

# CONTROL DE UN SERVOMOTOR DESDE APP INVENTOR VIA BT

## CONTROLANDO ARDUINO DESDE DISPOSITIVOS MÓVILES

Jose Luis Núñez  
José Pujol

CEP Sevilla 2019

# ÍNDICE



1. Finalidad del sistema
2. Arduino
3. Aplicación MIT APP Inventor
4. Propuestas de mejora



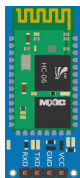
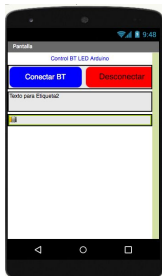
# FINALIDAD DEL SISTEMA



Controlar un servomotor de posición conectado a Arduino desde el teléfono móvil mediante un slider y conexión bluetooth

ENTRADAS

Dispositivo móvil  
Módulo Bluetooth



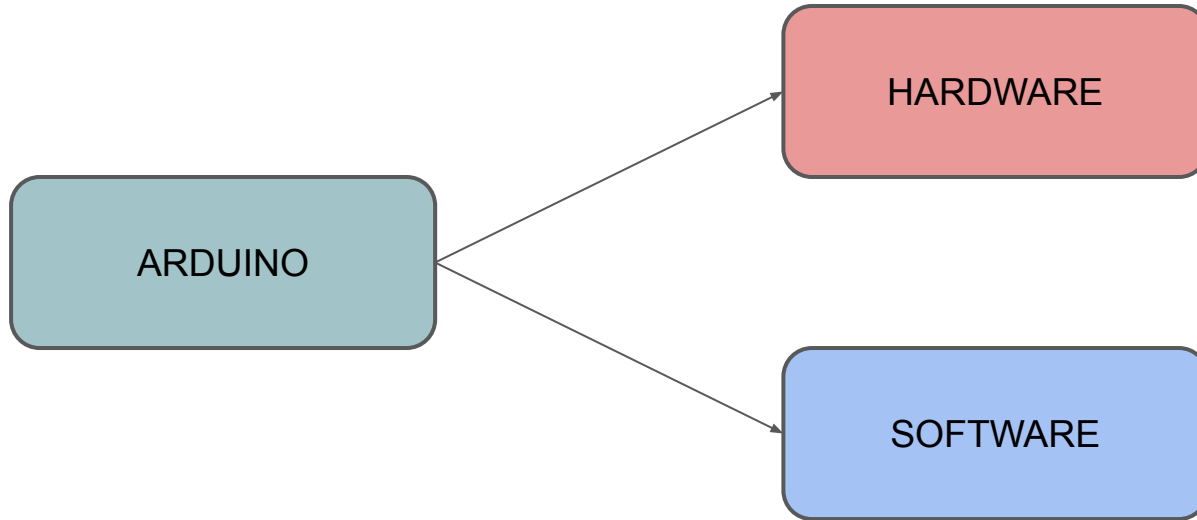
ARDUINO

SALIDAS

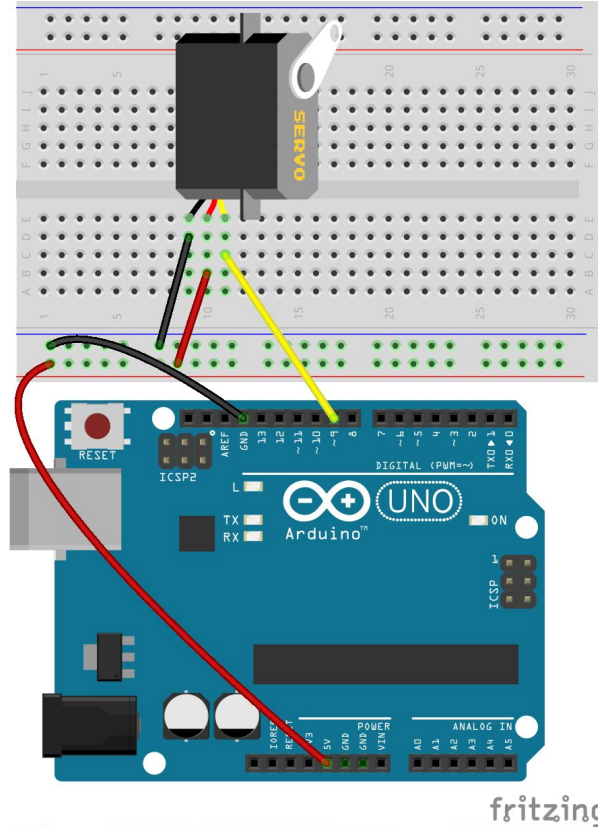
Servomotor (D)



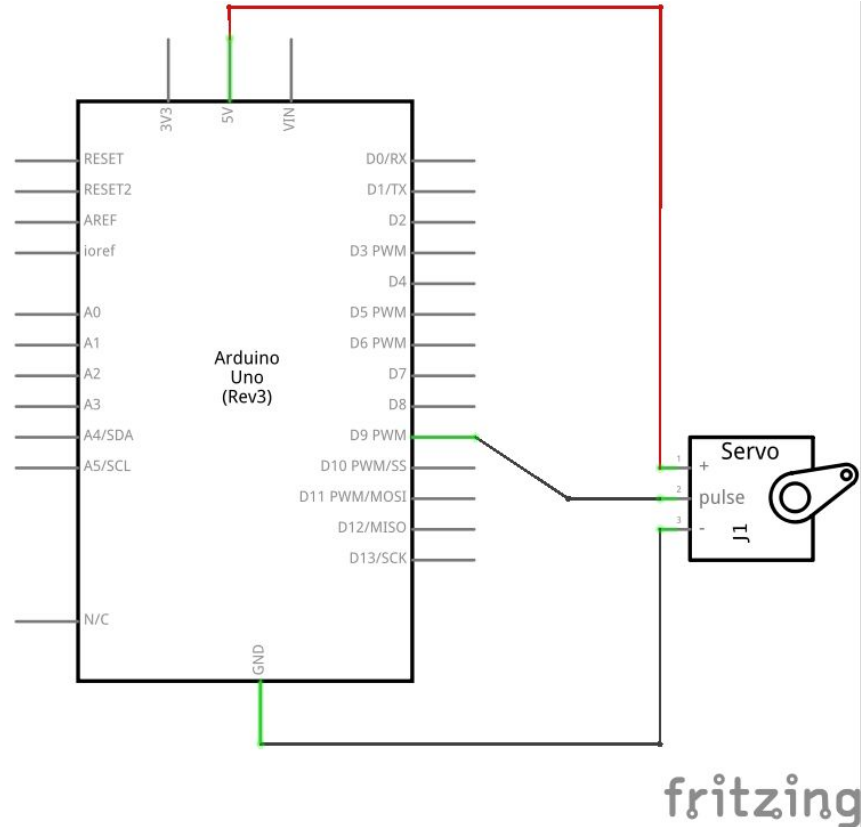
# ARDUINO



# Hardware Servo comunicación PC



fritzing

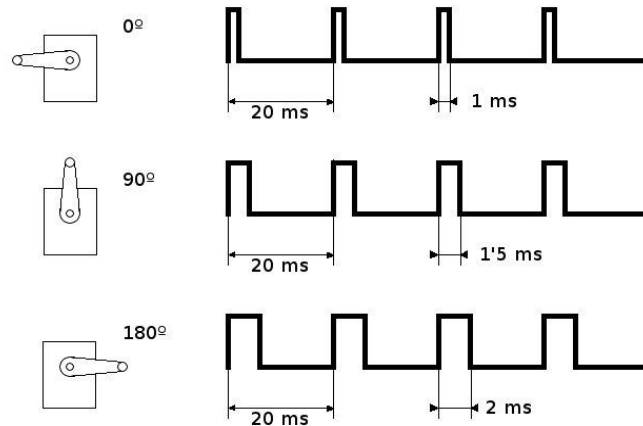


fritzing

# Libreria Servo.h

`myservo.write(0-180);`

Escribe un pulso entre 1 y 2 ms en un período de 20 ms para posicionar el servo desde 0º a 180º. Esto simplifica la programación pues en una sola orden se ha programado la secuencia de realización del pulso.



# Software

```
#include <Servo.h> // incluimos la libreria Servo.h
Servo myservo;    // Creamos el objeto tipo servo de nombre myservo

// Variables para recibir y almacenar datos
int incomingByte;  // variable para leer los bytes de entrada
int value = 0;     // variable para almacenar el valor numerico
int angle = 0;     // variable para enviar los grados al servomotor

void setup() {
    // inicializamos el led como pin digital salida
    Serial.begin(9600);
    myservo.attach(9); // attaches the servo on pin 9 to the servo object
}
```

# Software

```
void loop() {  
  // comprobamos si hay datos de entrada  
  if (Serial.available() > 0) {  
    // lectura del byte mas antiguo del buffer serial  
    incomingByte = Serial.read();  
    if (incomingByte >= '0' && incomingByte <= '9') {  
      //Acumula los datos numericos multiplicando por 10 el valor acumulado  
      value = (value * 10) + (incomingByte - '0'); // Resta 48 que es el valor decimal del 0 ASCII  
    }  
    else if (incomingByte == '>') // uso > como finalizador  
    {  
      angle = value; // Guarda el valor en la variable angle  
      Serial.println(angle); // imprime angle por monitor serie  
      value = 0; // Dejamos lista la variable para volver a escribir en ella  
    }  
  }  
  myservo.write(angle); // Escribimos el valor angle en el servo  
  delay(15); // damos tiempo al servo a posicionarse  
}
```

[Código](#)

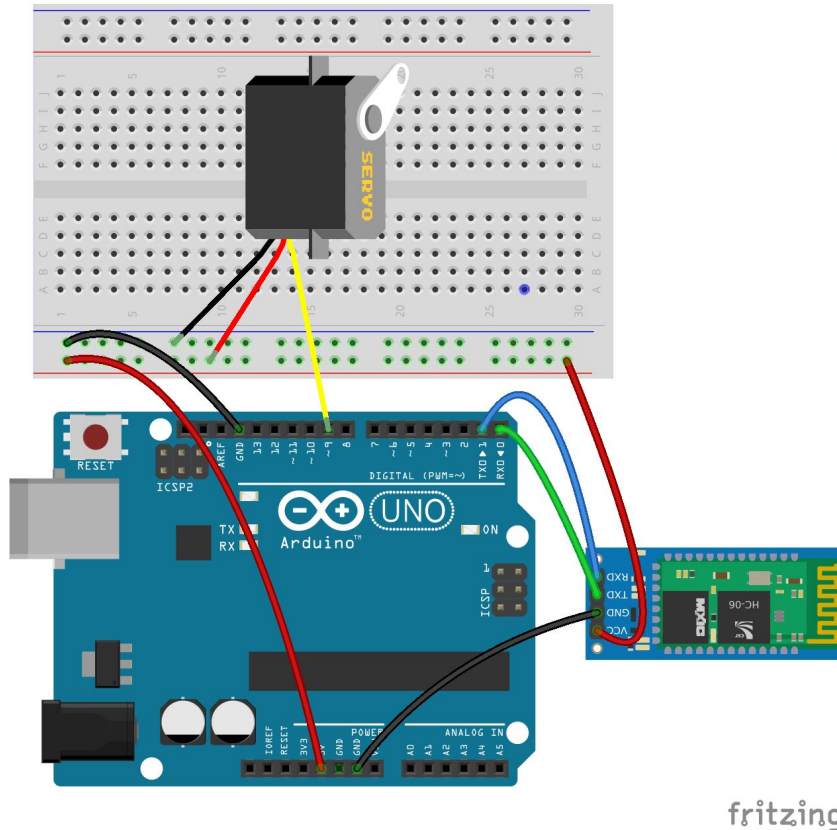


# Propuesta de actividades

---

- Controlar el servomotor desde el teclado del PC

# Hardware comunicación BT



HC06	ARDUINO
GND	GND
5V	5V
Tx(entr)	10
Rx(sal)	11

## Nota:

No podemos tener el Bluetooth conectado a los pines Rx, Tx en el momento que cargamos el programa

# APLICACIÓN APP INVENTOR



BT\_LED\_Servomotor Screen1 Añadir ventana Eliminar ventana Diseñador Bloques

**Paleta**

Search Components...

**Interfaz de usuario**

- Botón
- CasillaDeVerificación
- SelectorDeFecha
- Imagen
- Etiqueta
- SelectorDeLista
- VisorDeLista
- Notificador
- CampoDeContraseña
- Deslizador
- Desplegable
- Switch
- CampoDeTexto
- SelectorDeHora
- VisorWeb

**Disposición**

**Medios**

**Dibujo y animación**

**Mane**

**Visor**

☐ Mostrar en el Visor los componentes ocultos

Pantalla

Control BT LED Arduino

Conectar BT Desconectar

Texto para Etiqueta2

**Componentes**

- Screen1
  - Etiqueta1
  - DisposiciónHorizontal1
    - SelectordeLista1
    - Botón1
  - HorizontalArrangement1
    - Etiqueta2
  - HorizontalArrangement2
    - Slider1
  - ClienteBluetooth1

**Propiedades**

Slider1

ColorIzquierda  
Por defecto

ColorDerecha  
Por defecto

Ancho  
Ajustar al contenedor...

ValorMáximo  
180

ValorMínimo  
0

ThumbEnabled  
☒

PosiciónDelPulgar  
30.0

Visible  
☒

Cambiar nombre Borrar

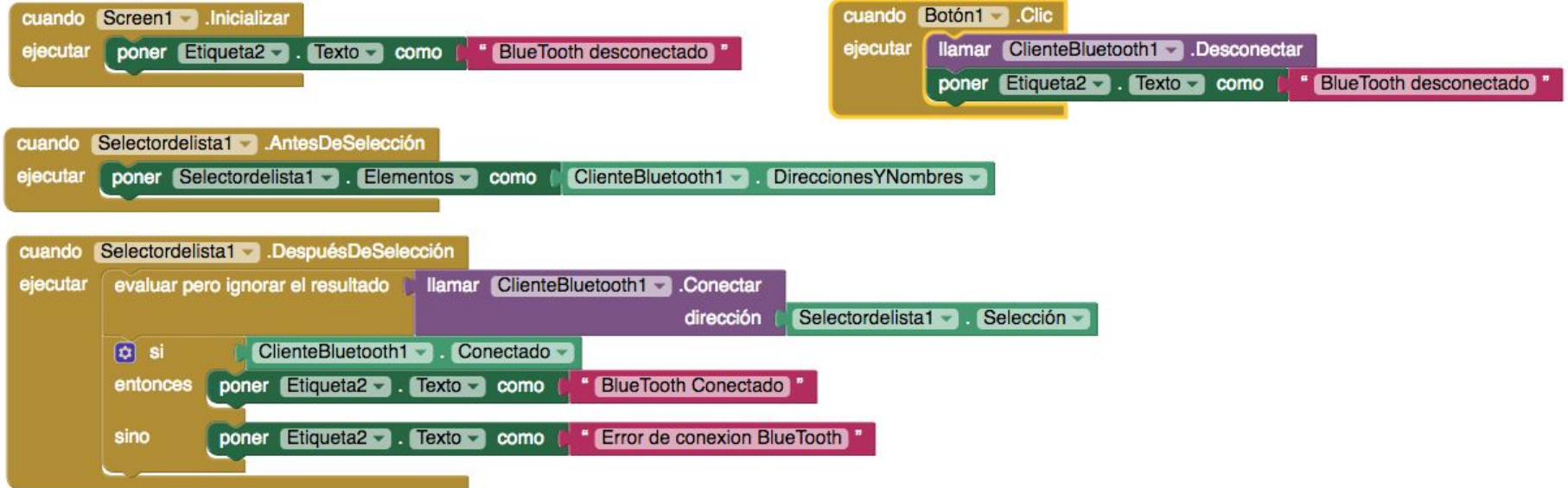
Medios

Subir archivo...

Componentes no visibles

ClienteBluetooth1

# Configuración del selector bluetooth

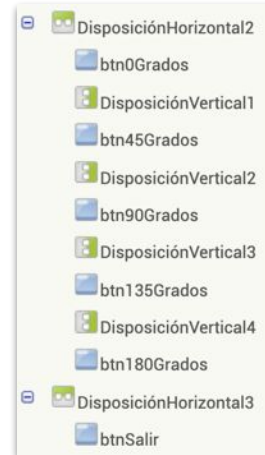


# Envío de información por BT



# Propuesta de actividades

- Replicar el sistema planteado
- Añadir botones y disponerlos en nuestra app que permita mover el servo según los grados especificados:



# LICENCIA



Esta guía se distribuye bajo licencia Reconocimiento-CompartirIgual Creative commons 4.0

Las diapositivas son obra de Jose Pujol y Jose Luis Núñez creadas para el curso “Controlando Arduino desde el teléfono móvil” para el CEP de Sevilla y han sido elaboradas usando parte del material elaborado para el curso “Tech Project: Arduino en el aula” que fue realizado por Jose Antonio Vacas y Jose Pujol en colaboración con Avante s.l.

