

└ Agenda

1. Recordar que el curso es de iniciación
2. Va a ser un curso intenso, descanso cortito
3. Comentar duración total del curso

Taller SODAR

- └ Presentación

- └ ¿Quienes somos?

- └ BricoLabs y la OSHWDem



1. BricoLabs: Asociación, Domus, Difusión del hardware y software libre, Tecnófilos, Divulgación, wiki,
2. OSHWDem evento barcamp

Taller SODAR

└ Presentación

└ ¿Quienes somos?

└ Ponentes

- @ctemes
- @pepdiz
- @Mto_1008
- @salvari

1. Para cualquier duda podéis contactar con nosotros en nuestros twitters

- └ Presentación

- └ ¿Quienes somos?

- └ Asistentes

- └ ¿Quién ha programado antes?

1. No hay mucho más que añadir

Taller SODAR

└ Presentación

└ ¿Quienes somos?

└ Asistentes

- ¿Quién ha programado antes?
- ¿Quién conoce el Arduino?

1. No hay mucho más que añadir

- └ Presentación

- └ ¿Quienes somos?

- └ Asistentes

- ↳ ¿Quién ha programado antes?
- ↳ ¿Quién conoce el Arduino?
- ↳ ¿Quién conoce Processing?

1. No hay mucho más que añadir

- └ Presentación

- └ ¿Quienes somos?

- └ Asistentes

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

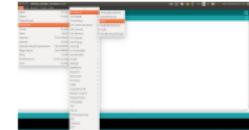
1. No hay mucho más que añadir

Taller SODAR

- └ Presentación

- └ Requisitos

- └ Revisar la instalación



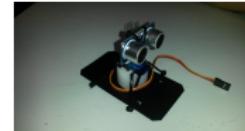
1. Separamos a los que tienen hecha la instalación y los que no, los primeros prueban el Blink y a los segundos los ayudamos.
2. Hora límite 17.00
3. Blink: Fichero::Ejemplos::Básicos::Blink

Taller SODAR

└ Arduino

└ Intro

└ SODAR



1. SODAR, igual que RADAR SONic Detection and Range
2. Básicamente dos partes movimiento y sensor
3. Una tercera parte será la estación de usuario
4. *****Corregir GRAFICOS*****

Taller SODAR

└ Arduino

└ Intro

└ SODAR



1. SODAR, igual que RADAR SONic Detection and Range
2. Básicamente dos partes movimiento y sensor
3. Una tercera parte será la estación de usuario
4. *****Corregir GRAFICOS*****

Taller SODAR

└ Arduino

└ Intro

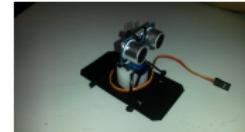
└ Arduino



Página Principal

Foto Familia

1. Empezamos con el Arduino
2. Año 2005
3. Procesadores de la familia Atmel AVR (AtMega)
4. IDE (basado en processing)
5. CPP con librerías variadas
6. Familia Arduino
7. Ecosistema asociado: shields, sensores, etc



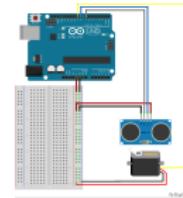
1. ***** CORREGIR FOTO *****
2. Tenemos que fijar el servo (cinta doble cara a la BB)
3. Tenemos que fijar el sensor a la corona del servo
4. Y por último montar la corona

Taller SODAR

└ Arduino

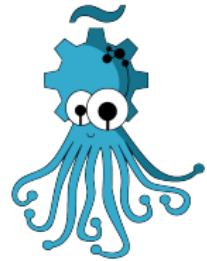
└ Montaje

└ Montaje II



1. ¡Mucho ojo con los cables del servo!
2. Rojo: 5 volt
3. Marron: tierra
4. Naranja: Señal

El C de Arduino



Taller SODAR

└ Arduino

└ Conceptos CPP

└ Estructura de un programa Arduino

```
Estructura_MiProyecto.ino
Archivo: MiProyecto.ino
Última ejecución: 08/11/2014 10:56:58
Sistema operativo: Arduino IDE 1.6.5

// setup()
// Función que se ejecuta al principio del programa. El módulo es iniciado
// en este punto.
void setup() {
}

// loop()
// La función loop() se ejecuta siempre, hasta el fin de los tiempos :(
void loop() {
}
```

1. Tres partes
2. Primera parte imports y variables – Tiempo de compilación
3. Tiempo de ejecución – dos partes setup y loop
4. setup al arrancar, después de un reset hay un arranque
5. loop para siempre jamás (mentira)

```
int medición(int instrumento){  
    //...  
    void setup();  
    Serial.println("Hello world");  
}
```

1. Explicamos una función Arduino

```
Serial.begin(9600);  
Serial.print("Hello world"); // Escribir en linea  
Serial.println("Hello world"); // Con retorno de linea  
Serial.print(10, DEC); // Escribir 100, 816, 65, 65, 65  
Serial.println(10, DEC, 2); // Imprime 10,11
```

1. Explicamos como se usa el Serial

Taller SODAR

└ SODAR

└ Movimiento

└ Servo



1. Que es un servo? Un motor y un pequeño circuito
2. Protocolo de comunicación, anterior a la era digital
3. PWM: 20ms de periodo para los servos
4. Tipos de servos, los nuestros son de 180 grados, 1,6 kg

Taller SODAR



```
#include <servo.h>
Servo myservo;
myservo.attach(0);
myservo.write(100);
```

1. Importar Librería
2. Declarar objeto servo
3. Asociar PIN
4. Ordenar una posición

Taller SODAR

└ SODAR

 └ Movimiento

 └ Barridos



1. ¿Como os imagináis que debe moverse el radar?

Taller SODAR

└ SODAR

└ Movimiento

└ Una solución

```
Una solución

positive 900000_000 0 // angulo máximo del servomotor (en grados)
positive 900000_180 0 // angulo mínimo del servomotor (en grados)
positive 900000_90 0 // paso de servomotor

positive 1000000_1 0
positive 1000000_0 0

longint i;
longint j(100);
servoServo1.attach(9); // angulo 90 milisegundos
servoServo2.attach(10); // angulo 90 milisegundos
// angulo en 900000_000 | dimension = 400000;
// angulo en 900000_180 | dimension = 400000;
// angulo en 900000_90 | dimension = 200000;
angulo = dimension * 5000_00000; // incremento al angulo de
// 1000000_000
// angulo en 1000000_100 | dimension = 100000;
```

1. Definimos un paso
2. Controlamos el ángulo
3. Usamos el propio loop del Arduino
4. Vemos nuestra solución en el IDE proyectado y comentamos

Taller SODAR

└ SODAR

└ Sensor

└ Sensor ultrasonidos



Sensor de distancia HC-SR04

1. Como funciona el sensor, tren de pulsos y mide el tiempo en que tarda en recibir el eco

Taller SODAR

- └ SODAR
- └ Sensor
 - └ Sensor ultrasonidos

Sensor ultrasonidos	
Electric Parameter	
Working Voltage	DC 3.7
Working Current	15mA
Working Frequency	40KHz
Max Range	60m
Min Range	1m
Measuring Angle	15 degree
Trigger Input Signal	High TTL pulse
Echo Output Signal	Low TTL, level signal and the range is proportional
Dimensions	47*29*12mm

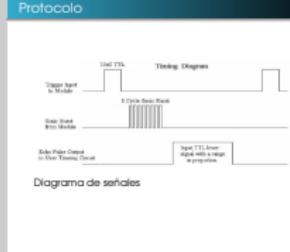
1. Como funciona el sensor, tren de pulsos y mide el tiempo en que tarda en recibir el eco

Taller SODAR

└ SODAR

└ Sensor

└ Protocolo



1. Explicamos el protocolo de señales del servo

Taller SODAR

└ SODAR

└ Sensor

└ Función Medida

```
/* Función que lee una sola vez el principio del programa. Si existe el archivo
 * config.txt, se lee su contenido y se ejecuta la función correspondiente.
 */
void leerConfig() {
    ifstream config("config.txt");
    if(config.is_open()) {
        string linea;
        while(getline(config, linea)) {
            if(linea == "Serial.read(1000);") // lee datos en serial
                Serial.read(1000);
            else if(linea == "Serial.write(1000);") // envia datos en serial
                Serial.write(1000);
        }
    }
}
```

1. Una función de bajo nivel

Taller SODAR

- └ SODAR
 - └ Sensor
 - └ Función Medida

```
/*  
 * La función mide una distancia, hasta el fin de los tiempos xD  
 */  
  
void loop() {  
    digitalWrite(10,LOW); //PA10, AIN01  
    delayMicroseconds(10);  
    digitalWrite(10,HIGH); //PA10, AIN01  
    delayMicroseconds(10);  
    analogRead(10); //PA10, AIN01  
    time = pulse(D10,PA10, AIN01);  
    dist = time / 29; //d  
    Serial.print(dist);  
    Serial.println();  
}  
  
delay(1000);
```

1. Una función de bajo nivel

Taller SODAR

- SODAR

- Sensor

- NewPing



1. Explicamos como se usa la librería NewPing