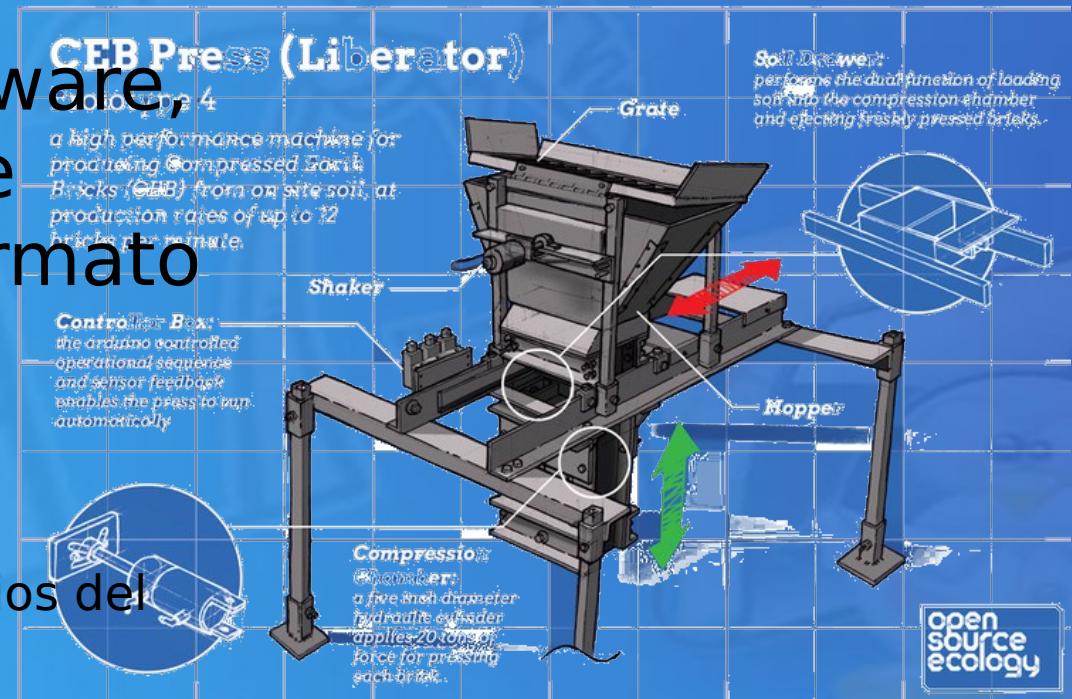
## Introducción al Hardware Libre



### Hardware Libre

Es hardware cuyo diseño está disponible al público de forma que cualquiera puede estudiarlo, modificarlo, distribuirlo, construirlo y vender tanto el diseño como el hardware basado en ese diseño.

El código fuente del hardware, su diseño, está disponible preferiblemente en un formato que permita hacerle modificaciones.

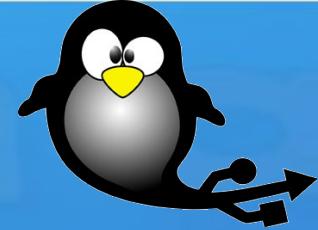


Traducción de la Declaración de Principios del  
OSHW.



# PINGUINO

## Introducción al Hardware Libre



### Algunos Proyectos de Hardware Libre

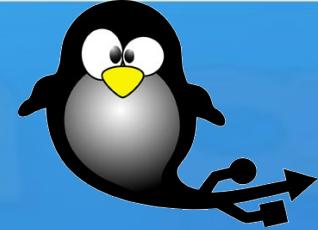
- Elphel, Inc. Cámaras reconfigurables de red





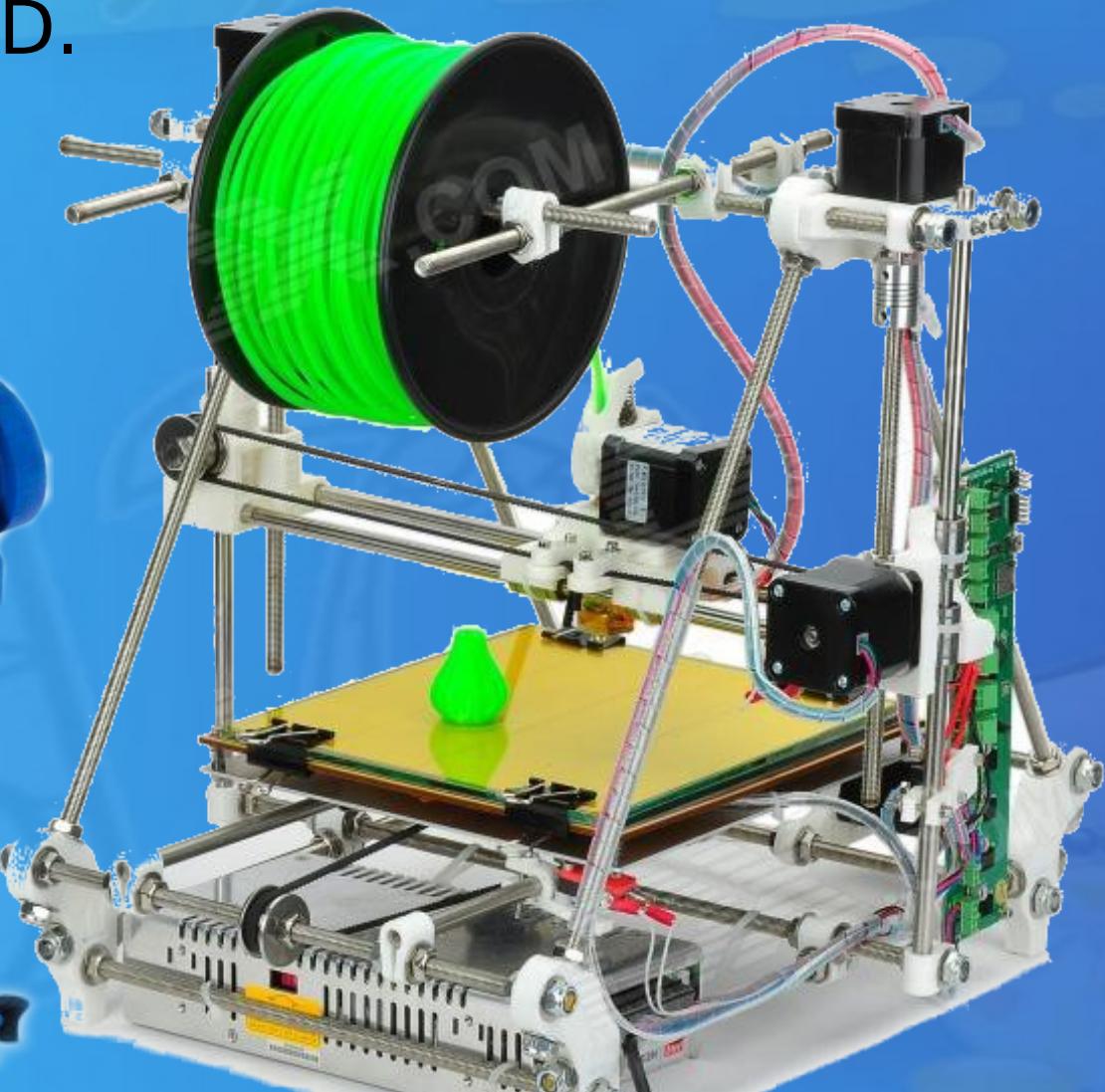
# PINGUINO

## Introducción al Hardware Libre



### Algunos Proyectos de Hardware Libre

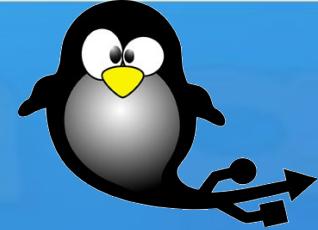
- RepRap: Impresoras 3D.





# PINGUINO

## Introducción al Hardware Libre

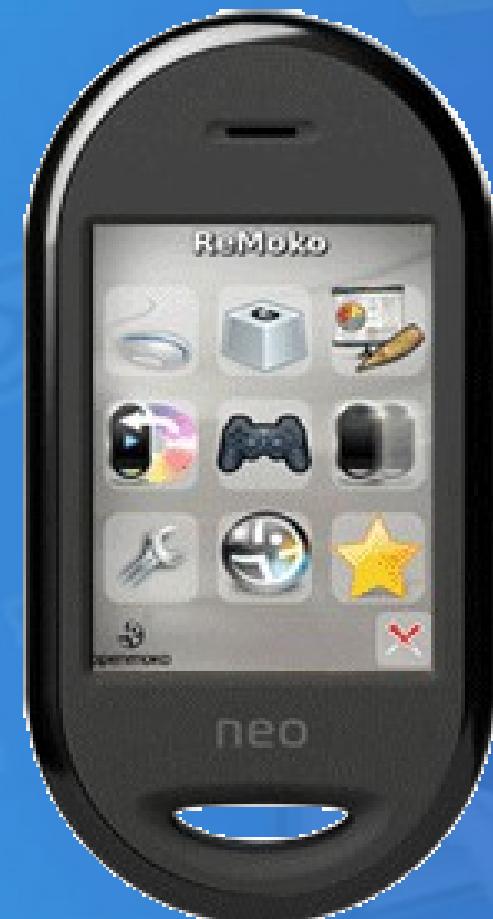


### Algunos Proyectos de Hardware Libre

- OpenMoko - Framework de open phone.



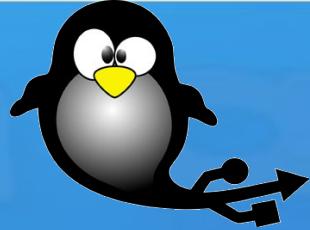
**openmoko**





# PINGUINO

## Introducción al Hardware Libre



### Algunos Proyectos de Hardware Libre

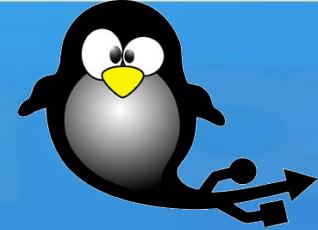
- OSCar: es el primer intento de diseñar un coche entero usando los principios de la cultura libre.





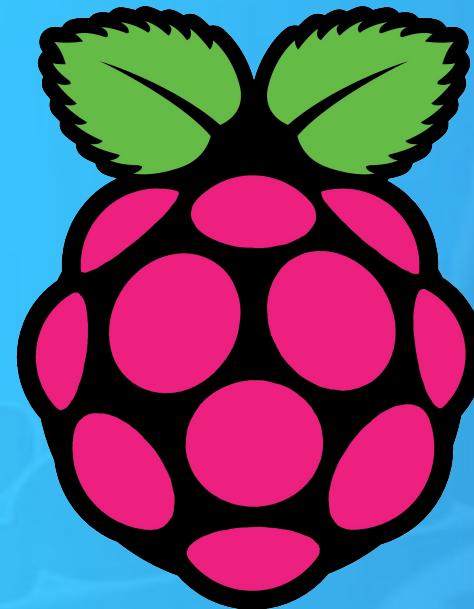
# PINGUINO

## Introducción al Hardware Libre



### Hardware Libre en dispositivos electrónicos

**Raspberry Pi.** es un ordenador de placa reducida o (placa única) (SBC) de bajo costo.

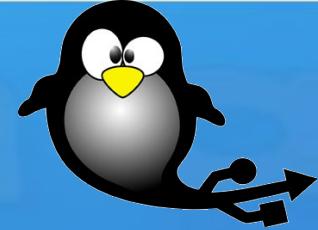


# Raspberry Pi



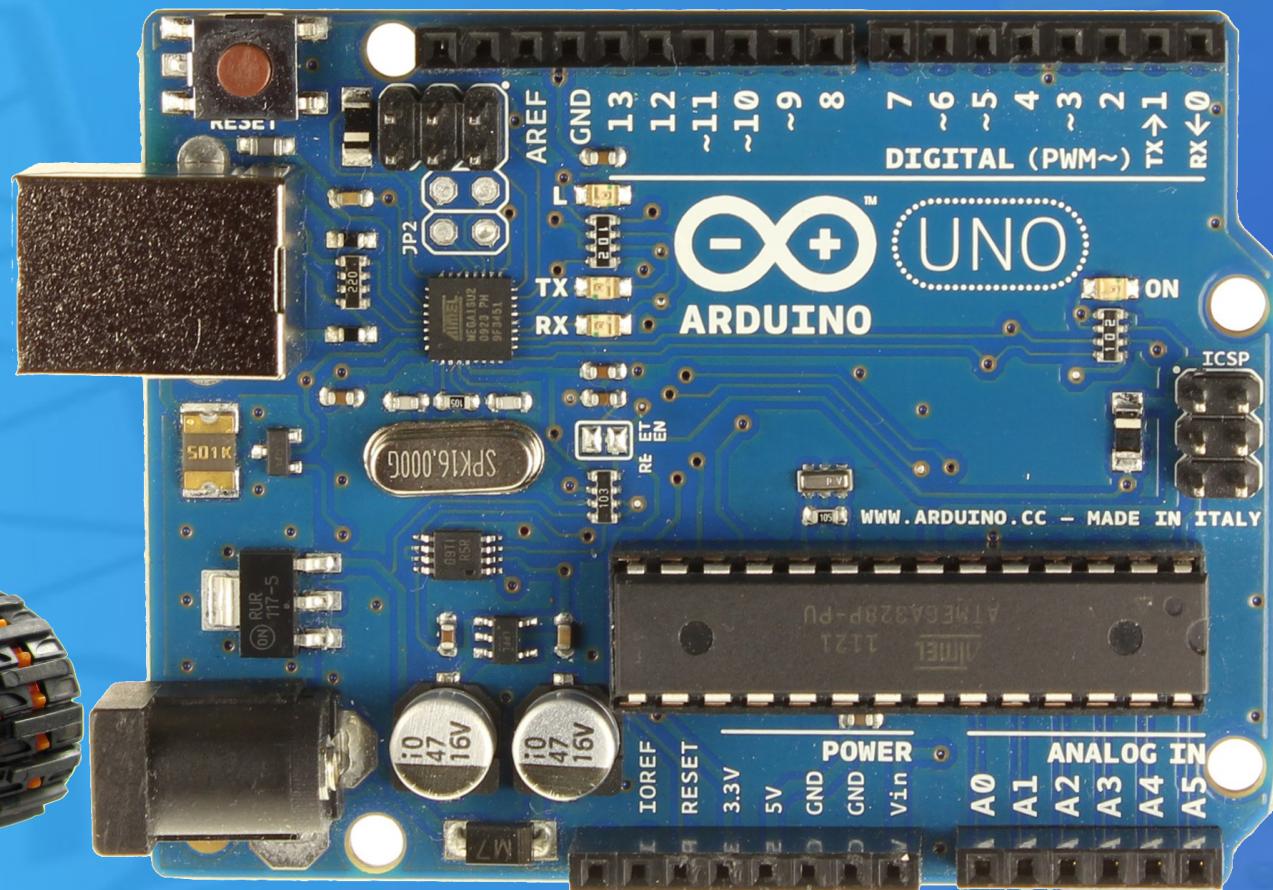
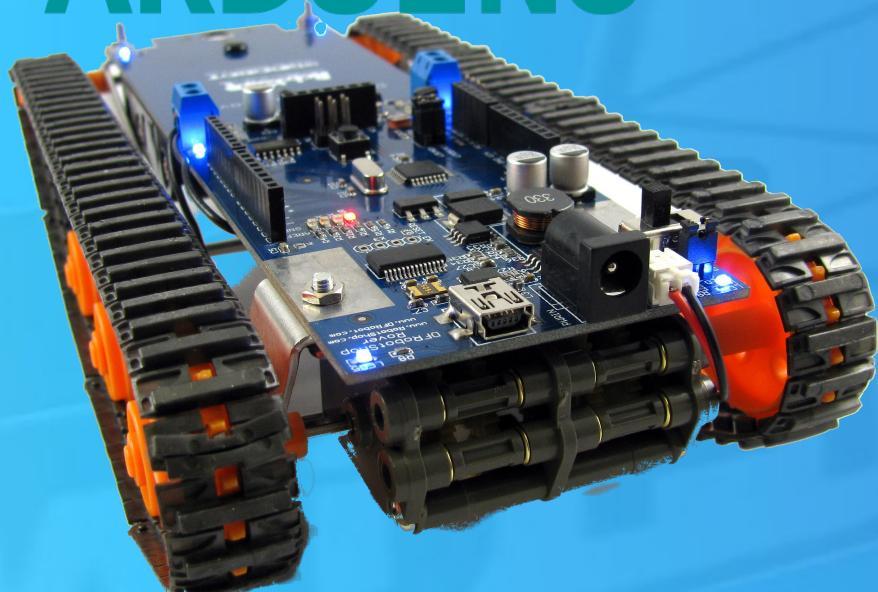
# PINGUINO

## Introducción al Hardware Libre



### Hardware Libre en dispositivos electrónicos

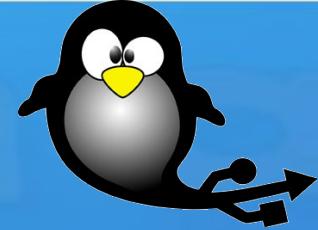
- Arduino. Tarjeta de I/O y ambiente de desarrollo.





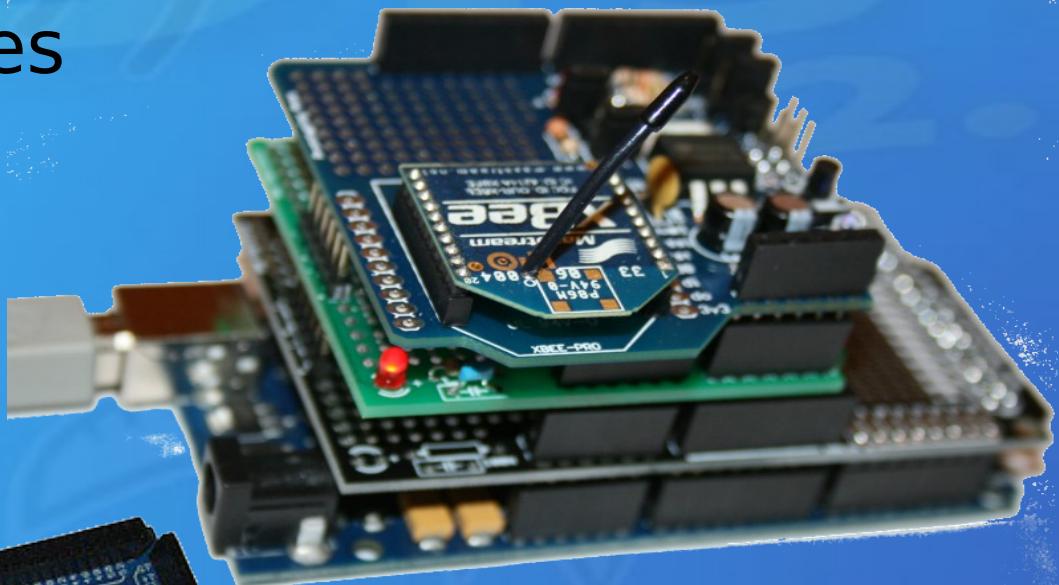
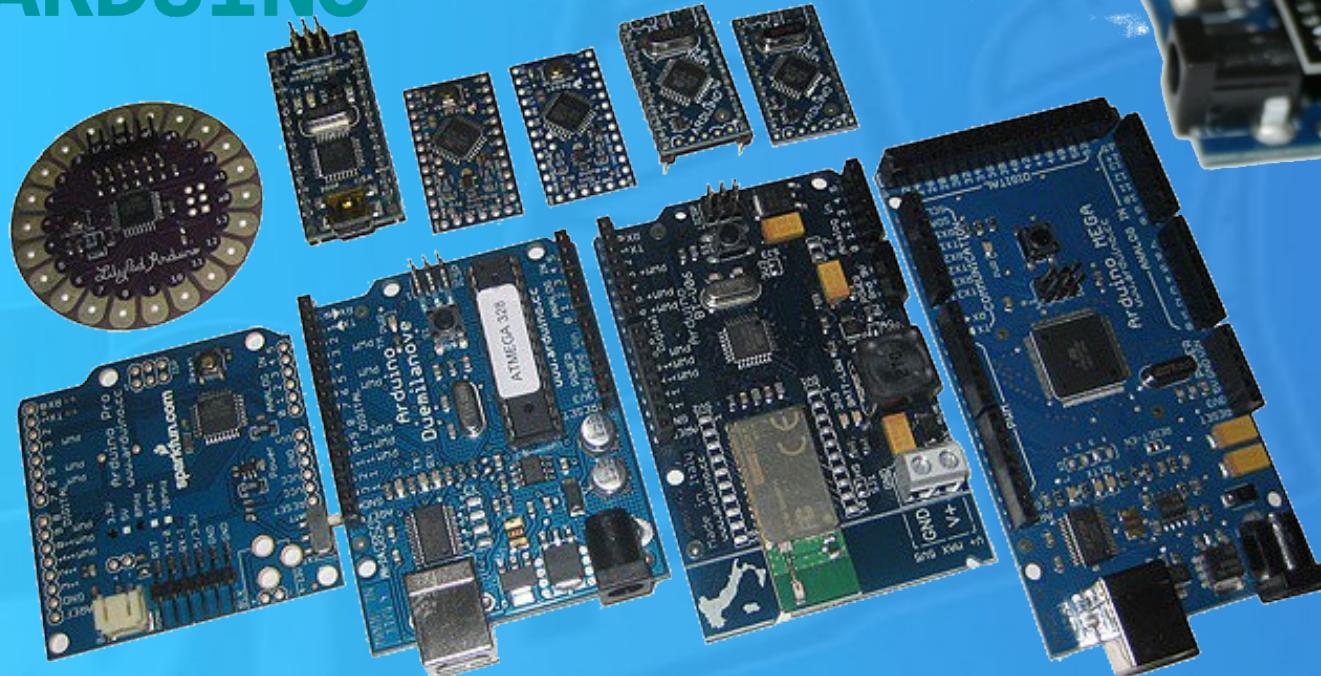
# PINGUINO

## Introducción al Hardware Libre



### Hardware Libre en dispositivos electrónicos

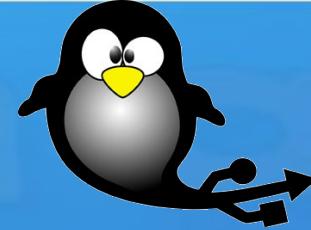
- Arduino. Diversos Sabores





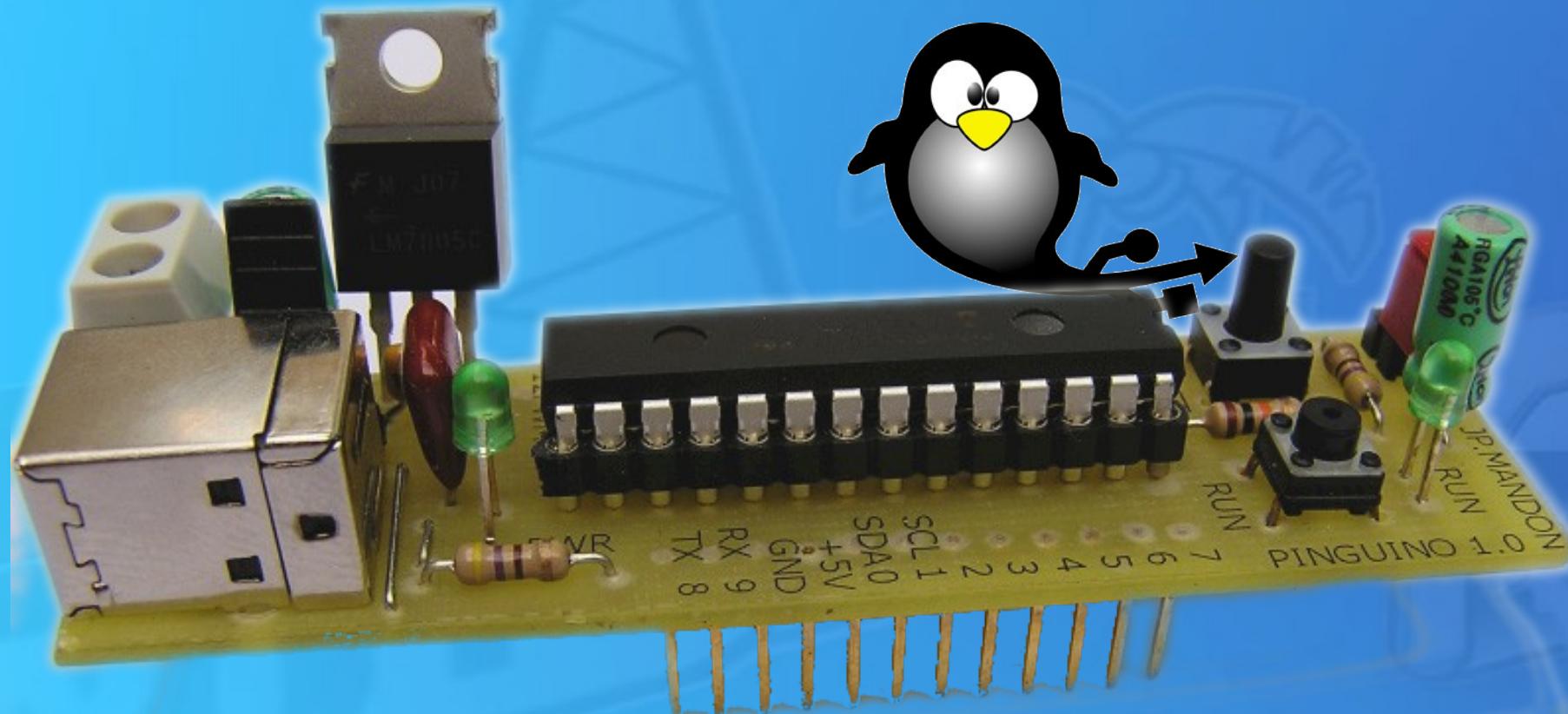
# PINGUINO

## Introducción al Hardware Libre



### Pinguino

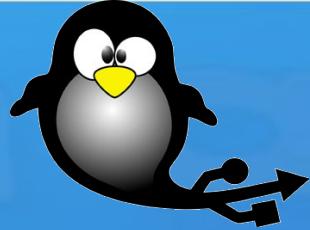
Es una plataforma similar a Arduino, y compatible con ella, pero basada en los micro-controladores PIC Microchip de 8 o 32 bits.





# PINGUINO

## Que es Pinguino?

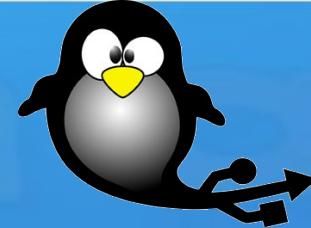


- 🐧 Proyecto de Código Abierto y Hardware Abierto.
- 🐧 Entorno de Desarrollo Integrado (IDE)
- 🐧 Compatible con los Micro-controladores ©Microchip de 8-bits (PIC18FX USB integrado) y 32-bits (PIC32MX).
- 🐧 Usa compiladores libres y gratuitos (SDCC y gcc-mips-elf) disponibles para GNU/Linux, Windows y Mac OS X.
- 🐧 Proyecto similar a Arduino. Esto implica que Pinguino tiene una compatibilidad cercana al 100% con el lenguaje y las librerías de Arduino.



# PINGUINO

## Documentación



🐧 <http://www.hackinglab.org/>



🐧 <https://sites.google.com/site/pinguinotutorial/home>

🐧 <http://pinguino.cc/>

🐧 [http://wiki.pinguino.cc/index.php/Main\\_Page](http://wiki.pinguino.cc/index.php/Main_Page)

Log in / create account

page discussion view source history

## Main Page

Language: English • Deutsch • Español • Français • 日本語

navigation

- Main Site
- Main Page
- Recent changes
- Wiki Help
- Functions
- Libraries
- Examples
- Downloads
- Pinguino Shop
- Contacts

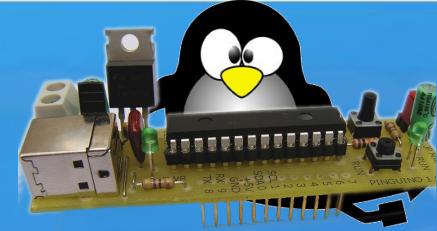
search

There are so many great platforms for creating digitally enabled devices that it has become hard to figure out which one to use. Arduino is maybe one of the most famous but it is based on ATMEL Microcontrollers. The goal of this project is to bring the simplicity of Arduino language to the Microchip PIC® microcontrollers with built-in USB hardware. [Read more...](#)



# PINGUINO

## Introducción al Hardware Libre



### Pinguino

vs

### Arduino

Microchip

USB integrado en el hardware.

IDE escrito en Python.

La comunidad Pinguino es muy pequeña (voluntarios ) en comparación con la comunidad de Arduino.

El proyecto Pinguino no se alimenta de grandes y rentables las ventas de tableros

Atmel

Chip conversor Serie a USB.

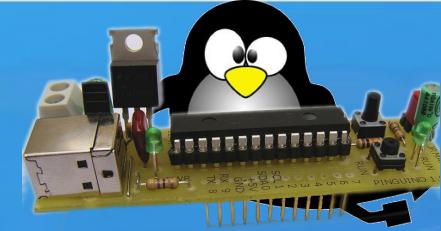
IDE esta escrito en JAVA.

Arduino ha generado una gran cantidad de compañías de clonación.



# PINGUINO

## Que necesitamos?



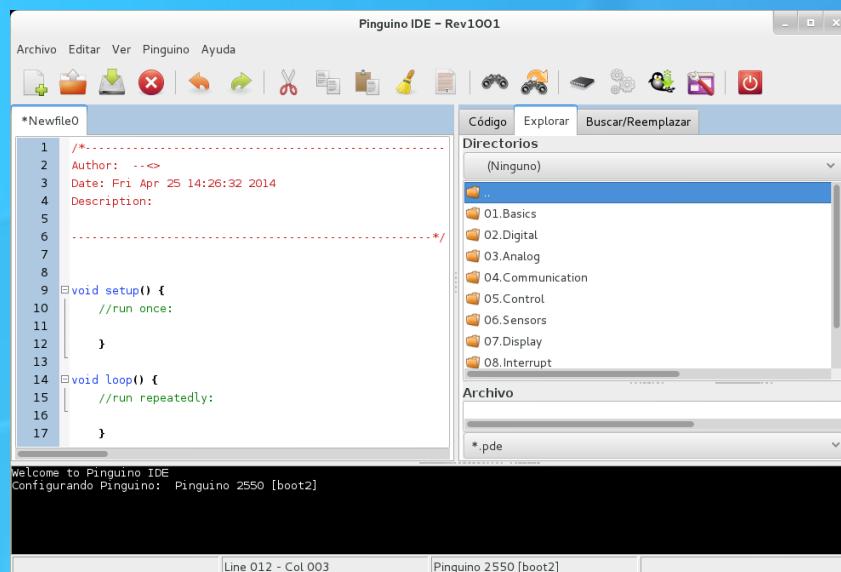
### Pinguino IDE

GNU/Linux

- python-usb
- python-wxgtk
- python-wxversion

OS X

Windows



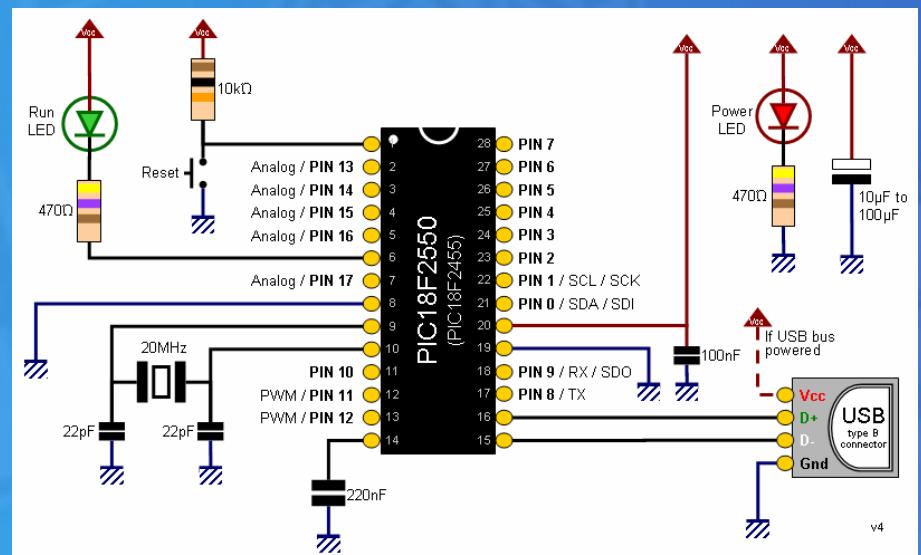
### Hardware

8-bit

PIC18F2550,  
PIC18F4550 and  
PIC18F26J50

32-bit

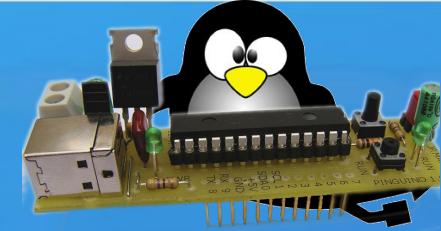
PIC32MX (Mips family)





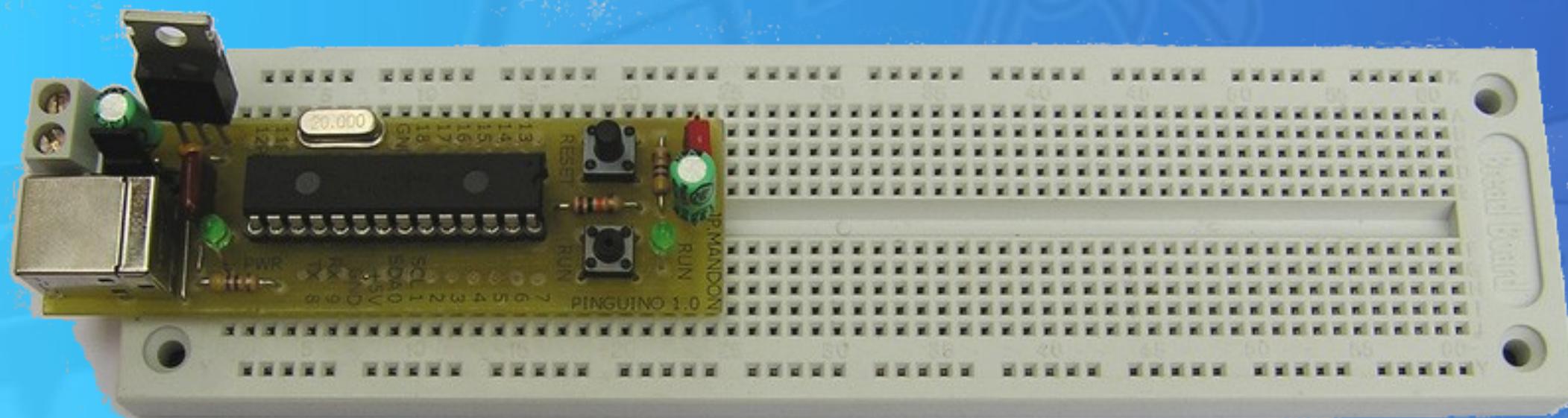
# PINGUINO

## Que necesitamos?



### Bootloader

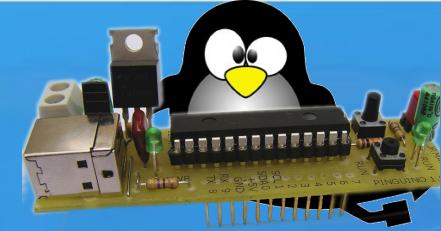
El bootloader es un pequeño programa ejecutándose dentro del micro-controlador de Pinguino (PIC18F o PIC32MX) que permite transferir el programa compilados del usuario, desde el ordenador, a la memoria de programa del micro-controlador, y dar el control a dicho programa una vez esta gravado correctamente.





# PINGUINO

## Como cargar el Bootloader?



### Programador de PIC

En las placas comerciales, el bootloader siempre esta pre-instalado.

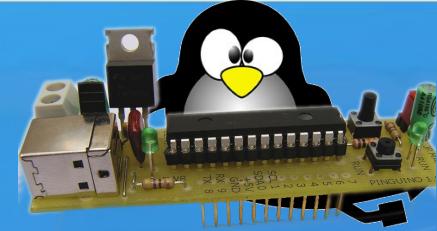
Si te has fabricado tu propia placa, entonces es necesario que programes el micro-controlador (una vez solo), usando un programador para PICs (Pickit 2/3, etc...) que soporte el micro-controlador que estés usando.



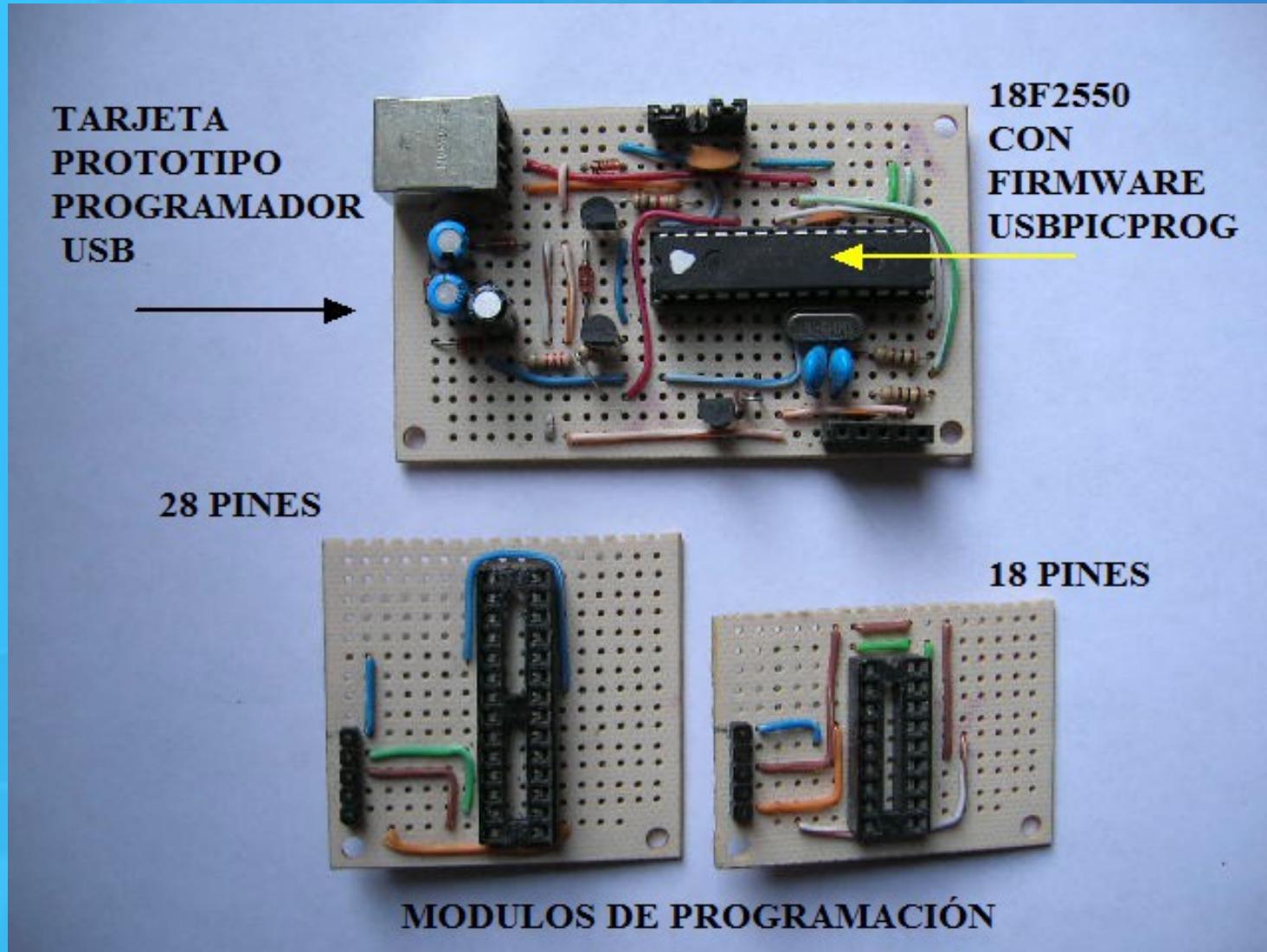


# PINGUINO

## Como cargar el Bootloader?



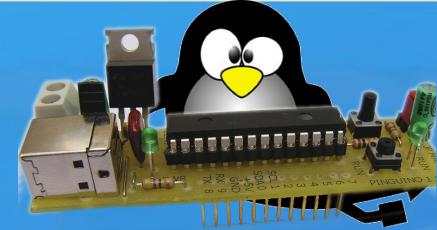
### Programador de PIC USB Casero



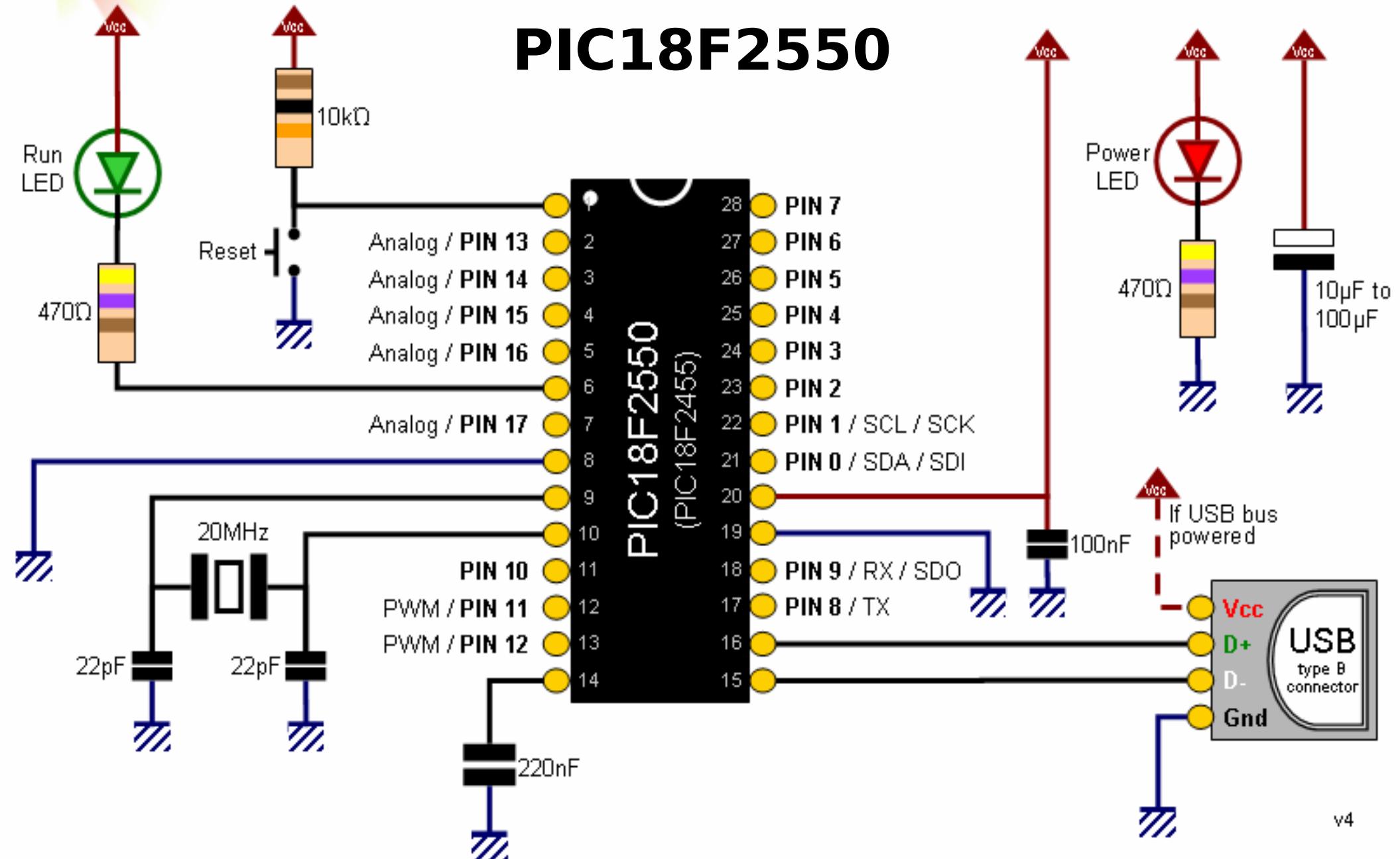


# PINGUINO

## Que necesitamos?



### PIC18F2550

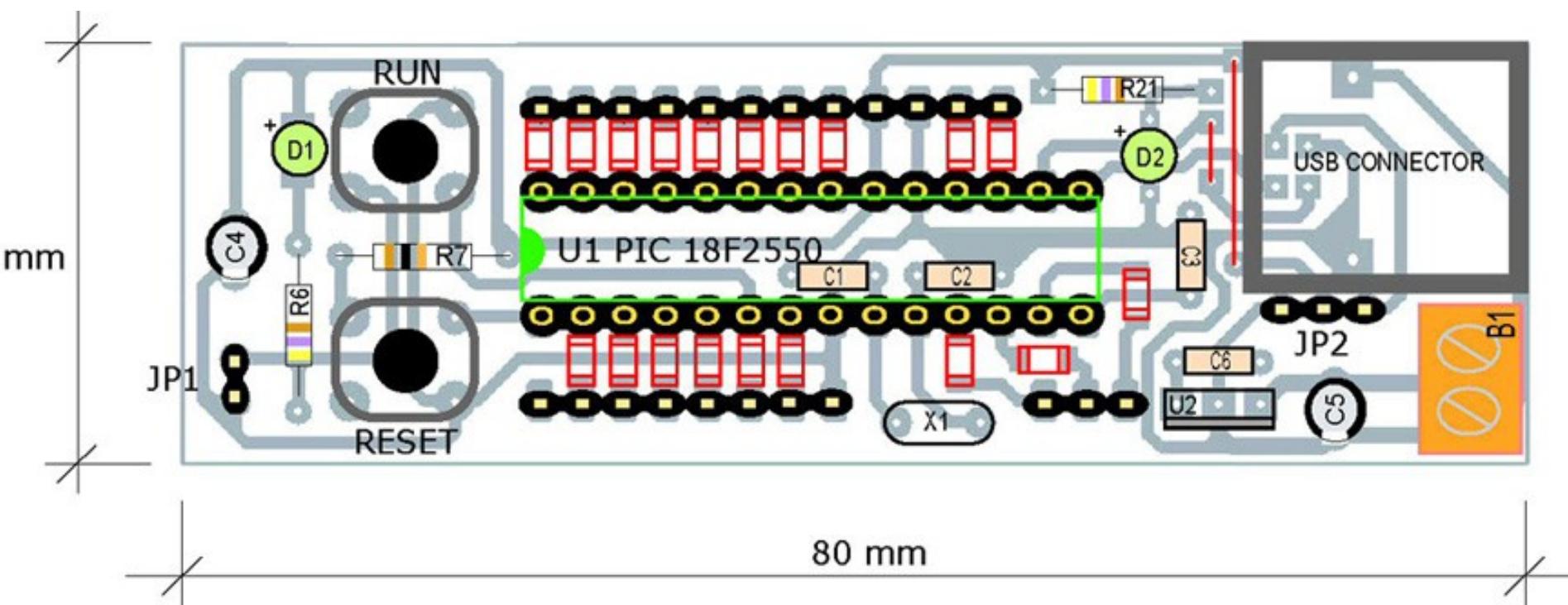
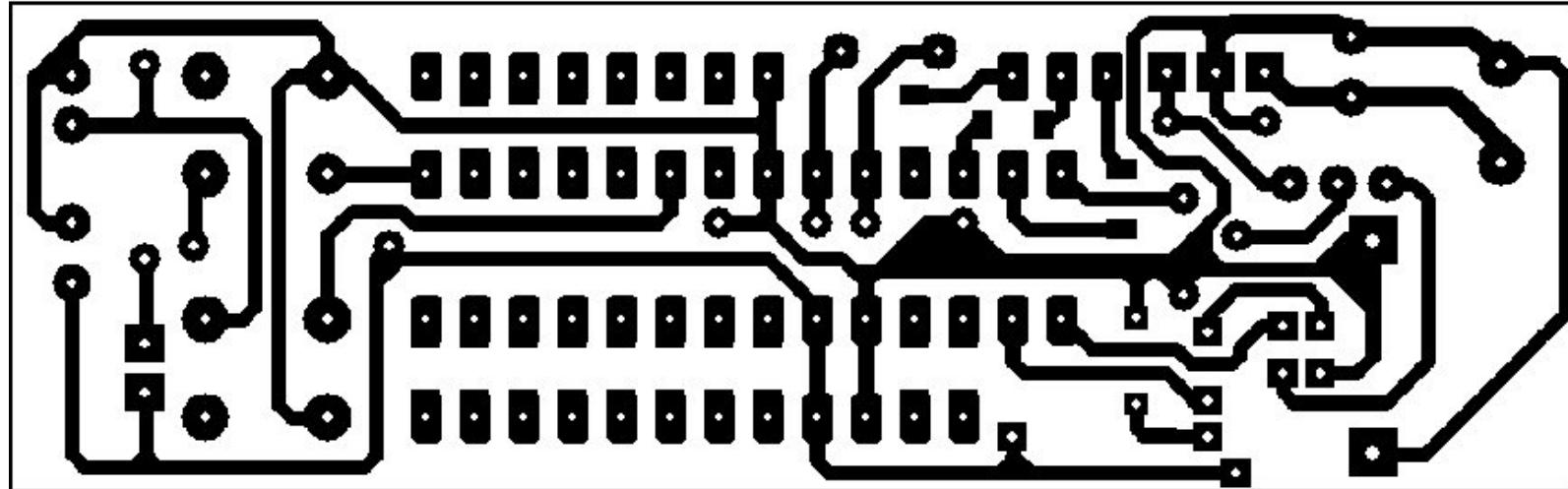
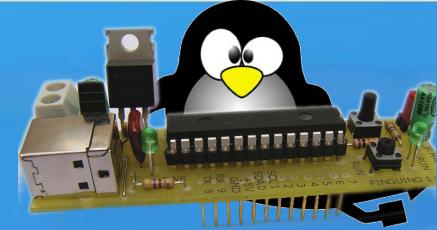


v4



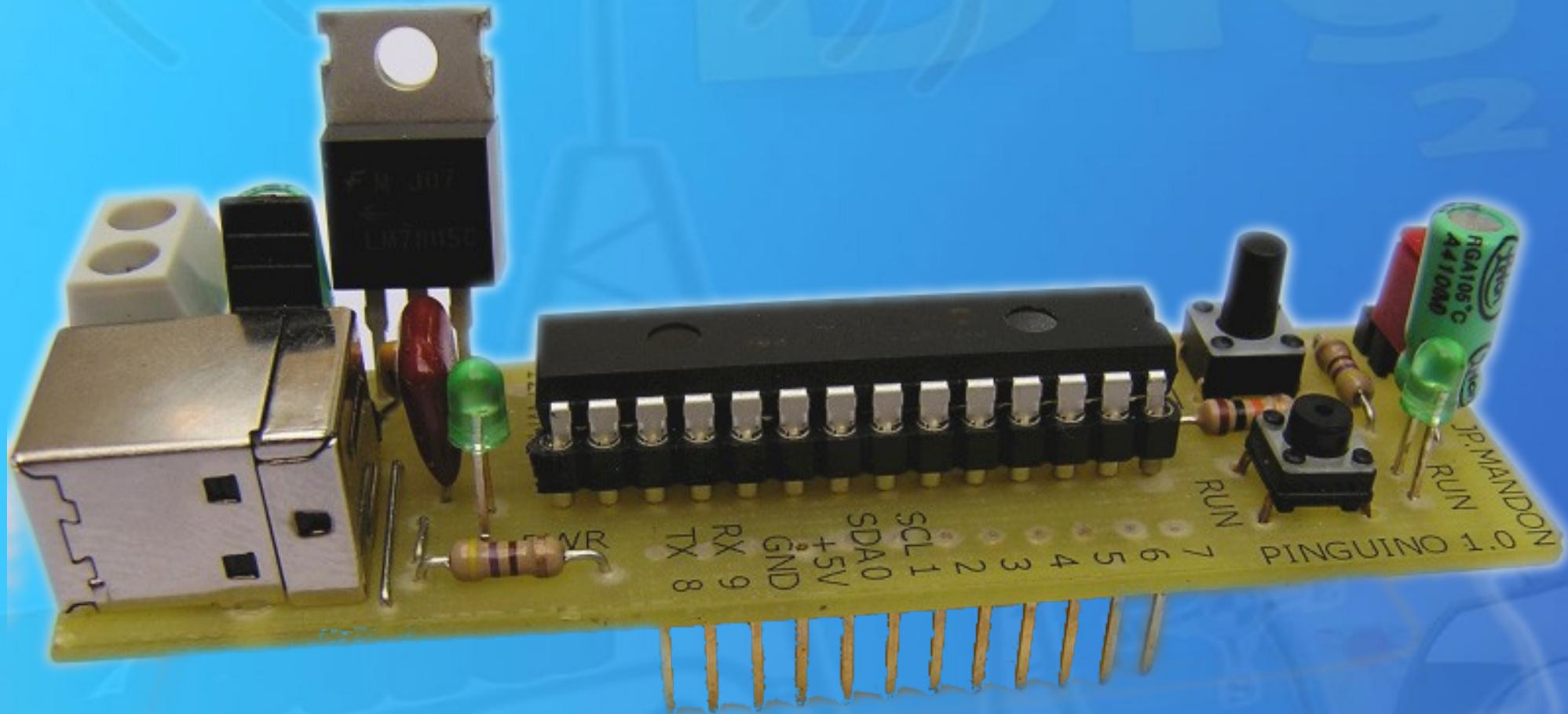
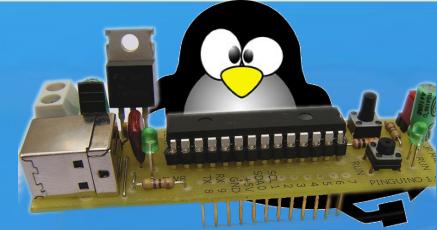
# PINGUINO

## Que necesitamos?



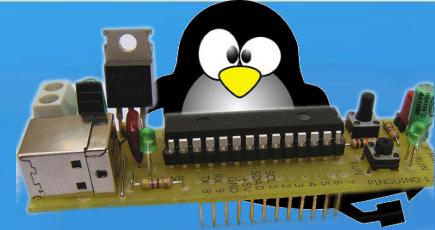


# PINGUINO PIC18F2550

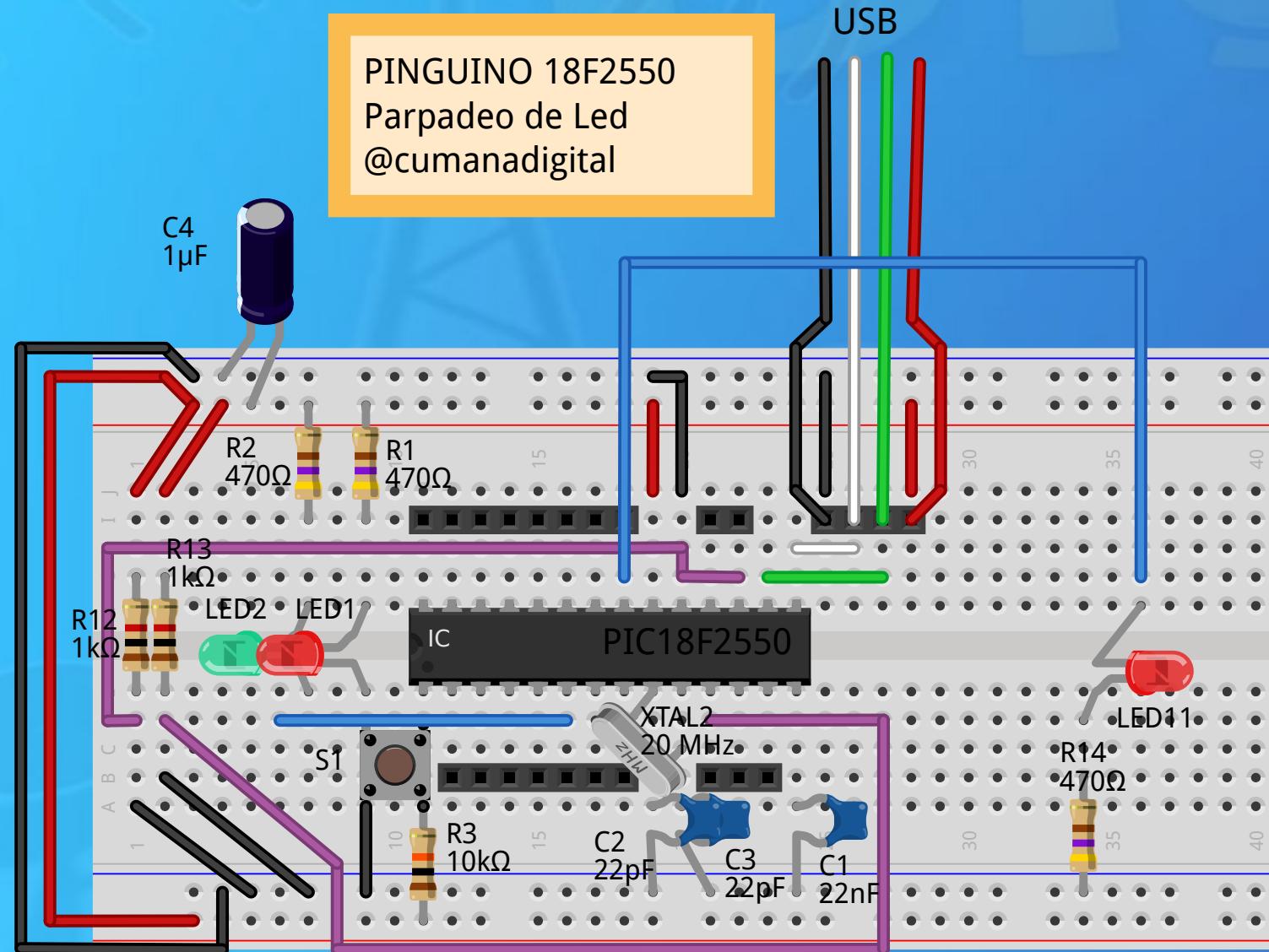




# PINGUINO PIC18F2550



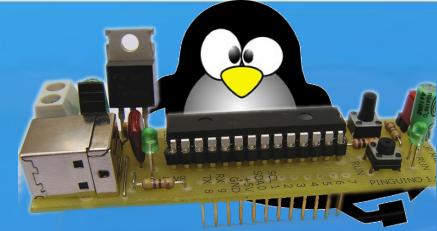
## Pinguino PIC18F2550 en protoboard





# PINGUINO

## Como programar el PIC



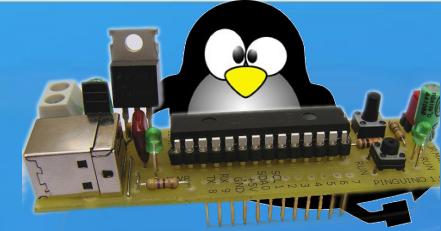
### Código Fuente (archivo .pde)

```
1  /* Blink a LED */
2
3 void setup()
4 {
5     // initialize the digital pin USERLED as an output.
6     pinMode(USERLED, OUTPUT);
7
8 void loop()
9 {
10    toggle(USERLED);      // alternate ON and OFF
11    delay(500);           // wait for 500ms
12}
13
```

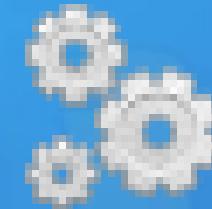


# PINGUINO

## Como programar el PIC



### **Compilar (genera archivo .hex)**



```
Tarjeta:          Pinguino 2550
Proc:    18f2550
Archivo:         /opt/pinguino32/x.4/examples/01.Basics/Blink/Blink.pde
compilando...
Compilacion completada
tamaño de codigo:2556 / 24576 bytes (10% usado)
0.3 segundos (Tiempo de proceso)
```

### **Upload to Pinguino Board**



```
Pinguino found

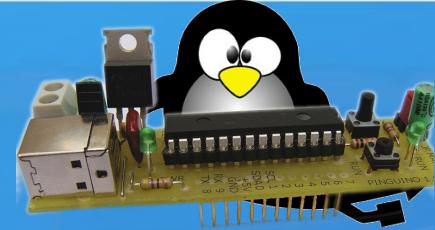
Bootloader v2.12

Writing ...

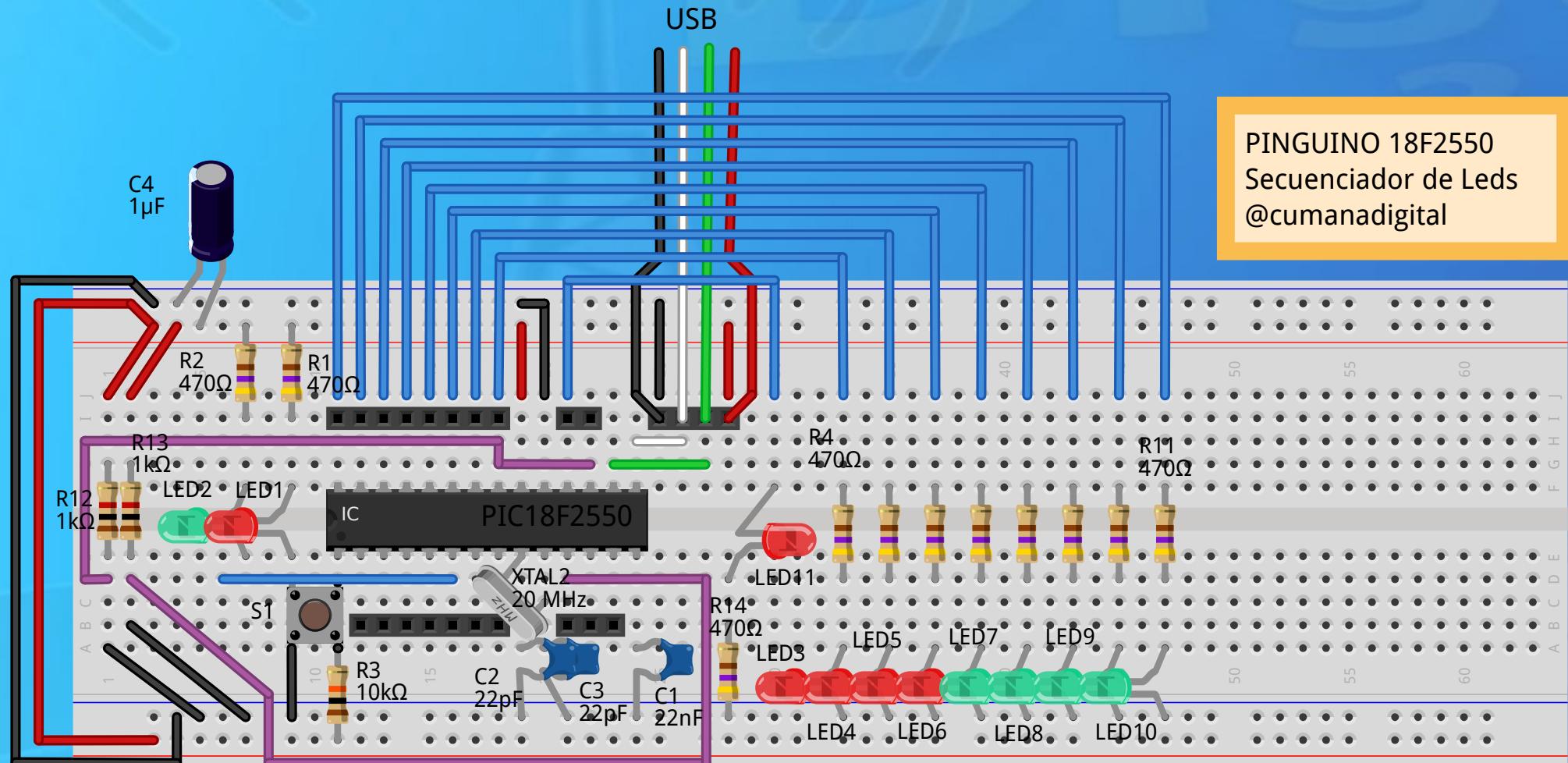
Blink.hex successfully uploaded
```



# PINGUINO PIC18F2550



## Pinguino PIC18F2550 en protoboard



# GRACIAS

Oswaldo Hernández  
Cel: 0416-593.63.95

oswaldoehc@gmail.com  
cumanadigital@hotmail.com  
<http://cumanadigital.blogspot.com>

