

Presentación  
Arduino  
SODAR  
Processing

## SODAR: Iniciación a Arduino + Processing

Un taller BricoLabs

ctemes eukelade Milo salvari

Asociación BricoLabs

7 noviembre / OSHWDem - 2014



ctemes, eukelade, milo, salvari Taller SODAR

Presentación  
Arduino  
SODAR  
Processing

### Agenda

- 1 Presentación
  - ¿Quienes somos?
  - Requisitos
- 2 Arduino
  - Intro
  - Montaje
  - Conceptos CPP
- 3 SODAR
  - Movimiento
  - Sensor
- 4 Processing
  - Introducción
  - Geometría



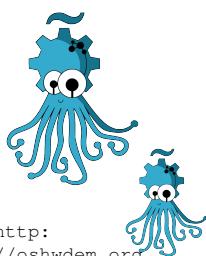
ctemes, eukelade, milo, salvari Taller SODAR

Presentación  
Arduino  
SODAR  
Processing

¿Quienes somos?  
Requisitos

## BricoLabs y la OSHWDem





<http://oshwdem.org>

ctemes, eukelade, milo, salvari Taller SODAR

Presentación  
Arduino  
SODAR  
Processing

¿Quienes somos?  
Requisitos

### Ponentes

- @ctemes
- @pepdiz
- @Milo\_1008
- @salvari



ctemes, eukelade, milo, salvari Taller SODAR

Presentación  
Arduino  
SODAR  
Processing

¿Quienes somos?  
Requisitos

### Asistentes

- ¿Quién ha programado antes?
- ¿Quién conoce el Arduino?
- ¿Quién conoce Processing?
- ¿Traéis los deberes hechos? :)

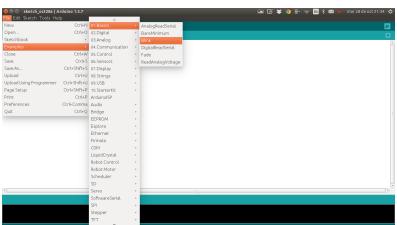


ctemes, eukelade, milo, salvari Taller SODAR

Presentación  
Arduino  
SODAR  
Processing

¿Quienes somos?  
Requisitos

### Revisar la instalación





ctemes, eukelade, milo, salvari Taller SODAR

Presentación  
Arduino  
SODAR  
Processing

Intro  
Montaje  
Conceptos CPP

### SODAR





Página Principal

Presentación  
Arduino  
SODAR  
Processing

Intro  
Montaje  
Conceptos CPP

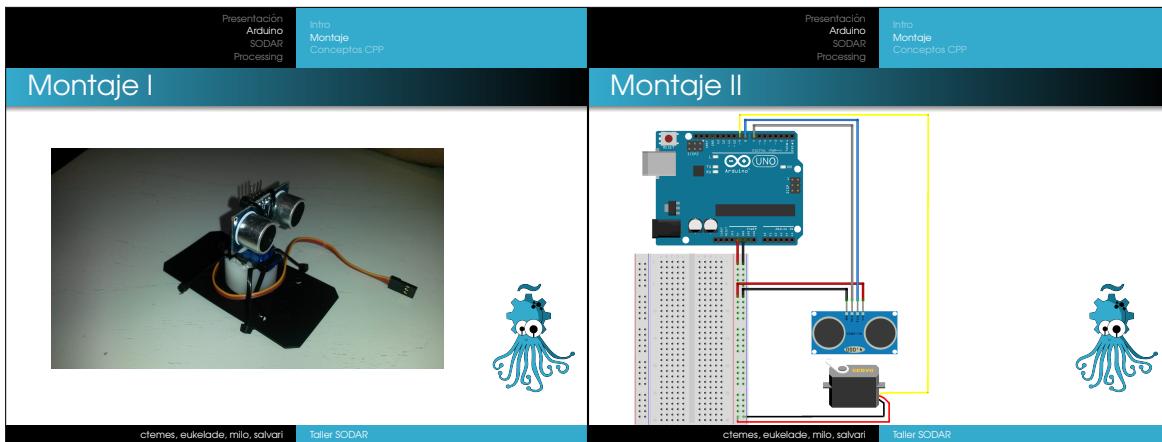
### Arduino



Foto Familia



ctemes, eukelade, milo, salvari Taller SODAR



El C de Arduino

Presentación Arduino SODAR Processing

Intro Montaje Conceptos CPP

Estructura de un programa Arduino

```
#include <Servo.h>
#define SERVO_PWM_PIN 9
Servo myservo;
//-----setup-----
// Se ejecuta una sola vez al principio del programa. O cuando el arduino se resetea.
void setup() {
}
//-----loop-----
// Se ejecuta siempre, hasta el fin de los tiempos :-)
void loop() {
}
```

Presentación Arduino SODAR Processing

Intro Montaje Conceptos CPP

ctemes, eukelade, milo, salvati Taller SODAR

ctemes, eukelade, milo, salvati Taller SODAR

Función

Presentación Arduino SODAR Processing

Intro Montaje Conceptos CPP

```
int medida(int intentos){
}

void canta(){
    Serial.println("La, la, la!");
}
```

Presentación Arduino SODAR Processing

Intro Montaje Conceptos CPP

Serial

Presentación Arduino SODAR Processing

Intro Montaje Conceptos CPP

```
Serial.begin(9600);
Serial.print("Hola"); // Sin cambiar de linea
Serial.println("mundo"); // Con retorno de linea
Serial.print(10, DEC); // Con formato (DEC, BIN, OCT, HEX)
Serial.println(10.1234, 2); // Imprime 10.12
```

ctemes, eukelade, milo, salvati Taller SODAR

ctemes, eukelade, milo, salvati Taller SODAR

Movimiento Servo

Presentación Arduino SODAR Processing

Movimiento Servo

#include <Servo.h>

Barridos

Presentación Arduino SODAR Processing

Movimiento Servo

ctemes, eukelade, milo, salvati Taller SODAR

ctemes, eukelade, milo, salvati Taller SODAR

Presentación Arduino <b>SODAR</b> Processing	Movimiento Sensor
<h1>Una solución</h1> <pre>#define ANGULO_MIN 0 // angulo minimo del barrido (en grados) #define ANGULO_MAX 180 // angulo maximo del barrido (en grados) #define PASO_ANGULO 1 // paso de barrido  #define ATRAS -1 #define ADELANTE 1  loop() {     delay(50); // espera 50 milisegundos     miservo.write(angulo); // avanza el servomotor al angulo indicado      if (angulo &gt;= ANGULO_MAX) {direccion = ATRAS;}     if (angulo &lt;= ANGULO_MIN) {direccion = ADELANTE;}      angulo += direccion * PASO_ANGULO; // incrementa el angulo de  // barrido un paso de barrido }</pre> 	

Presentación	Movimiento										
Arduino	Sensor										
Processing											
<h1>Sensor ultrasonidos</h1> 											
<p><b>Sensor de distancia HC-SR04</b></p> <p><b>Electric Parameter</b></p> <table border="1"><tbody><tr><td>Working Voltage</td><td>DC 5 V</td></tr><tr><td>Working Current</td><td>15mA</td></tr><tr><td>Working Frequency</td><td>40Hz</td></tr><tr><td>Max Range</td><td>4m</td></tr><tr><td>Min Range</td><td>2cm</td></tr></tbody></table>	Working Voltage	DC 5 V	Working Current	15mA	Working Frequency	40Hz	Max Range	4m	Min Range	2cm	
Working Voltage	DC 5 V										
Working Current	15mA										
Working Frequency	40Hz										
Max Range	4m										
Min Range	2cm										

**Presentación**  
Arduino  
SODAR  
Processing

**Movimiento**  
Sensor

## Protocolo

**Timing Diagram**

Trigger Input to Module

Sonic Burst from Module

Echo Pulse Output to User Timing Circuit

10µS TTL

8 Cycle Sasic Burst

Input TTL lever signal with a range in proportion

Diagrama de señales

Presentación  
Arduino  
**SODAR**  
Processing

Movimiento  
Sensor

# Función Medida

## Una función de bajo nivel

```
/*
-----  
  setup  
    Se ejecuta una sola vez al principio del programa. O cuando el arduino  
    se reinicia  
-----  
void setup() {  
  pinMode(TRIGGER_PIN, OUTPUT); // pin trigger es salida  
  pinMode(ECHO_PIN , INPUT); // pin echo es entrada  
  
  Serial.begin(9600); // Abrimos el puerto serie  
}  
  
/*
-----  
  loop  
    Se ejecuta siempre repetidamente, hasta el fin de los tiempos :-)  
-----  
void loop() {  
  digitalWrite(TRIGGER_PIN, LOW); // Preparamos el sensor  
  delayMicroseconds(5); // Nivel bajo para estabilizar  
  // garantizamos 5 microsegundos  
  
  digitalWrite(TRIGGER_PIN, HIGH); // Enviamos un pulso de 10 microsegundos  
  delayMicroseconds(10);  
  digitalWrite(TRIGGER_PIN, LOW);  
  
  time = pulseIn(ECHO_PIN, HIGH); // Leemos el pulso viene modificado en  
  ctremes, eukeladlo, milo, salvan  
}
```

<p>Presentación Arduino SODAR Processing</p> <h1>NewPing</h1> <pre>#include &lt;NewPing.h&gt;  NewPing sonar(TRIGGER_PIN,               ECHO_PIN,               DISTANCIA_MAXIMA); // Creamos un objeto sensor  int cm = sonar.ping_cm(); // Medimos</pre>	<p>Movimiento Sensor</p> 
--	--

Presentación  
Arduino  
**SODAR**  
Processing

Movimiento  
Sensor

# Solución SODAR

```
/*
obtenerDistanciaEnviar
esta función obtiene la distancia a la que se encuentra un objeto
y envía por el puerto serie el angulo y la distancia del objeto encontrado
*/
int obtenerDistanciaEnviar(int angulo) {
    int cm = sonar.ping_cm();
    Serial.print(angulo, DEC); // obtiene la distancia en cm
    Serial.print(cm, DEC); // envia por puerto serie la distancia y el angulo
    Serial.print("*");
    Serial.println(cm, DEC); // formato;
    Serial.println("distanzia<LINEA>");
}

#define ANGULO_MIN 0 // angulo minimo del barrido (en grados)
#define ANGULO_MAX 180 // angulo maximo del barrido (en grados)
#define PASO_ANGULO 1 // paso de barrido

#define ATRAS 1
#define ADELANTE 1

loop() {
    delay(50);
    miservo.write(angulo); // espera 50 milisegundos
                           // avanza el servomotor al angulo indicado

    obtenerDistanciaEnviar(angulo); // obtiene la distancia y envia
                                    // los datos por el puerto serie
}

ctemes, eukelade, milo, salvani
```

Presentación  
Arduino  
SODAR  
Processing

Introducción  
Geometría

# Estructura de un programa

---

```
import processing.serial.*;

// ---- variables ----

int numPuntos=0;
String contenidoSerie;
Serial miPuerto;

// -----
setup {
    Se ejecuta una sola vez al principio del programa
}

void setup() {
    background(255);
    size(MAXX, MAXY);
}

// -----
draw {
    Se ejecuta eternamente hasta el fin de los tiempos
}

void draw() {

}


```

---



ctemes, eukelade, milo, salvati

Taller SODAR

Pantalla de Processing

Presentación  
Arduino  
SODAR  
Processing

Introducción  
Geometría

Coordenadas

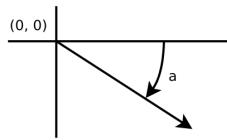
(0, 0)

x

y

Coordenadas en Processing

## Pantalla de Processing Ángulos



$$\beta = 2\pi - \text{radians}(\alpha)$$

