

Arduino: Computación física libre

@gnuardo



I+D+i
JAVANTE
CONSEJERÍA DE SALUD

El proyecto

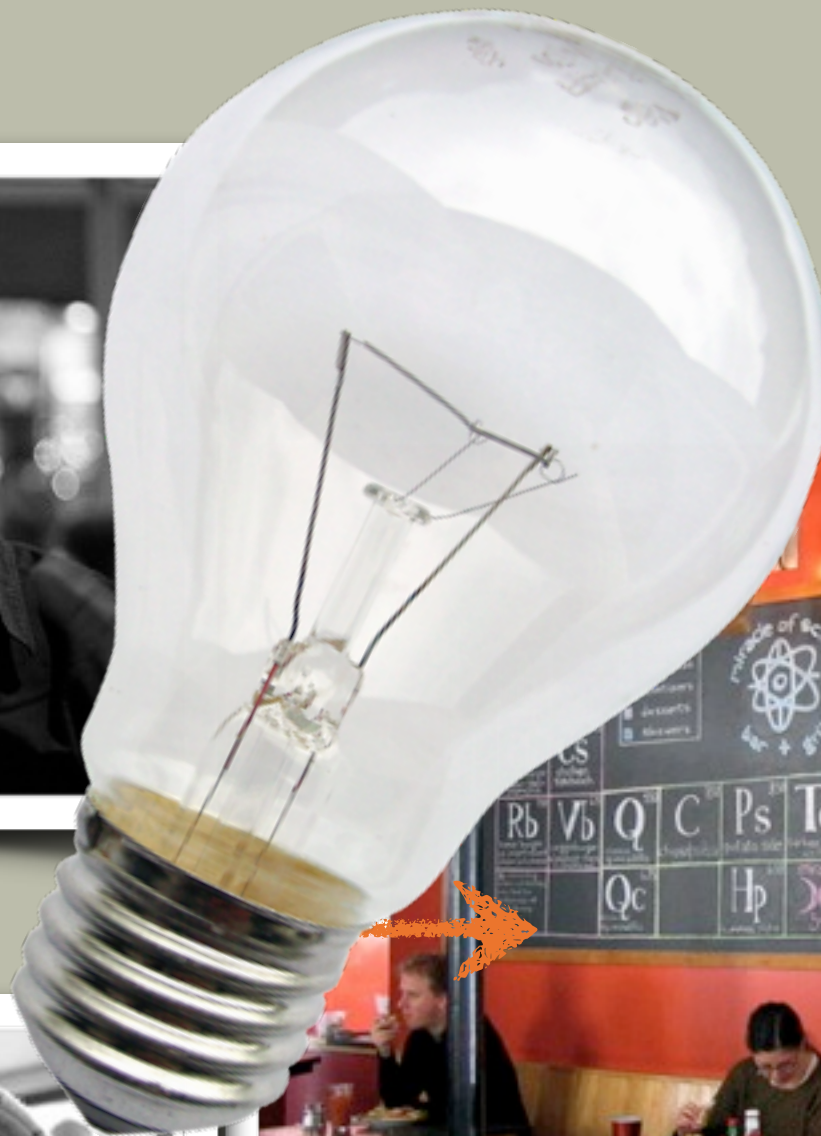


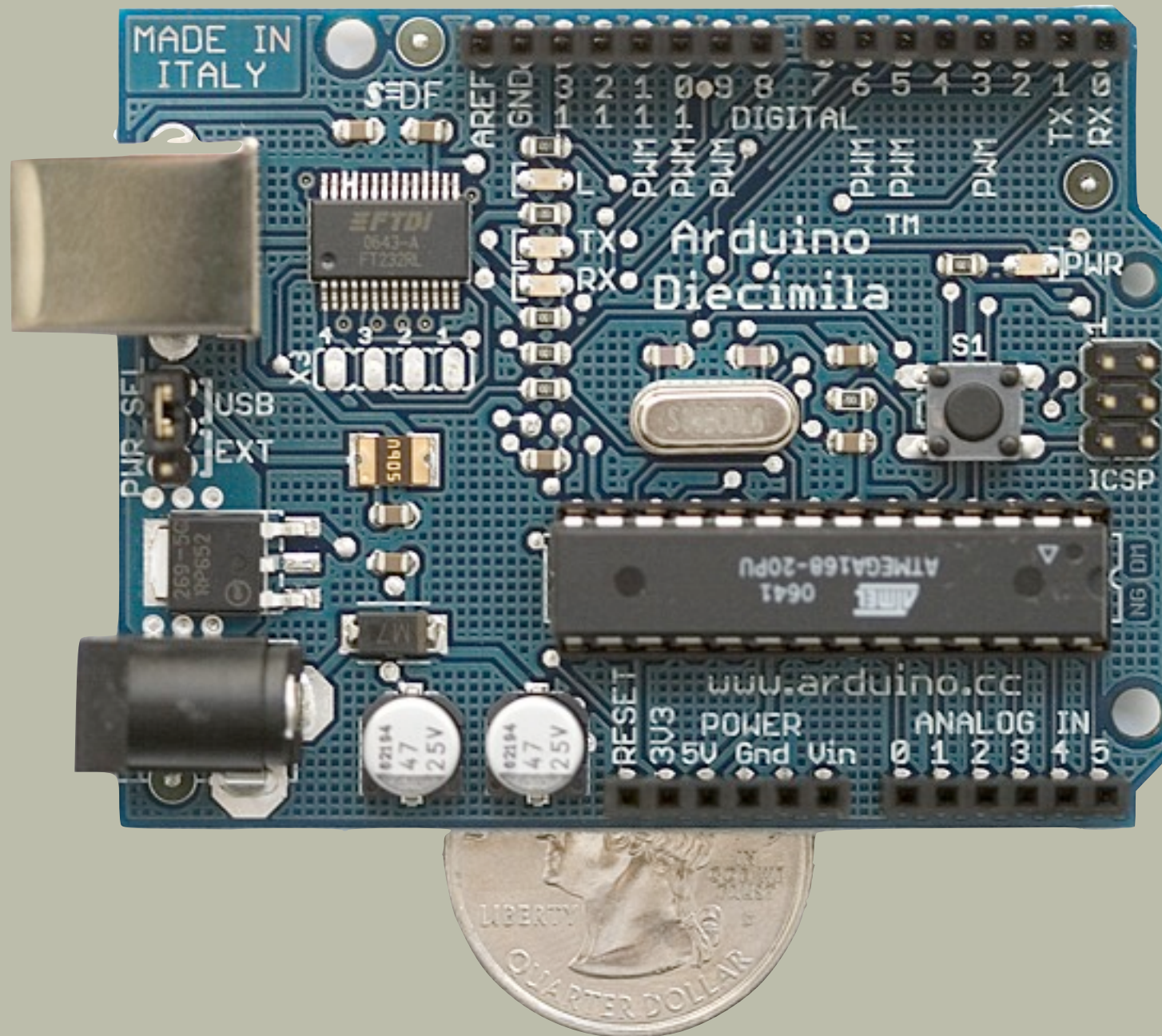
I+D+i
CONSEJERÍA DE SALUD





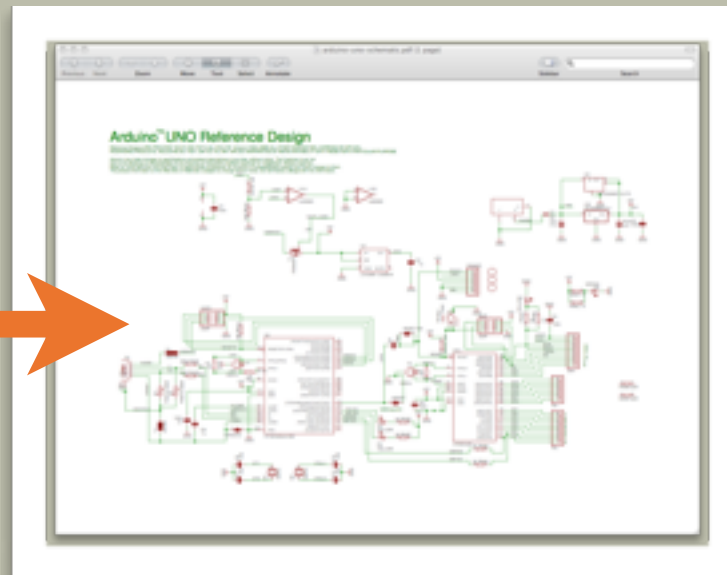






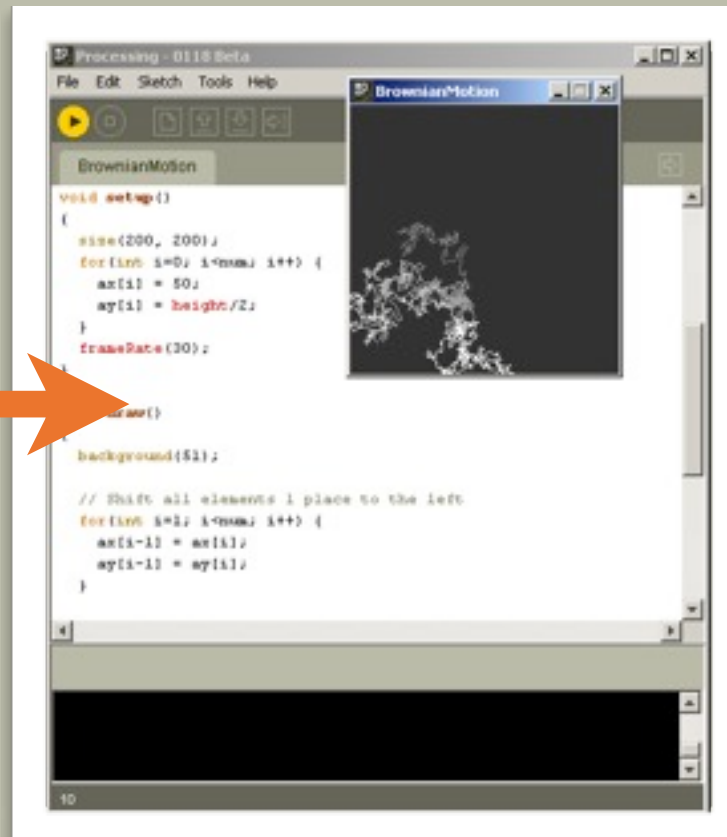
¿Qué es arduino?

Open
Hardware



Todos los esquemas se
se pueden descargar
desde la misma página
de arduino: [http://
arduino.cc](http://arduino.cc)

Open
Software



El lenguaje de
programación,
el IDE y todo el entorno
necesario
también son libres.

¿Y la física?

momentum
 $p = mv$

uniform
circular
motion

FBD



freefall
 $\Delta y = v_i t - \frac{1}{2} g t^2$

vectors



Projectile



(c) MCTgarcia photography

Computación física

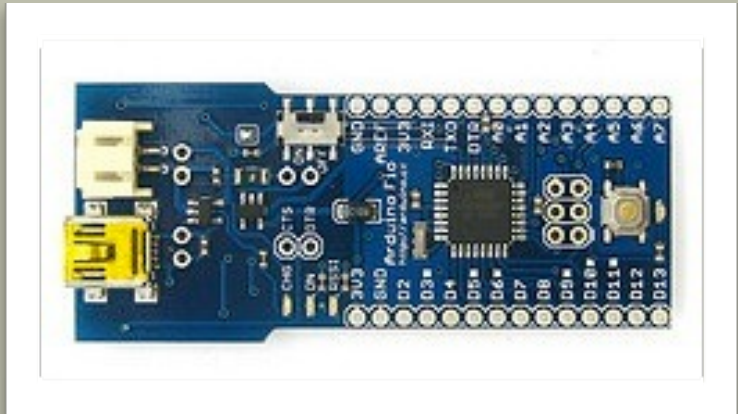
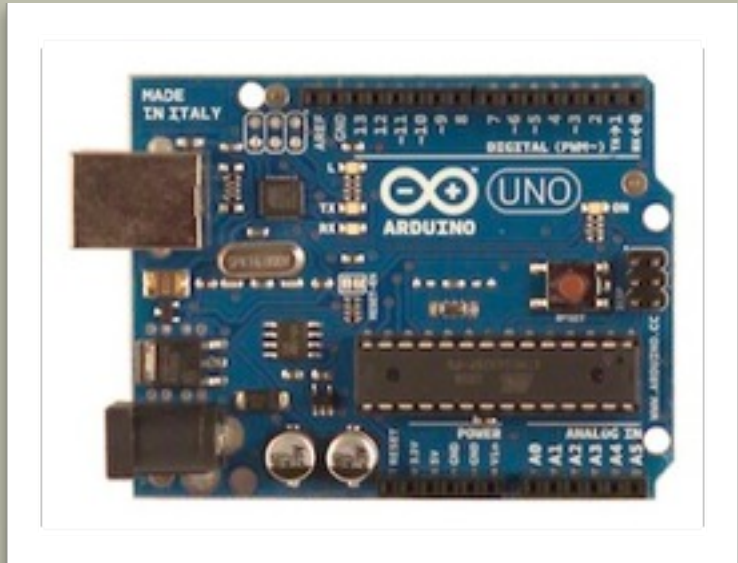
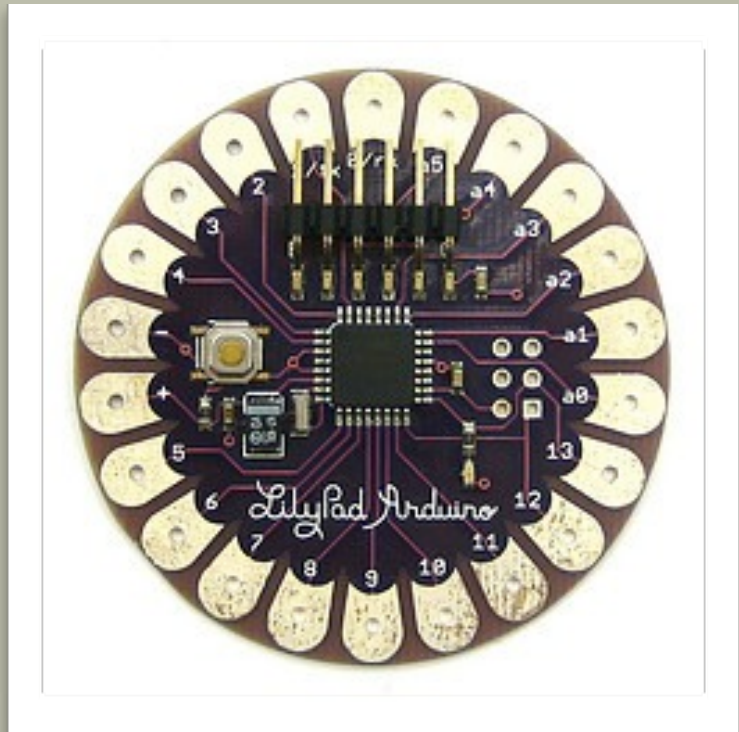
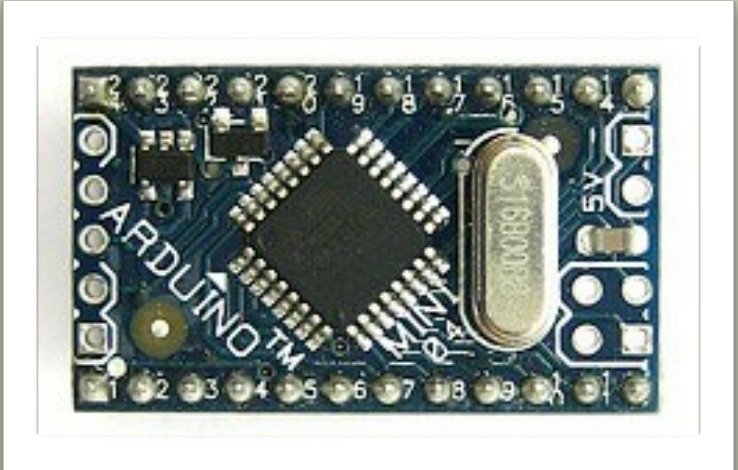
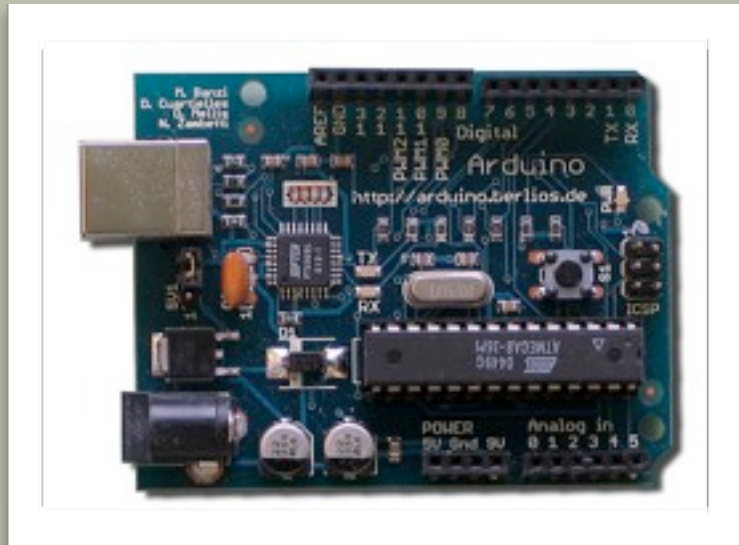
Dan O'Sullivan y Tom Igoe:
(physicalcomputing.com)

“Physical computing is about creating a conversation between the physical world and the virtual world of the computer”

El hardware



IAVANTE
CONSEJERÍA DE SALUD



Todos tienen en común:

1.

FAMILIA DEL
MICRO
(ATMEL
ATMEGA)*

2.

LENGUAJE DE
PROGRAMA-
CIÓN

3.

EEPROM,
SDRAM Y LAS
SIGUIENTES
E/S

* casi siempre

Tipos de E/S

1.

Entrada Analógica

2.

Entrada/Salida Digital

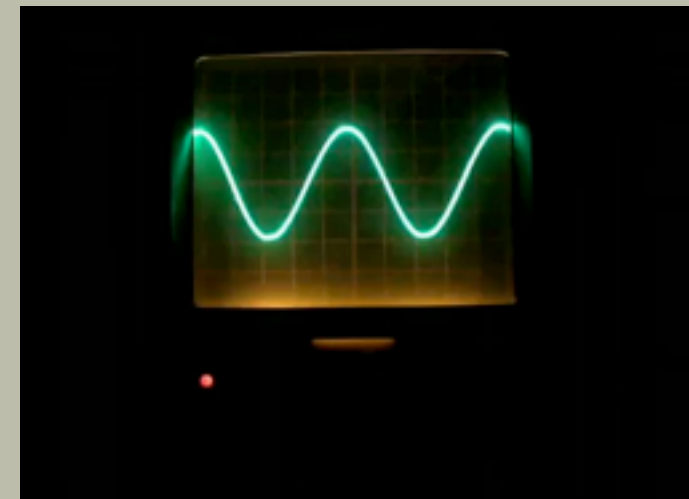
3.

Modulación por ancho de pulsos

Tipos de E/S

1.

Entrada Analógica



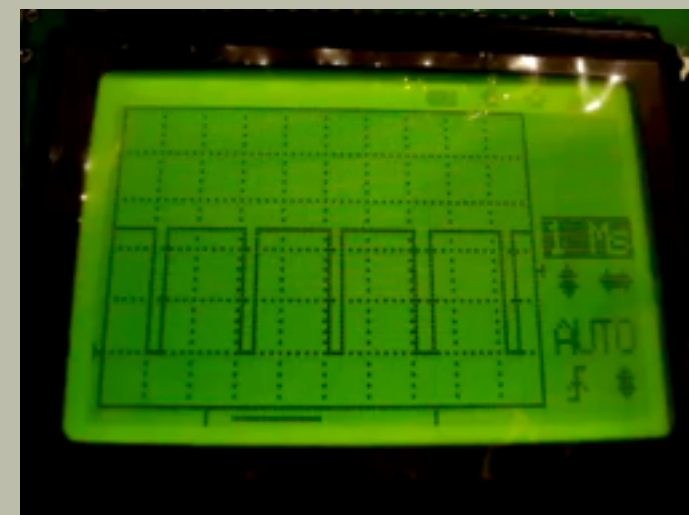
2.

Entrada/Salida Digital



3.

Modulación por ancho de pulsos



Escudos o “shields”

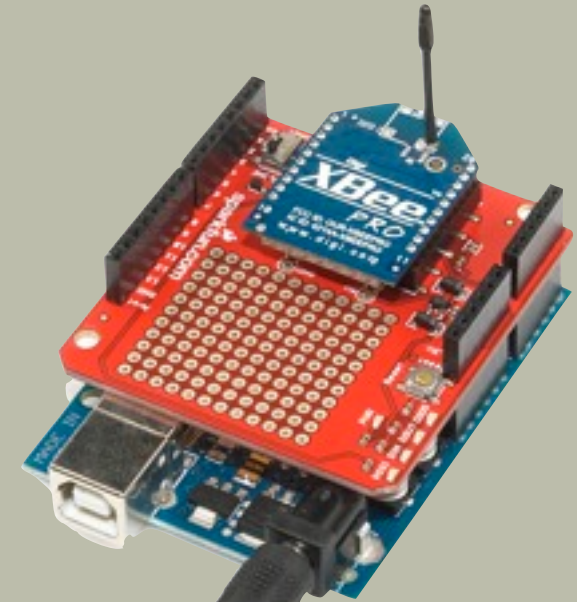
1.

Ethernet



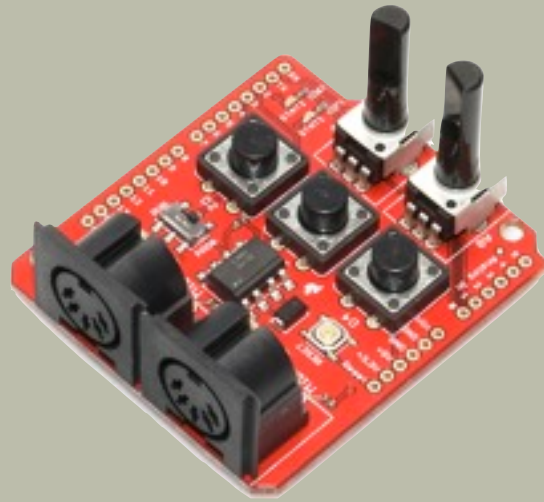
2.

ZigBee



3.

Midi



4.

CAN



5.

LCD



6.

GSM



Sensores

1.

Alcohol



2.

CO



3.

Acelerómetros



4.

Ultra
Sonidos



5.

Luz



6.

Temperatura



Principales diferencias entre el hardware arduino

1.
EEPROM



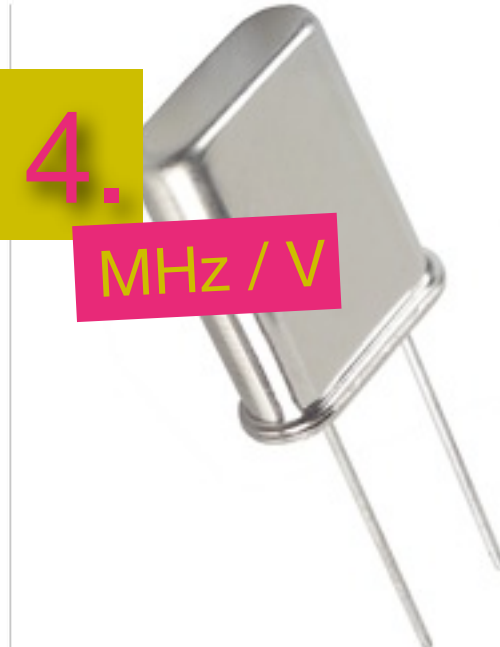
2.
SDRAM



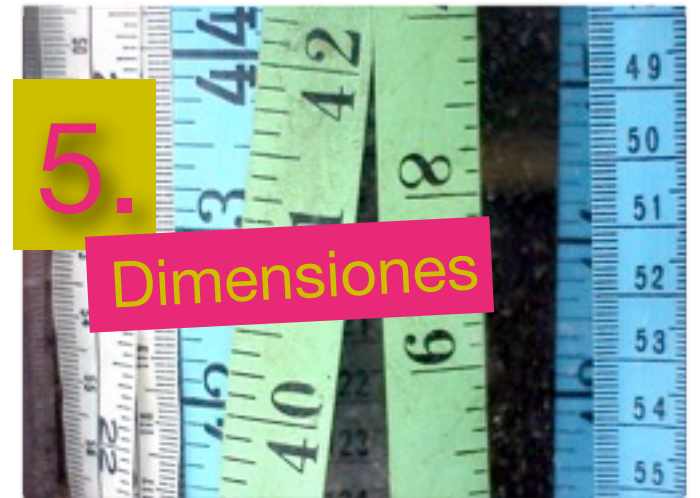
3.
I/O



4.
MHz / V



5.
Dimensiones



El Software



JUNTA DE ANDALUCÍA

IAVANTE
CONSEJERÍA DE SALUD

Lenguaje e IDE



Processing: Creado por Casey Reas y Ben Fry. Sintaxis muy similar a C. Público objetivo: artes electrónicas y “comunidades de diseño visual”.



**C++
Simplificado**



Wiring: Creado por Hernando Barragán. Es la biblioteca que encapsula todo el acceso a la E/S de los dispositivos.



**Acceso al Hardware
Simplificado**

Lenguaje

Estructura

`[] = , // {} . /* */ #include () ; void`

Datos

`boolean, byte, char, double, ...`

Control

`== > < != for() while() break if else ..`

E/S

`digitalWrite() pinMode() ...`

Lenguaje

Math



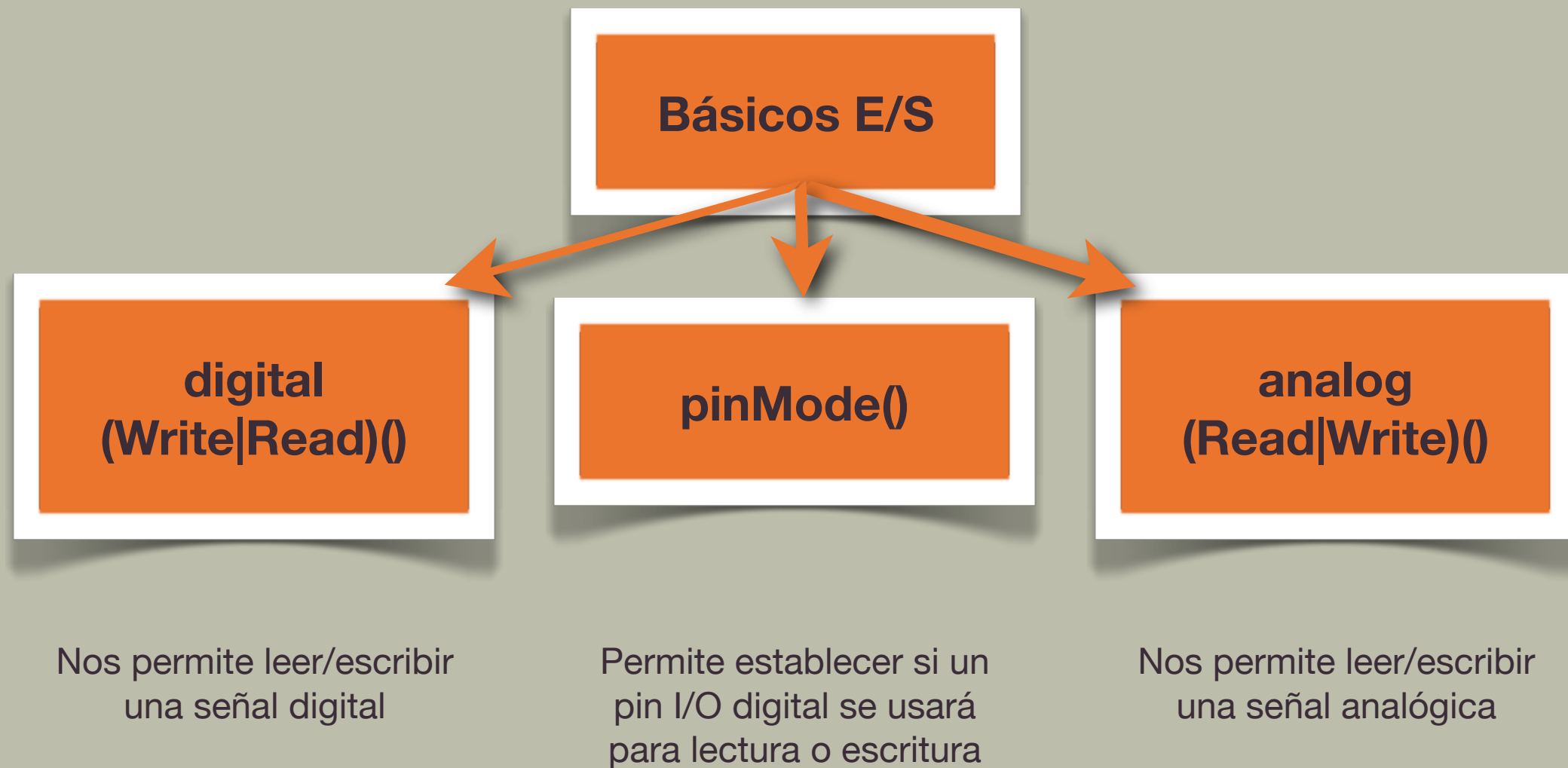
abs() sqrt() log() >> <<

Constantes



PI, HIGH, LOW, OUTPUT

Lenguaje



Bibliotecas

Serial

Soporte de las comunicaciones entre el hardware y dispositivos externos vía puerto serie.

EEPROM

Escritura/Lectura de la EEPROM interna del hardware.

Servo

Manipulación de motores servo.

Liquid Crystal

Manipulación básica de LCDs.

Bibliotecas

Matrix

Manipulación de matrices LED.

Sprite

Animaciones en matrices LED.

NMEA

Descodificación de datos GPS.

OSC

Mensajería OSC entre el PC y el hardware.

IDE



Menu

Editor de texto

Área de
mensajes

Consola

IDE

1.

Sketches

Todo el código de nuestro proyecto se agrupa en un “sketch”. Cada sketch tiene su propio directorio. Si se necesitan ficheros externos al sketch se compila un .hex con el resultado antes de subir el código al hardware.

2.

Autoformato

Reformatea el código para permitir una mejor lectura “humana” por parte del mismo.

3.

Monitor serie

Permite abrir una conexión por el puerto serie y depurar todos los datos provenientes del hardware. Muy útil para depuración y verificación.

Insta- lación



IAVANTE
CONSEJERÍA DE SALUD

Paso a paso

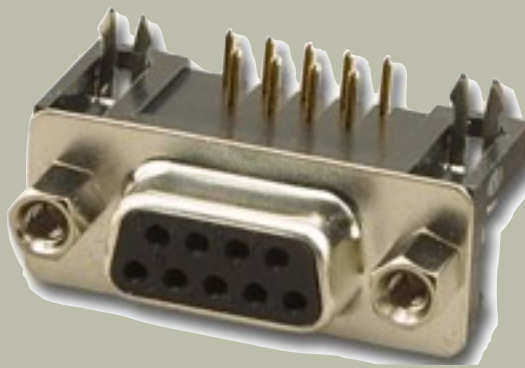


1.

Instalar librerías
de emulación de
32 bits para
JAVA

Gentoo:

```
emerge emul-linux-x86-java
```



2.

Dotar al usuario
de permisos
sobre el puerto
serie a usar

Gentoo:

```
sudo gpasswd -a $user uucp
```

Paso a paso



3.

Descargar el
entorno

Gentoo:

wget [http://wiring.org.co/
download/wiring-0027.tgz](http://wiring.org.co/download/wiring-0027.tgz)



4.

Descomprimir el
paquete

Gentoo:

tar xpvf wiring-0027.tgz

Paso a paso



5.

Ejecutar

Gentoo:
./wiring

Paso a paso (modo consola)

Gentoo:

Dependencias:

Si tienes un S.O. de 64 bits necesitarás las bibliotecas de emulación de 32bits para java:

(En gentoo el nombre del paquete es: app-emulation/emul-linux-x86-java)

Necesitarás java y libXext.

Ejemplos de permisos para los puerto serie:

```
gerardo@funtoo-GNUardo ~ $ ls -lh /dev/ttyS*
```

```
crw-rw---- 1 root uucp 4, 64 Nov 25 11:51 /dev/ttyS0
```

```
crw-rw---- 1 root uucp 4, 65 Nov 25 11:51 /dev/ttyS1
```

```
crw-rw---- 1 root uucp 4, 66 Nov 25 11:51 /dev/ttyS2
```

```
crw-rw---- 1 root uucp 4, 67 Nov 25 11:51 /dev/ttyS3
```

Añadir al usuario al grupo uucp:

```
sudo gpasswd -a gerardo uucp
```

```
gerardo@funtoo-GNUardo ~ $ cat /etc/group | grep gerardo | grep uucp
```

```
uucp:x:14:uucp,gerardo
```

... después salir y volver a entrar en la sesión (o en la shell).

Descargar wiring / arduino:

```
wget http://wiring.org.co/download/wiring-0027.tgz
```

Descomprimir el paquete:

```
tar xpvf wiring-0027.tgz
```

y ejecutarlo:

```
./wiring
```

Voila!

5 Prácticas



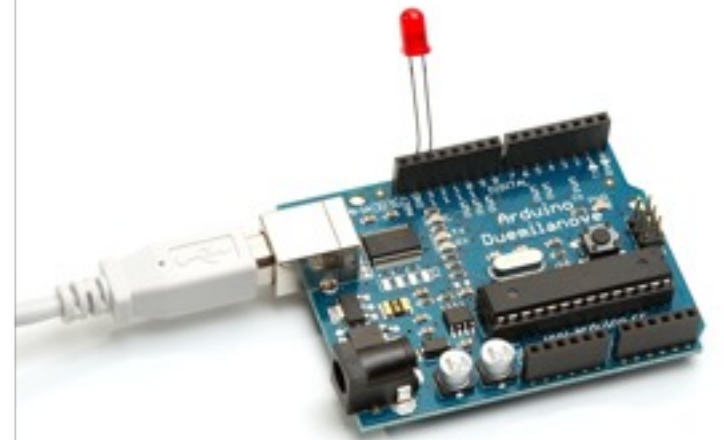
I+D+i
CONSEJERÍA DE SALUD

¿Qué vamos a hacer?

1.

Encendido de un led

Nuestro primer ejemplo será encender y apagar un simple led con una de las E/S digitales.



2.

Mover un motor con un H-Bridge

En el segundo ejemplo usaremos electrónica adicional que nos ayudará a mover un pequeño motor.



¿Qué vamos a hacer?

3.

Encendido de una bombilla 220VAC

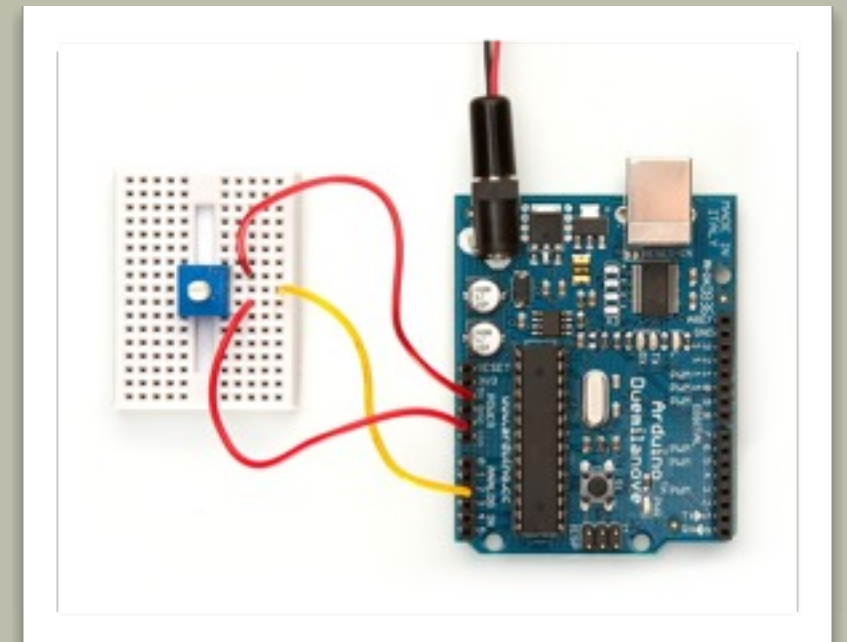
Usaremos un triac y un desacoplador óptico para conmutar corrientes 220VAC



4.

Dibujado de datos a tiempo real

Usaremos la comunicación serie y las entradas analógicas para, ayudándonos del software kst, realizar un trazado en tiempo real de los datos recibidos.



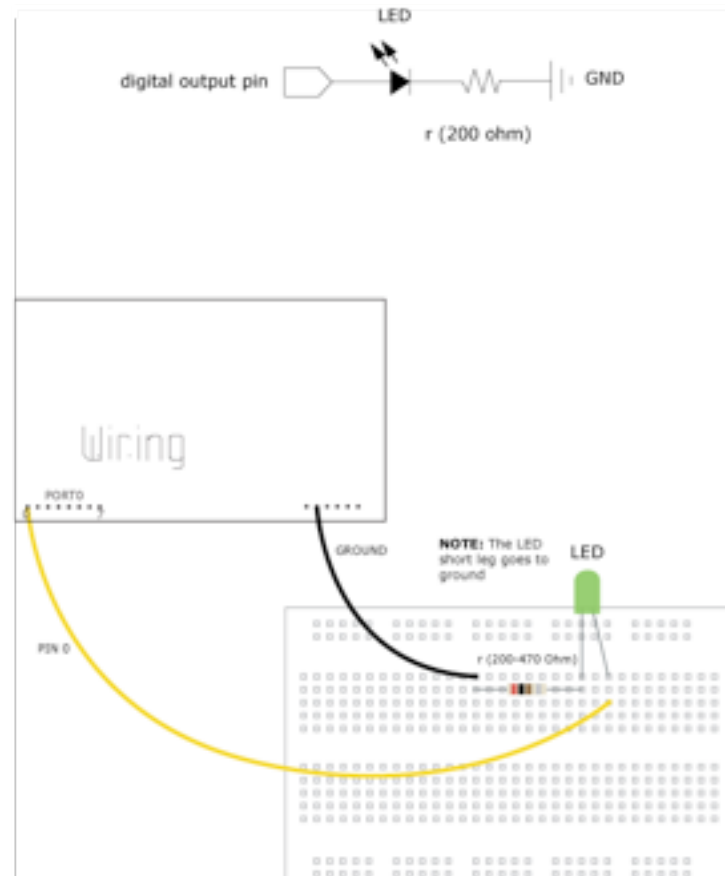
1.

Encendido de un led

Nuestro primer ejemplo será encender y apagar un simple led con una de las E/S digitales.



Esquema



Código fuente

```
int ledPin = 0; // LED connected to digital pin 0

void setup()
{
  pinMode(ledPin, OUTPUT); // set ledPin pin as output
}

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

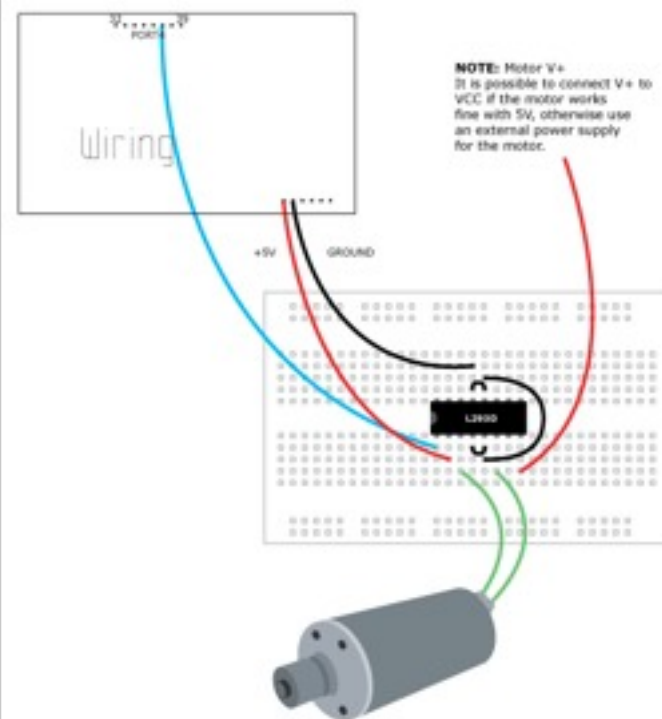
2.

Mover un motor con un H-Bridge

En el segundo ejemplo usaremos electrónica adicional que nos ayudará a mover un pequeño motor.



Esquema



Código fuente

```
int motorPin = 37; // L293D Pin En1 connected to pin PWM 37

void setup()
{
  pinMode(37, OUTPUT);
}

void loop() {
  analogWrite(motorPin, 512); // turn ON the Motor at medium speed
  delay(100);
}
```

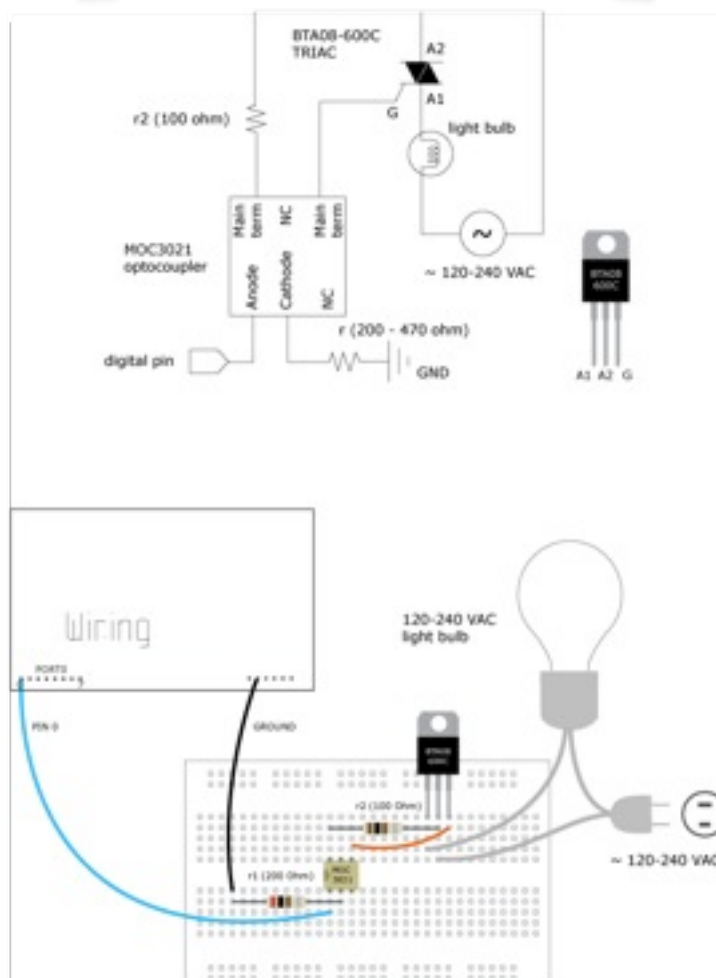
3.

Encendido de una bombilla 220VAC

Usaremos un triac y un desacoplador óptico para conmutar corrientes 220VAC



Esquema



Código fuente

```
int optocouplerPin = 0; // Optocoupler anode (+) pin connected to digital pin 0

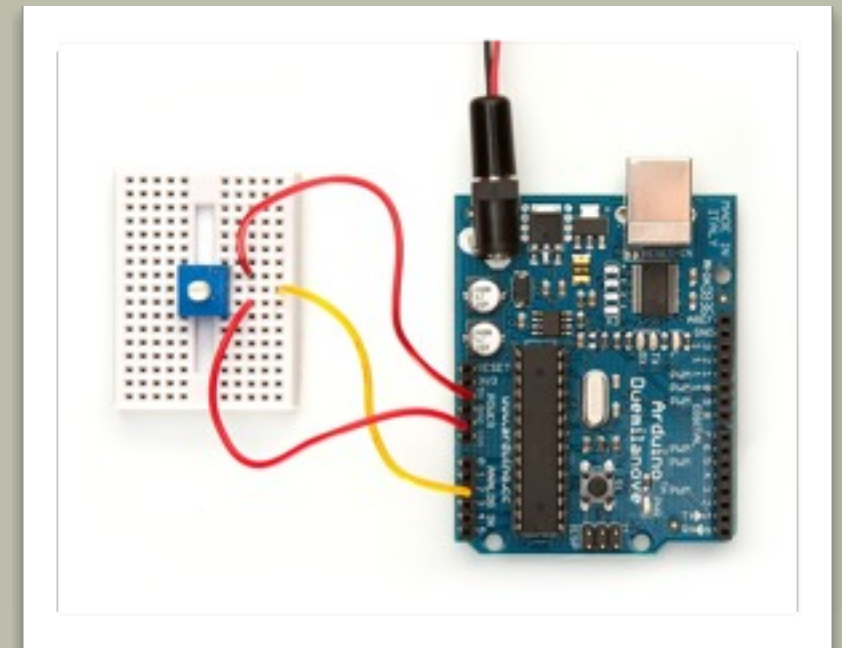
void setup()
{
  pinMode(optocouplerPin, OUTPUT); // sets the digital pin as output
}

void loop()
{
  digitalWrite(optocouplerPin, HIGH); // switch ON the triac
  delay(1000); // waits for a second
  digitalWrite(optocouplerPin, LOW); // switch OFF the triac
  delay(1000); // waits for a second
}
```

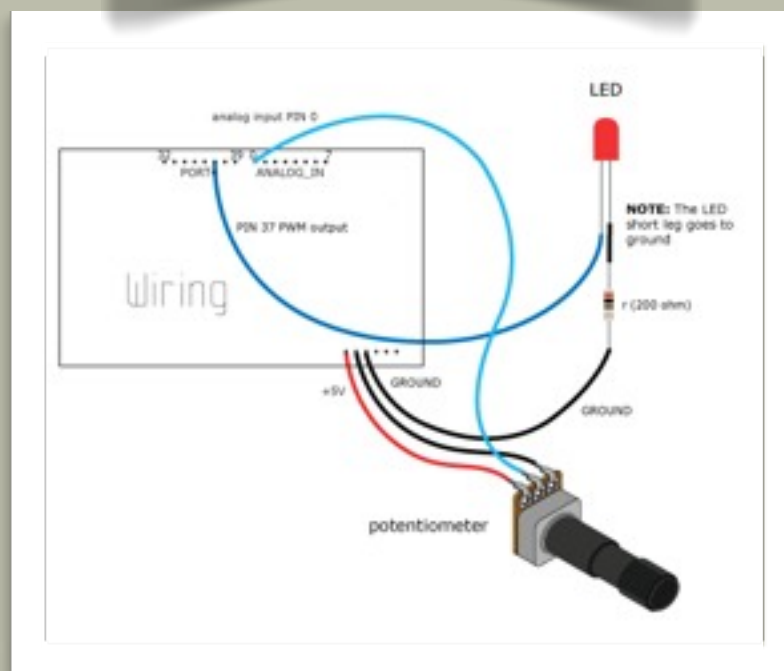
4.

Dibujado de datos a tiempo real

Usaremos la comunicación serie y las entradas analógicas para, ayudándonos del software kst, realizar un trazado en tiempo real de los datos recibidos.



Esquema



Código fuente

```
void setup() {
  Serial.begin(9600);
}

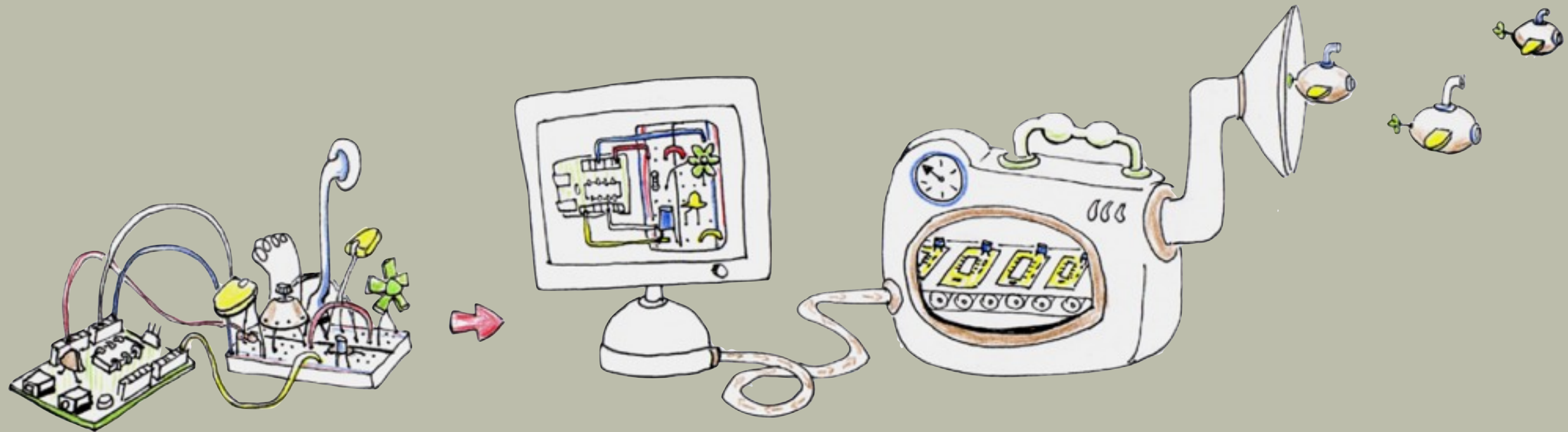
void loop() {
  int sensorValue = analogRead(0);
  Serial.println(sensorValue, DEC);
  delay(100);
}
```

Productos



I+D+i
CONSEJERÍA DE SALUD

Nos falta sólo el último paso

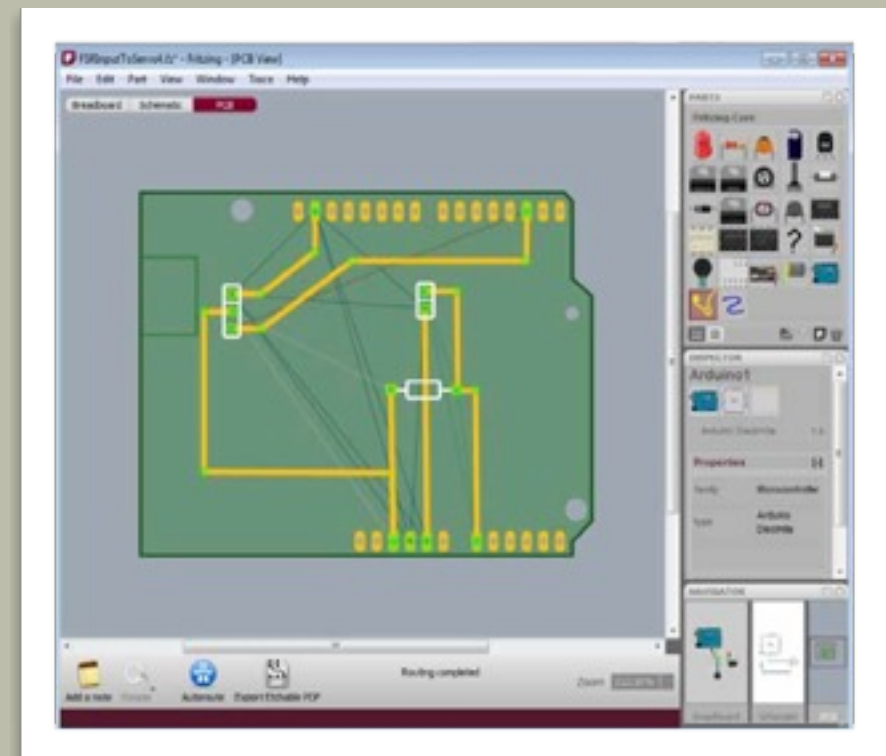
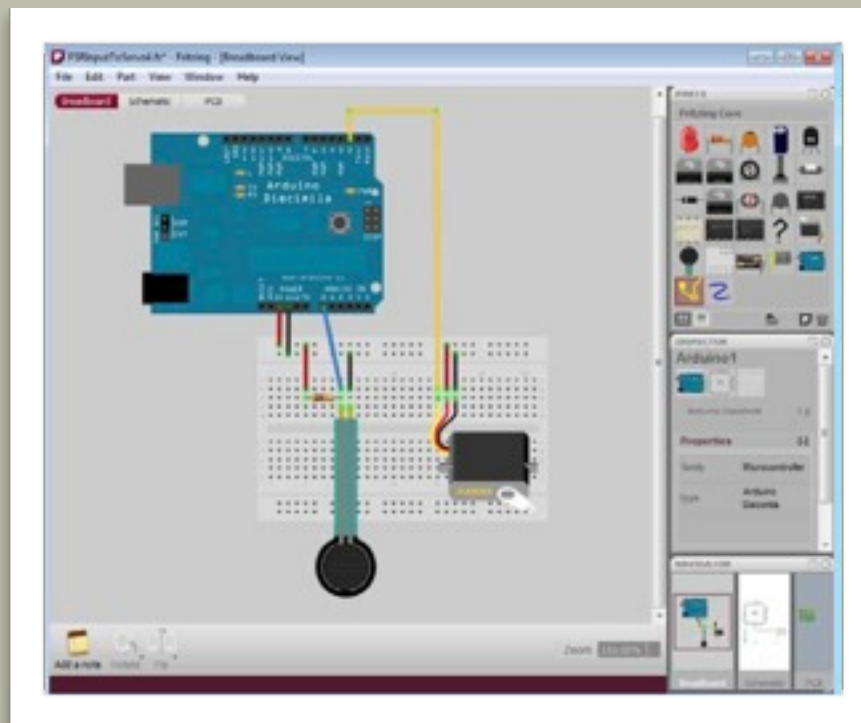


1.

Podemos ayudarnos de proyectos como fritzing.org



Utilizando como base a processing y arduino pero dando el último paso y realizando el pcb.

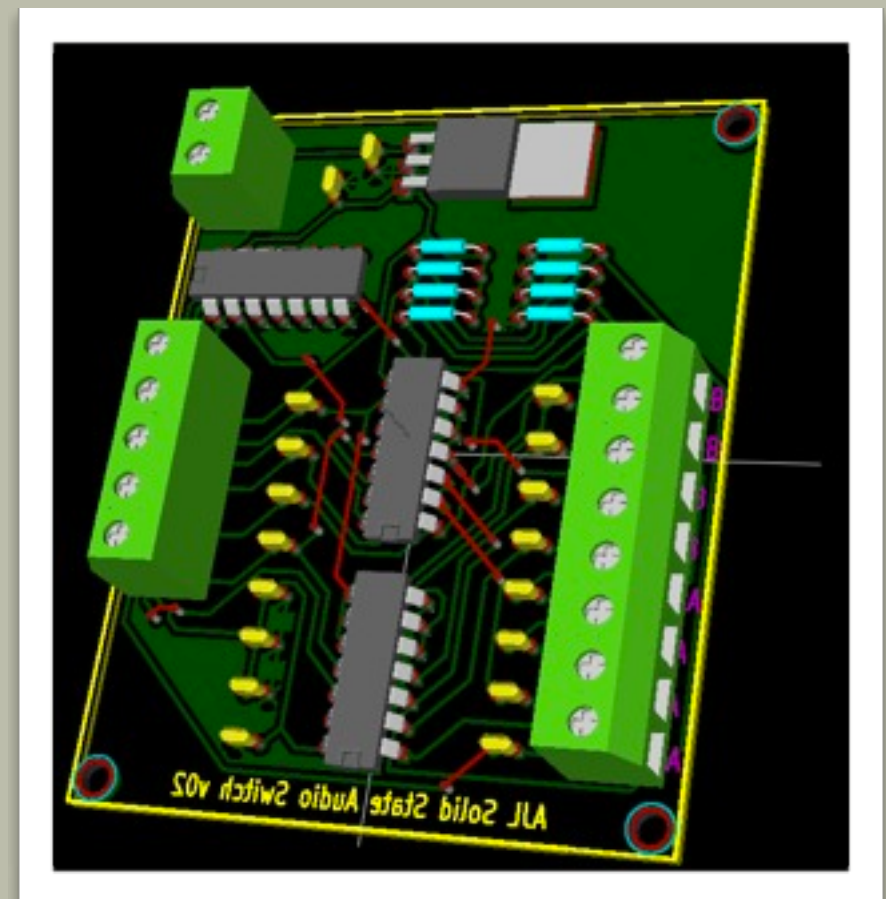
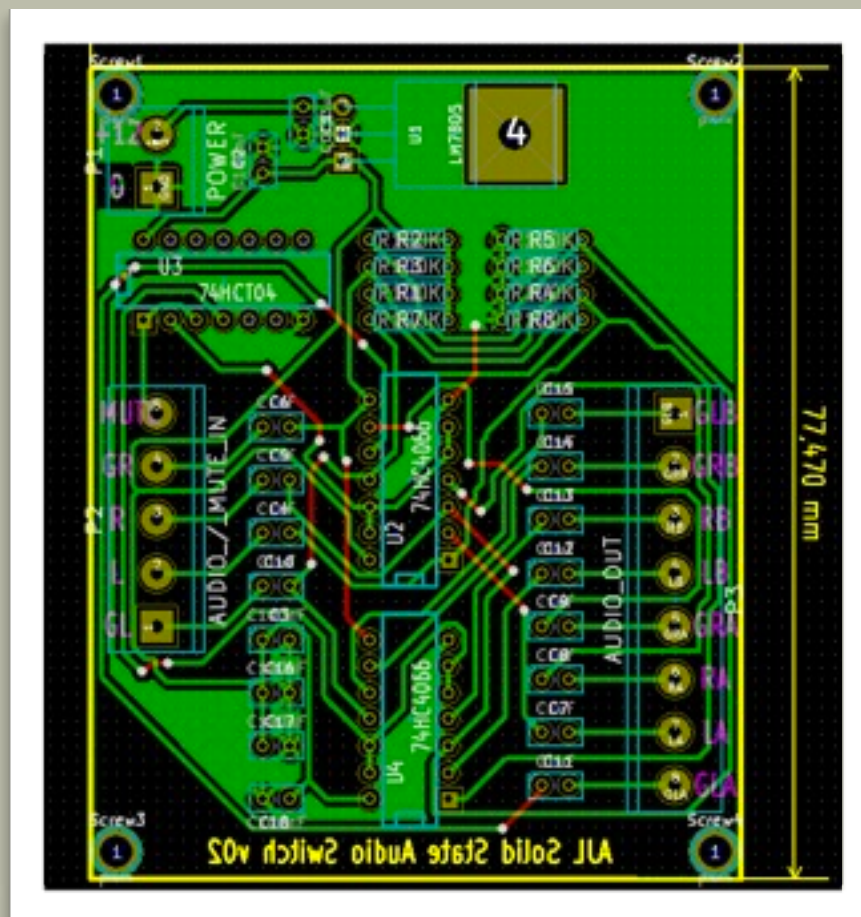


2.

Usando
cualquier
software EDA*
como kicad o
geda



Aquí tenemos la libertad de
cargar la EEPROM de
cualquier micro de la serie
ATmega con nuestro
software e incluirlo en
nuestro propio diseño

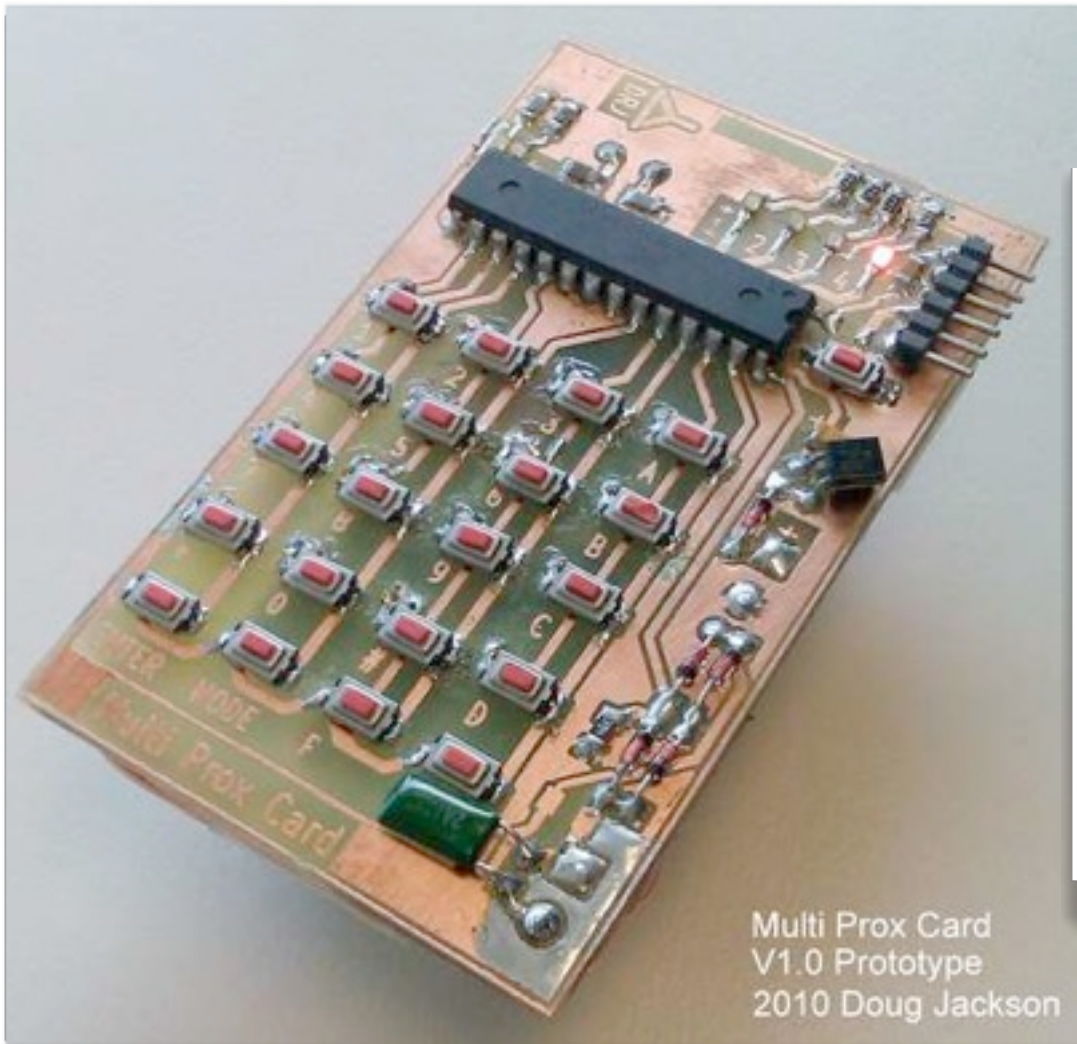


* electronic design automation

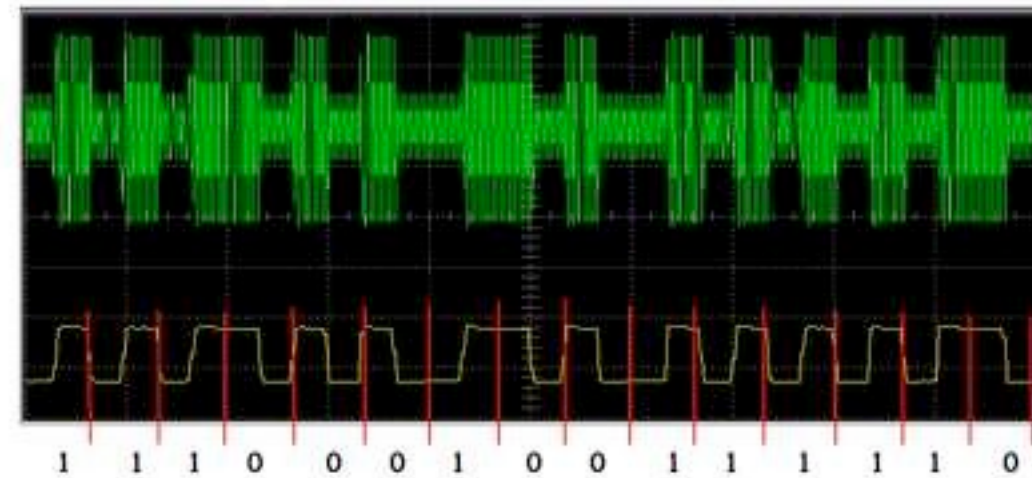
Ejemplo:

Llave universal RFID

([instructables.com drj113](http://instructables.com/drj113))



Multi Prox Card
V1.0 Prototype
2010 Doug Jackson



RFID Transmitted Signal

Modulated
125Khz Carrier

Data to
Transmit

7 Resumen



IAVANTE
CONSEJERÍA DE SALUD

Resumen

1.

Computación física

No es meter los dedos en el puerto USB de nuestro PC

2.

Arduino

Muchas grandes ideas nacen en un **bar**.

3.

Hardware

Nos vale con cualquier hardware arduino o compatible para empezar a hacer nuestros pinitos.

Resumen

4.

Software

Muy **fácil** de usar incluso para neófitos de la programación. La documentación es extensa.

5.

¡Practica!

No tengas miedo de **fallar**, yo hago electrónica de ensayo y error ;)

6.

Productos

Al final tendrás que realizar **PCBs** antes o después, así que ¡manos a la obra!

¡Gracias!



I+D+i
CONSEJERÍA DE SALUD

Más ...



gerardo.puerta@iavante.es



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



IAVANTE
CONSEJERÍA DE SALUD

Créditos

Slide Sources

1	Image Breadboard: flickr @hobonerd
3	Snapshot: wikipedia.org
4	Image Banzi: diydrones.ning.com Image David: amphibia.com.ar Image Student: flickr @stuartpilbrow
5	Image bulb: globalwarmingart.com Image miracle of science: cuboidal.org
6	Image arduino: kaosat.net
7	Snapshots: arduino.cc and wikimedia.org
8	Image physics: MCTgarcía
11	Image hardware: arduino.cc
12	Video analog signal: tronixstuff.com Video digital signal: tronixstuff.com Video PWM signal: youtube @ nkcElectronics
14	Images shields: sparkfun.com

Créditos

Slide Sources

15	Images: cooking-hacks.com
16	Image EEPROM: itp.nyu.edu Image SDRAM: wikimedia.org Image SCART: firstsing.com Image cristal: egr.msu.edu Image measures: jek-a-go-go flickr
18	Image MIT Media Lab: wikimedia.org Snapshot: wiring.org.co
27	Image 64 bits: librenet.files.wordpress.com Image serial port eertbd.com
28	Images boxes: interfacelift @ Alejandro Lopez Ulloa
29	Image play: psdgraphics.com
32	Image arduino Led: halley.cc Image H-Bridge: earthshineelectronics.com
33	Image TO-220: sterenshopusa.com Image analog Input: halley.cc
34-37	Codes and schematics: wiring.org.co
39	Image: fritzing.org

Créditos

Slide Sources

40 Snapshots: fritzing.org

42 Images: instructables.com @drj113

47 Image twitter bird: wordpress.com @seevirtual



IAVANTE
CONSEJERÍA DE SALUD