



Guía de Prácticas

ASIGNATURA:	Control y Supervisión de Sistemas Dinámicos		
CENTRO:	Centro Internacional de Postgrado		
ESTUDIOS:	Master en Ingeniería Mecatrónica		
CURSO:	1º	CUATRIMESTRE:	2
CARÁCTER:	Obligatoria	CRÉDITOS ECTS:	6
PROFESORADO:	Borja Millán, Ignacio Alvarez		

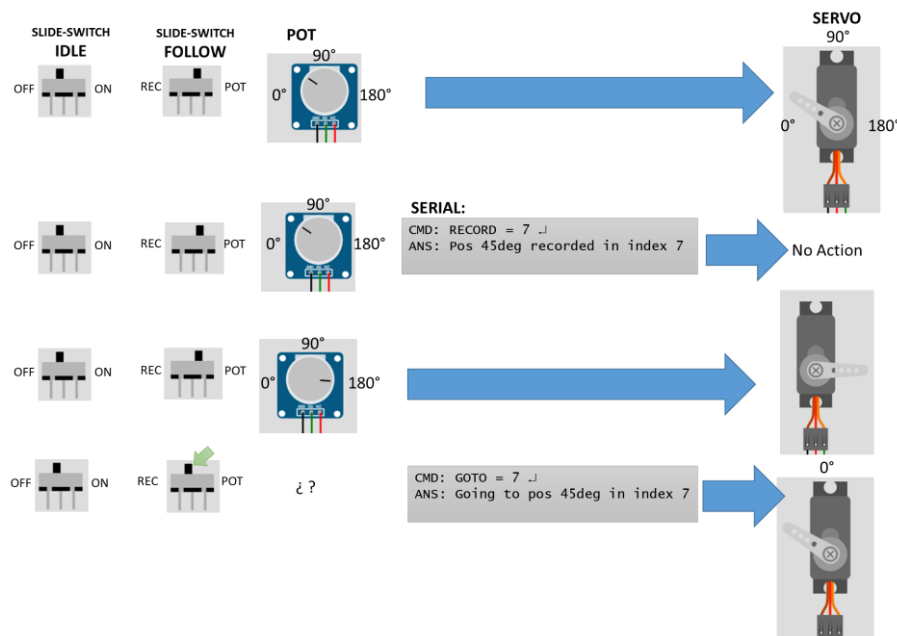
PRACTICA 01: Uso de clases estándar para programar un servomotor con ESP32

Utilizando el simulador Wokwi (<https://wokwi.com>), realizar un programa que permita controlar un servomotor con un ESP32, según los criterios siguientes:

- Se utilizará un potenciómetro como indicador de la posición deseada para el servomotor (ángulo entre 0° y 180°).
- El sistema puede estar en 3 estados: **IDLE** (no hace nada), **FOLLOW_POT** (el servomotor sigue los movimientos del potenciómetro), **FOLLOW_REC** (el servomotor va a posiciones previamente guardadas). Se utilizará un switch para indicar **IDLE/no IDLE**, y otro para seleccionar el modo **FOLLOW** cuando el estado no es **IDLE**.
- Mediante una comunicación serie RS-232, se enviarán dos posibles comandos para el servomotor:
 - RECORD=índice 0 a 19** (sólo aceptable en modo **FOLLOW_POT**)
 - GOTO=índice 0 a 19** (sólo aceptable en modo **FOLLOW_REC**)

El objetivo es que, mediante el potenciómetro, se establezcan posiciones (ángulo) deseadas para el servomotor, y éstas se guarden en una tabla mediante el comando **RECORD**. El comando **GOTO** sirve para ir automáticamente a las posiciones grabadas anteriormente.

Ejemplo:



Se usarán las clases:

- `Serial` para comunicación RS-232
- `ESP32Servo` para el servomotor
- `std::vector<float>` para la tabla de ángulos guardados
- `std::string` para cadenas de caracteres

y las funciones:

- `pinMode()`, `digitalRead()` para obtener el estado del switch
- `analogRead()` para obtener el valor del potenciómetro

AMPLIACIONES PROPUESTAS:

- Cambiar los slide-switch por un solo pushbutton que sirve para pasar de un estado a otro, y tres leds para indicar el estado actual (rojo: `IDLE`, amarillo: `FOLLOW_POT`, verde: `FOLLOW_REC`). Usar clase Pushbutton de librería PushButton para detectar transición evitando el rebote del pulsador.
- Cambiar tres leds de indicación de estado por un display LCD (modo I2C) en el que se visualiza el estado. Usar clase LiquidCrystal_I2C de librería LiquidCrystal I2C (ver <https://docs.wokwi.com/parts/wokwi-lcd1602>).
- Añadir un buzzer para emitir un aviso sonoro durante 3 segundos cuando se introduce un comando en un modo no válido.
- Añadir comando que permita ir a múltiples destinos de forma temporizada (sólo aceptable en modo `FOLLOW_REC`):

`MULTI = índice1 , tiempo1_ms ; índice 2, tiempo2_ms ; ...`

(ir a posición índice 1, esperar tiempo1, ir a posición índice 2, esperar tiempo 2, ...)