

## **Sistemas de Tempo Real: Microcontroladores 2013/14**

### ***Práctica 4: Sistema de Pan&Tilt controlado mediante joystick***

#### **Objetivo**

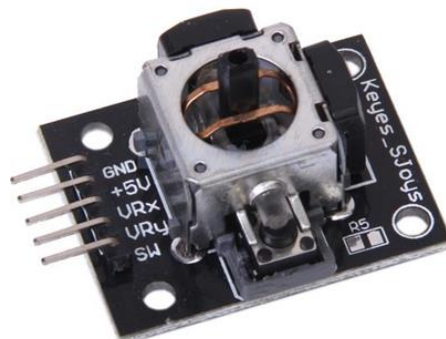
Aprender a manejar un joystick y servomotores. Asimismo, se verá como activar y gestionar las interrupciones externas que lleguen a través de pines al Arduino.

#### **El Joystick**

El joystick que se utilizará está constituido por dos potenciómetros y un botón. Los potenciómetros no son más que resistencias variables que cambiarán de valor en función de las posiciones vertical y horizontal del joystick.



El conexionado del potenciómetro es muy sencillo. Consta de 5 líneas: 5V, GND, VRx (voltaje horizontal de salida), VRy (voltaje vertical de salida) y SW (estado del switch). VRx y VRy van a pines analógicos y SW irá conectado a un pin digital.



De estas líneas hay que tener en cuenta que VRx y VRy darán una salida a  $VCC/2$  (i.e. 2,5V) cuando el joystick esté centrado<sup>1</sup>. SW estará en abierto normalmente, pero en el momento que se puse se conectará a GND. Para garantizar que un pin digital lea un '1' cuando está en abierto y un '0' al pulsarse, es necesario hacer uso de la resistencia de pull-up interna del pin al que esté conectado SW (explicado en Práctica 3).

Para realizar la lectura de los potenciómetros y el botón, es preciso configurar los pines correctamente y utilizar las funciones *analogRead* y *digitalRead* sobre estos:

```
int vertical, horizontal, boton;

[...]

vertical = analogRead(VERT); // en el rango 0-1023
horizontal = analogRead(HORIZ); // en el rango 0-1023
boton = digitalRead(SEL); // estará a HIGH (1) si no está pulsado o a LOW (0) si sí lo está

[...]
```

### Detección mediante interrupciones

Existen dos formas de detectar los cambios en el joystick: mediante *polling* (consultando constantemente el estado controlando los cambios que se produzcan) y mediante interrupciones (en cuanto se produzca un cambio se ejecuta una función concreta). En la gran mayoría de los casos la detección mediante interrupciones es bastante más eficiente, aunque tiene el inconveniente de que la ISR (*Interruption Service Routine*) tiene que ser lo más corta en tiempo posible.

En este caso haremos uso de las interrupciones para detectar cuándo se pulsa el botón del joystick y de polling para leer la posición de los potenciómetros del joystick. La implementación del segundo caso es directa a partir del código mostrado anteriormente, mientras que en el primer caso, será necesario hacer uso de la función *attachInterrupt(número, función\_a\_llamar, tipo\_de\_cambio)* (explicado en las transparencias de teoría). Es importante tener en cuenta que la interrupción 0 hace uso del pin 2 mientras que la 1 depende del pin 3.

### Sistema de Pan&Tilt

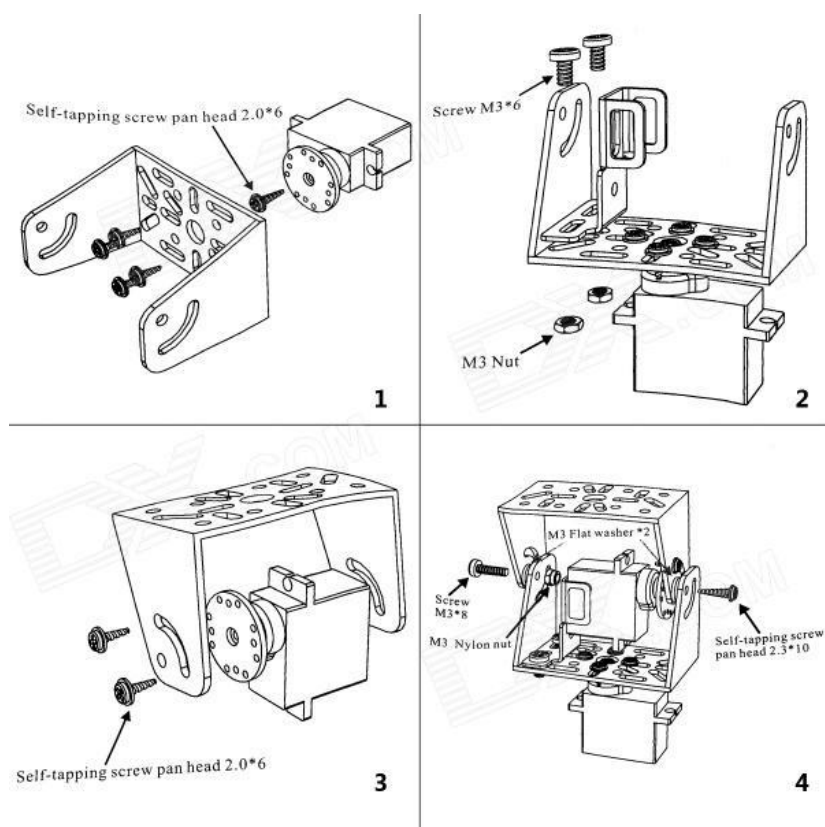
Un sistema de *pan&tilt* nos permite controlar la posición de un elemento (e.g. brazo mecánico, cámara, etc...) mediante dos servomotores.

---

<sup>1</sup> Esto es cierto en la teoría, pero en la práctica suele variar. Lo normal es calibrar el joystick antes de usarlo, midiendo el valor de los pines cuando el joystick esté centrado.

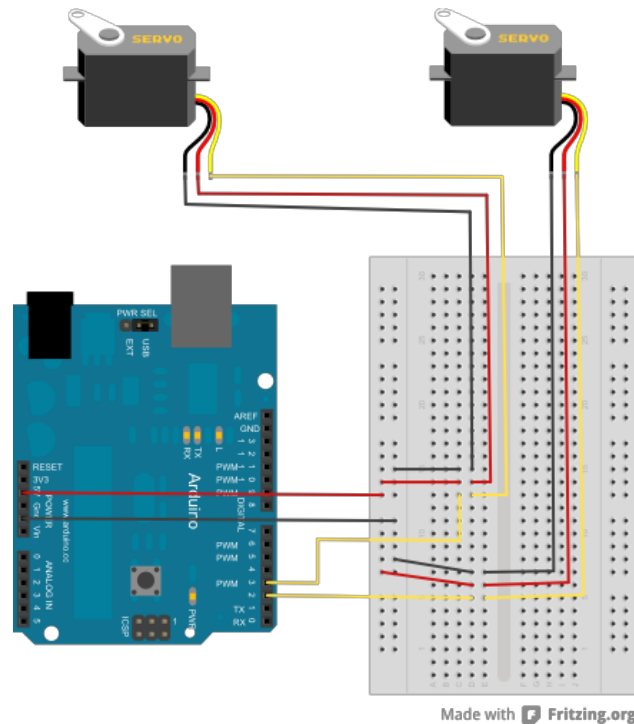


La construcción del sistema es relativamente sencilla siguiendo las instrucciones de montaje<sup>2</sup>:



<sup>2</sup> **Atención:** no excederse apretando los tornillos.

Respecto a la conexión de los servos al Arduino también es muy sencilla. En el siguiente diagrama se puede ver como se conectarían los servos al usar los pines 2 y 3<sup>3</sup>:



Gracias a la librería *Servo* de Arduino, el control de cada servomotor se simplifica bastante. El siguiente código muestra un ejemplo sencillo de inicialización de los servos:

```
#include <Servo.h>

//Se declaran los dos servos
Servo servoTilt, servoPan;

void setup(){
  servoTilt.attach(2); //Se indica que el servo de Tilt está en el pin 2.
  servoPan.attach(3); //Se indica que el servo de Pan está en el pin 3.
  servoTilt.write(90); //se ponen los dos servos a 90 grados
  servoPan.write(90);
}
```

---

<sup>3</sup> **Atención:** en los servomotores entregados el cable marrón va a GND, el rojo a +5V y el naranja al pin digital.

### **Enunciado de la práctica**

Haciendo uso del joystick y del sistema de pan&tilt, se pide implementar un sistema que permita controlar mediante el joystick la posición del sistema de pan&tilt. A mayores, el sistema deberá enviar por el puerto serie las coordenadas del joystick cada vez que se pulse el botón del joystick (acción que deberá controlarse mediante una interrupción).

Con todo el trabajo desarrollado, crear una memoria explicativa y enviarla a [tiago.fernandez@udc.es](mailto:tiago.fernandez@udc.es) acompañada del código fuente comentado.