



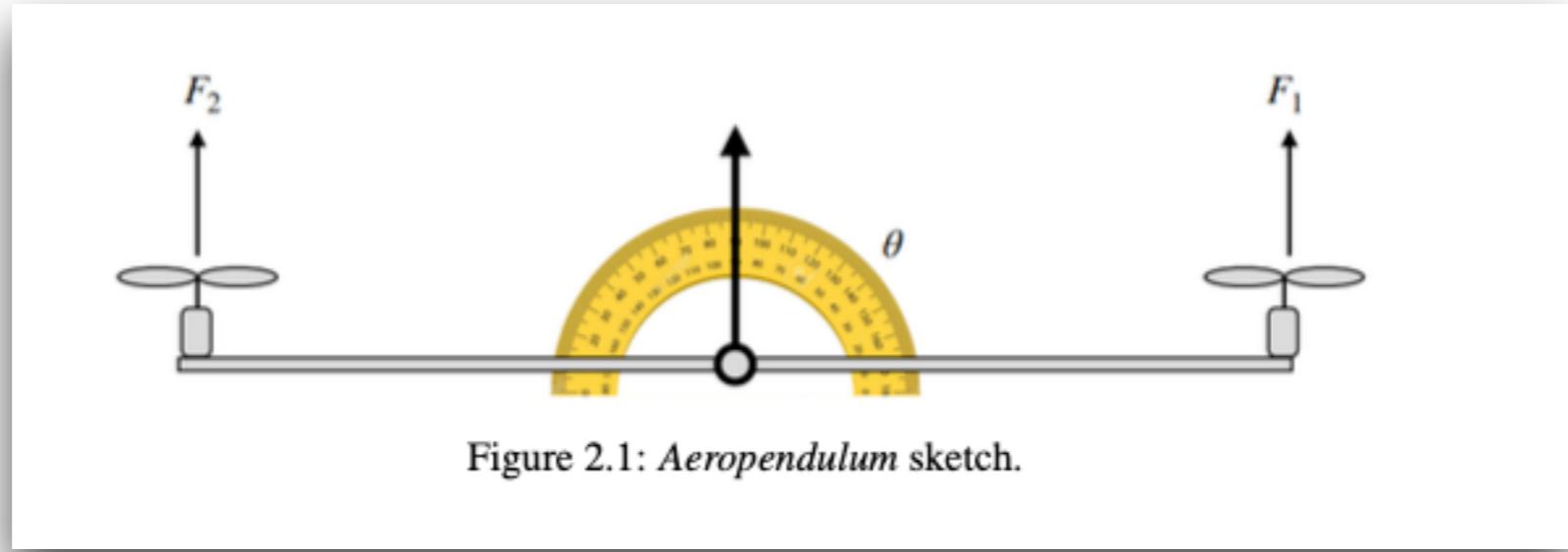
Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Documentación aeropéndulo

P1 demo aeropéndulo

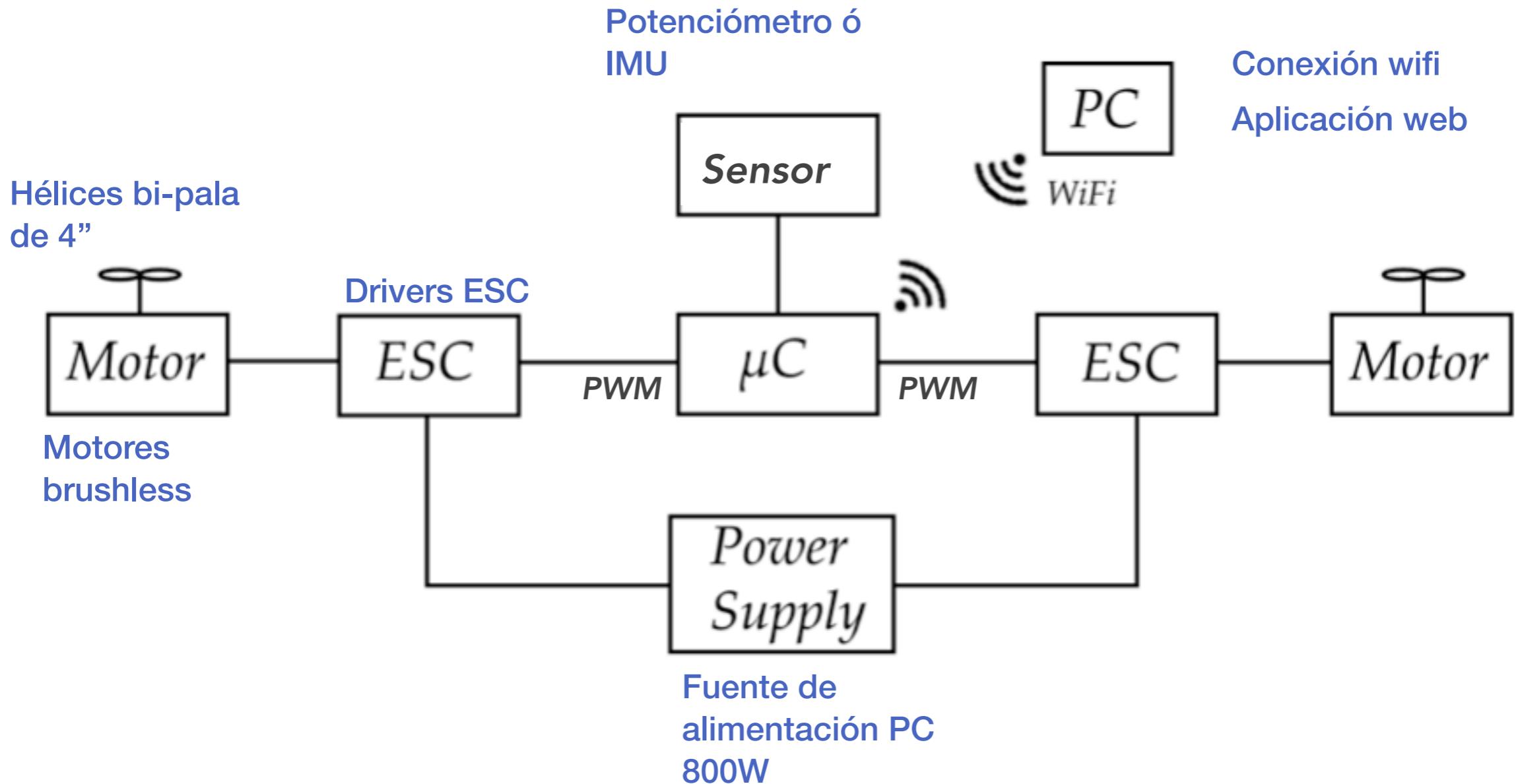
Sistemas de control

Sistema

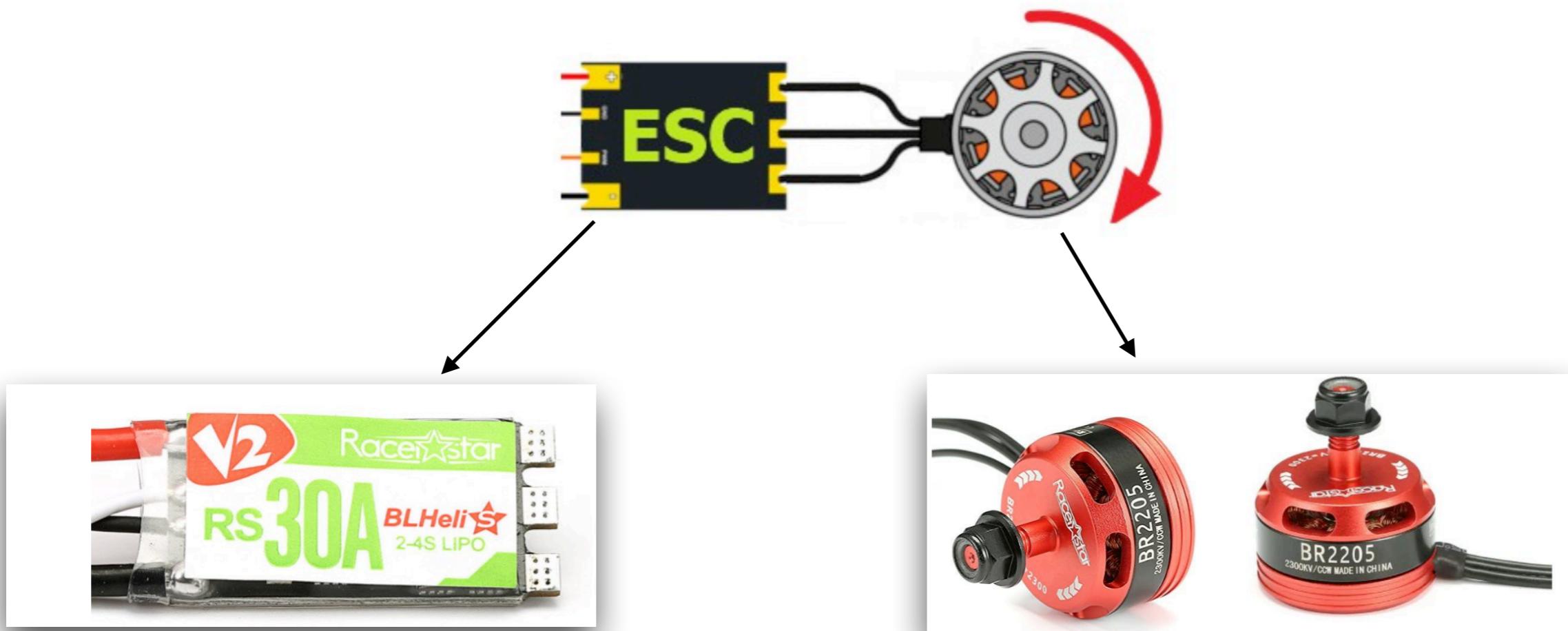


El objetivo es diseñar e implementar sistemas de control de posición y velocidad angular para el sistema aeropéndulo. Con la finalidad de ser utilizado para la enseñanza de las prácticas de laboratorio sobre control de sistemas dinámicos.

Esquema del sistema



Motores y drivers ESC



- Beheli_S 30A ESC
- Configuración mediante BLheliSuite
- Una vez configurados la **consigna de velocidad se introduce por PWM**
- 2-4S lipo battery
- https://www.racerstar.com/racerstar-rs30a-v2-30a-blheli-s-esc-opto-2-4s-support-oneshot42-multishot-16_5-dshot600-p-9.html

- Racerstar Racing Edition 2205 BR2205
- 2600KV Brushless Motor
- 2-4S lipo battery
- <https://www.racerstar.com/Racerstar-BR2205-2600KV-Brushless-Motor-p-15.html>

MODEL	KV (rpm/V)	Voltage (V)	Prop	Load Current (A)	Pull (g)	Power (W)	Efficiency (g/W)	Lipo Cell1	Weight (g) Approx
BR2205	2300	11.1	5045	19.2	660	213	3.1	2-4S	28
		14.8		27.6	950	408	2.3		
	2600	11.1	4045	18.5	530	205	2.6		
		14.8		23.2	710	343	2.1		

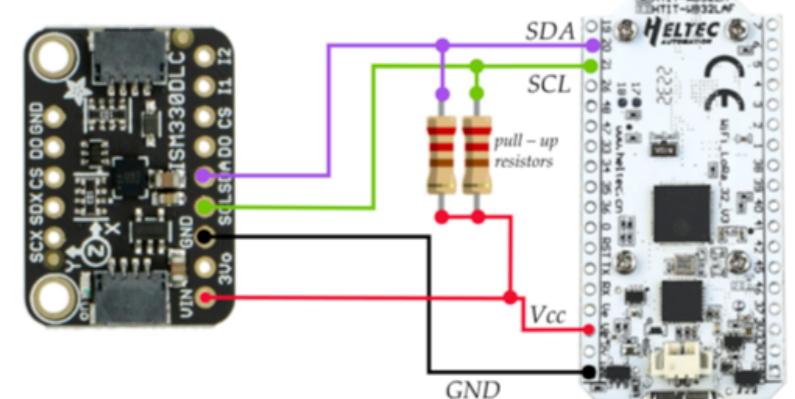
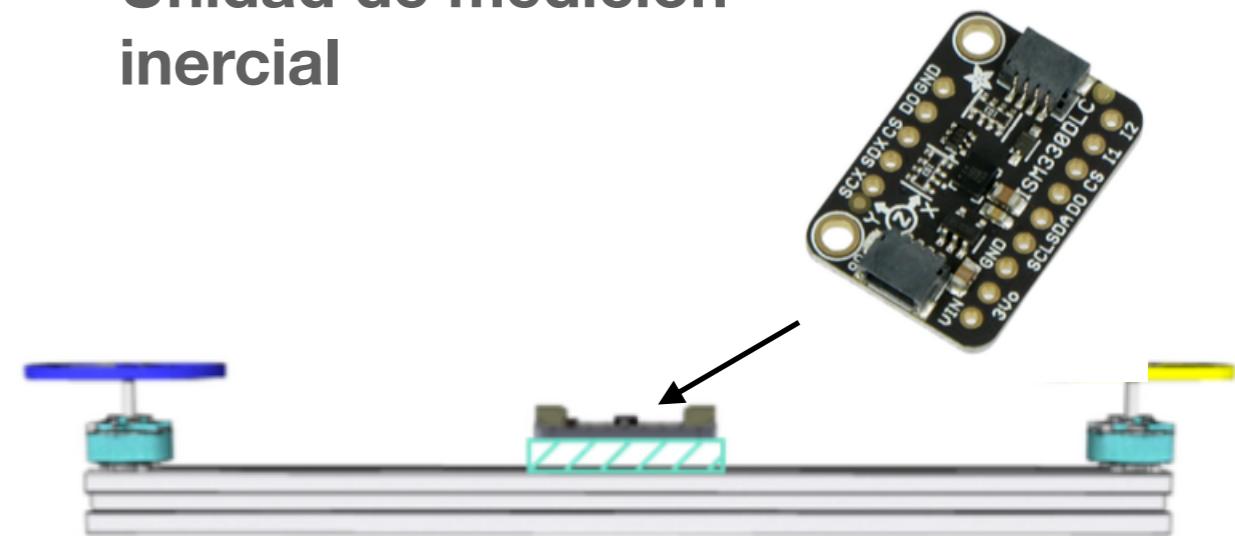
Sensores

Potenciómetro



- ➡ Potenciómetro lineal acoplado al eje.
- ➡ $R = 100 \text{ k}\Omega$
- ➡ Se alimenta directamente desde el micro a 3.3V.
- ➡ La caída de tensión de salida se lee en una entrada ADC.

Unidad de medición inercial



- ➡ Modelo SM330DHCX IMU
- ➡ 3 acelerómetros y 3 giróscopos. (6 DOF)
- ➡ Se alimenta directamente desde el micro a 3.3V.
- ➡ Las señales de los sensores se comunican al micro mediante I2C

Microcontrolador

ESP 32

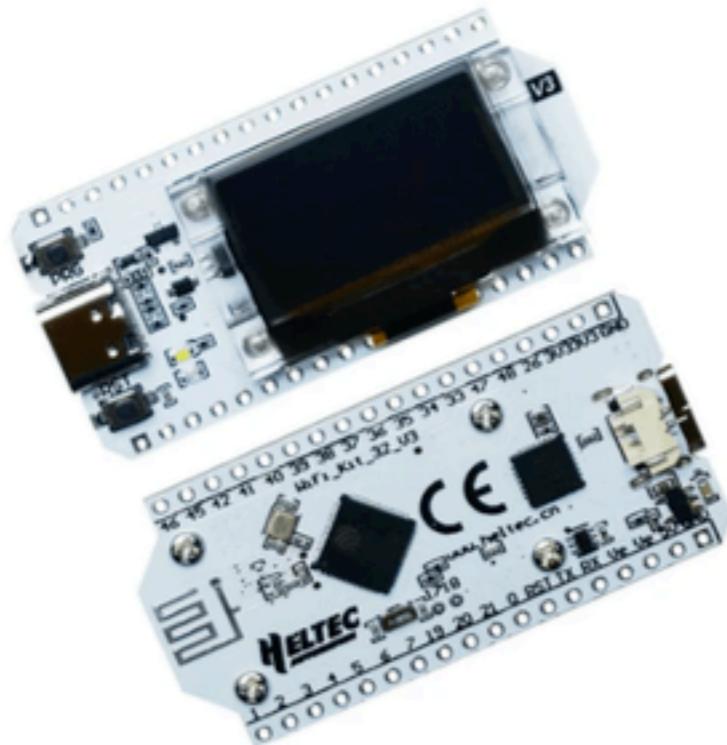


Figure 3.15: *Heltec WiFi kit 32 V2.*

Microcontroller	ESP32, Dual-Core 32-bit LX6
Frequency	240MHz
Flash	8MB
RAM	320KB
WiFi	802.11b/g/n/e/i
Bluetooth	v4.2 BR/EDR and BLE
USB to Serial Chip	CP2102
Interface	Micro USB
Vendor	<i>Heltec Automation</i>

Table 3.1: *WiFi kit 32 V2 characteristics.*

Sistema dinámico

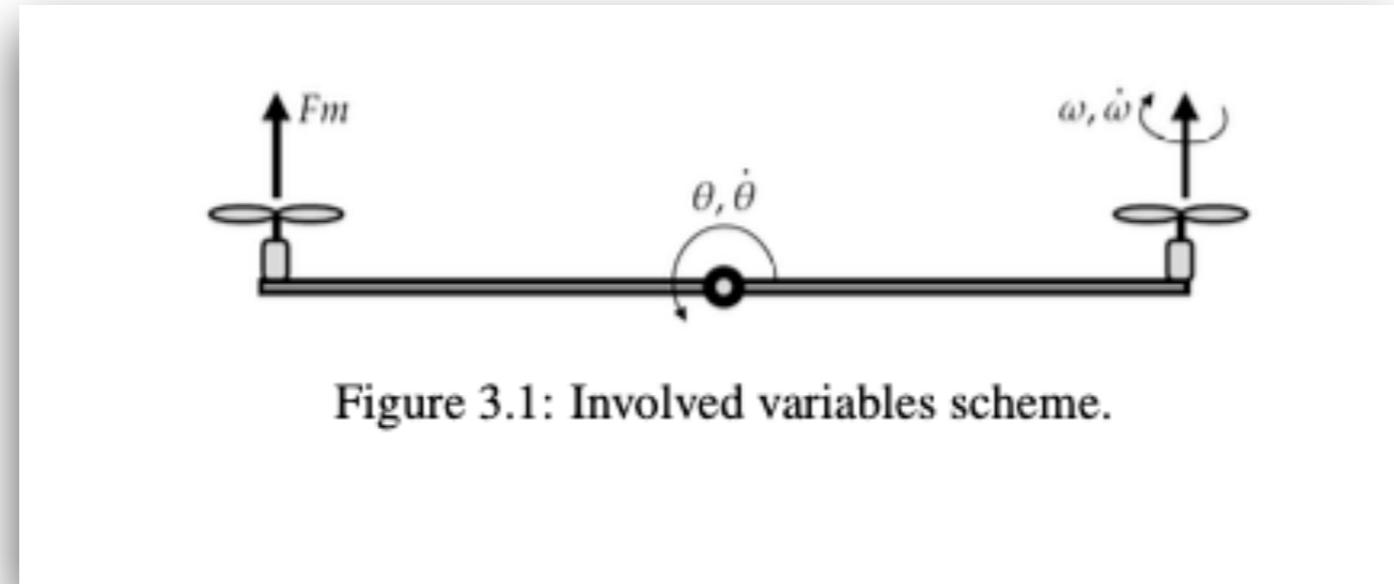


Figure 3.1: Involved variables scheme.

Dinámica del péndulo:

$$J_p \ddot{\theta} = -b_p \dot{\theta} + l_p F \implies \ddot{\theta} = -\frac{b_p}{J_p} \dot{\theta} + \frac{l_p}{J_p} F$$

$$F = F_1 - F_2$$

$$G_{pend} = \frac{K}{s(s+p_p)}$$

Dinámica de los motores:

$$\ddot{\omega} = -\frac{b_m}{T_m} \dot{\omega} + \frac{1}{T_m} \omega_{ref}$$

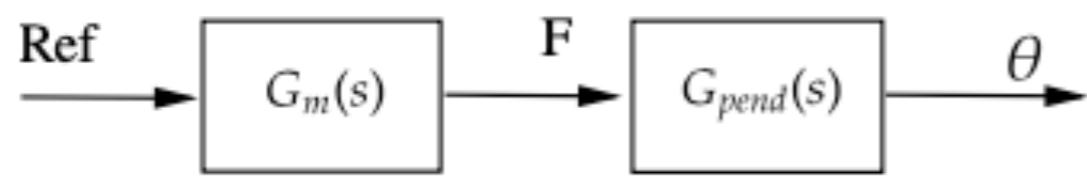
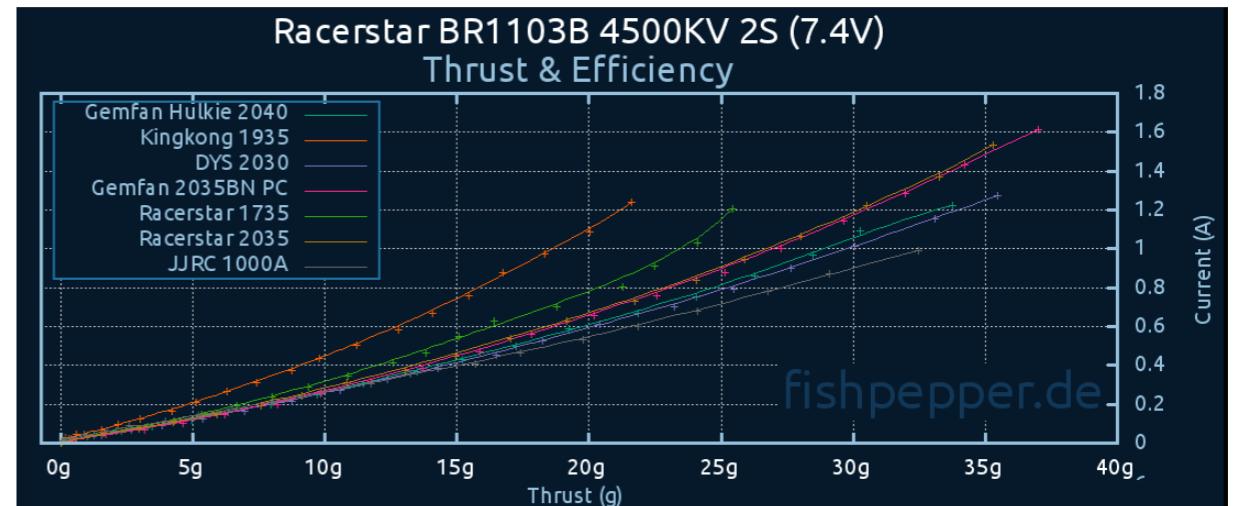
$$F_m = K_a \omega^2$$

Sistema dinámico

Simplificamos las ecuaciones de los motores eliminando la no linealidad cuadrática

$$F_m = K_a \omega^2 \rightarrow F_m \approx K_a \omega$$

$$G_m(s) = \frac{K_m}{s + p_m}$$



$$G(s) = \frac{Y(s)}{U(s)} = \frac{K}{s(s + p_m)(s + p_p)}$$

Figure 3.2: Open loop system block diagram.

Sistema dinámico

En la práctica el polo del motor es mucho más rápido que el del péndulo.
Lo podemos despreciar

$$G(s) = \frac{K}{s(s + p_p)}$$

Un sistema con un polo en el origen y un polo en semiplano negativo

Inestable en cada abierta

Identificación del sistema

$$G(s) = \frac{K}{s(s + p_p)}$$

Se obtienen los parámetros de $G(s)$ experimentalmente. Al ser inestable en cadena abierta se descarta los mecanismos habituales.
Se opta por:

- ➡ Identificar “a mano” el sistema realimentado con un regulador K_p , midiendo M_p y T_p de su respuesta ante escalón.
- ➡ Identificar el sistema realimentado mediante mínimos cuadrados en frecuencia o tiempo.
- ➡ Utilizar la Matlab system identification toolbox.

Identificación del sistema

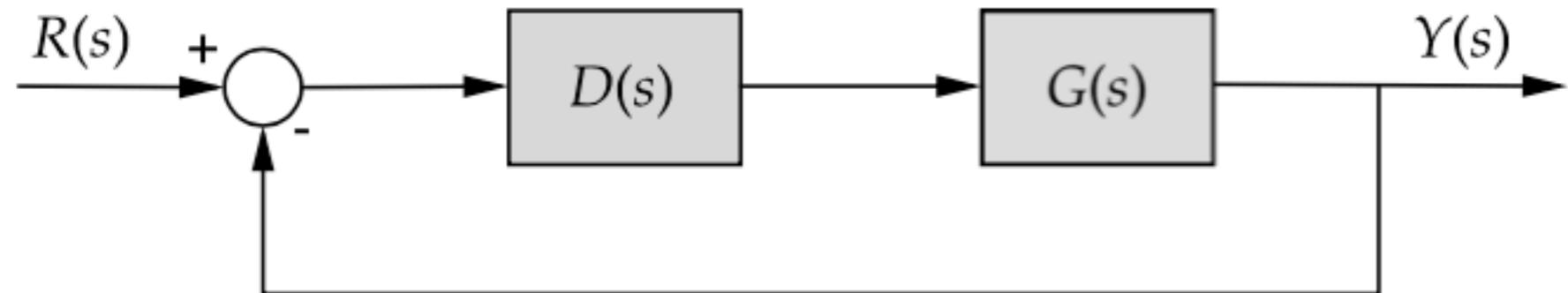


Figure 3.28: Proportional feedback control loop.

$$D(s) = K_p$$

$$G(s) = \frac{K}{s(s + p_p)}$$

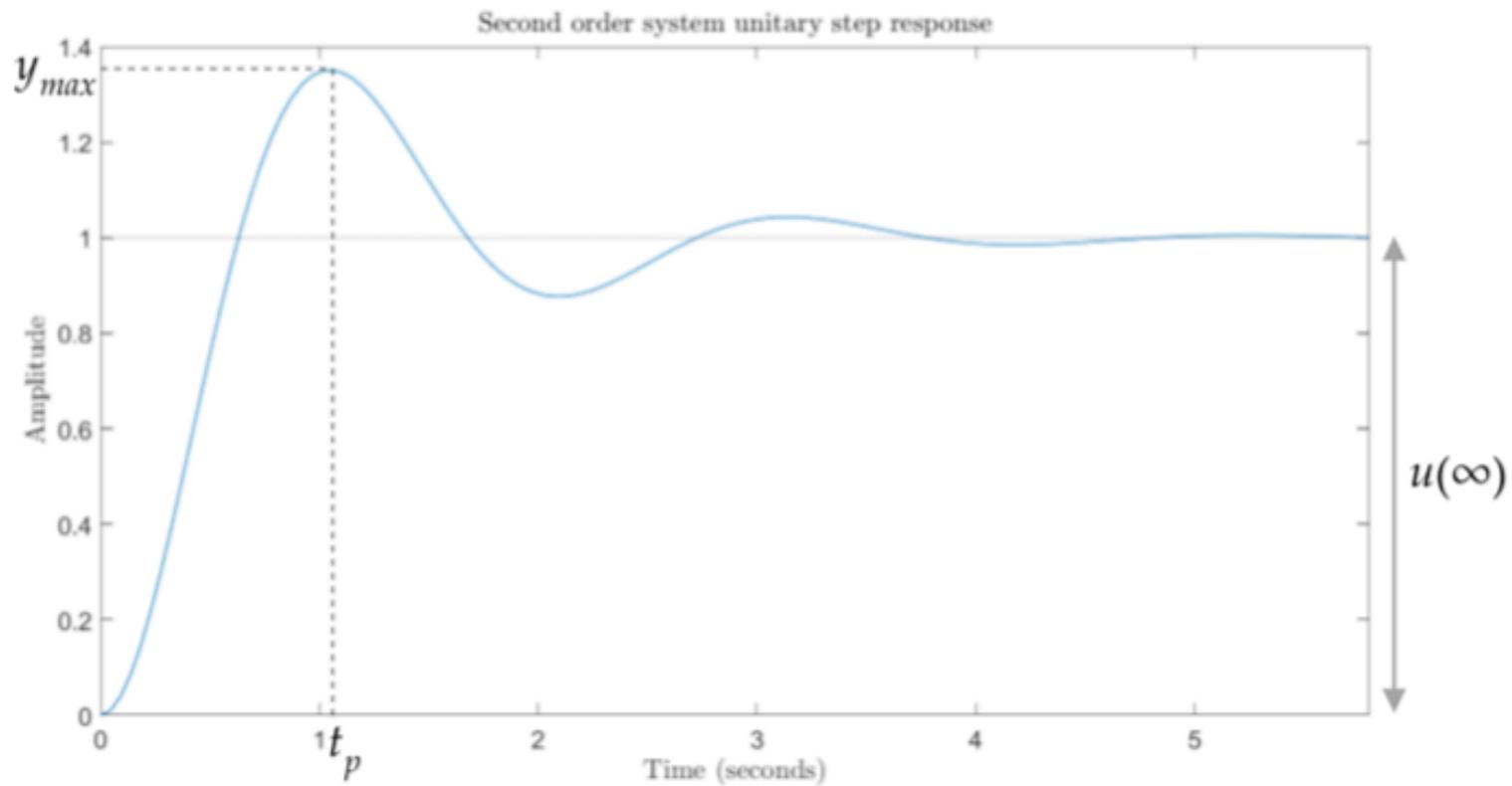
$$1 + D(s) G(s) = 0$$

↓

$$1 + \frac{KK_p}{s(s + p_p)} = s^2 + p_p s + KK_p = 0$$

Si se conoce K_p se puede despejar los parámetros K y p_p

Identificación del sistema



Obtener a partir de M_p y t_p

w_n ζ \longrightarrow

$$K = \frac{w_n^2}{K_p}$$

$$p_p = 2w_n\zeta$$

\longrightarrow

$$G(s) = \frac{0.853}{s(s + 0.607)}$$

Reguladores

El ESP32 ejecuta los siguientes reguladores

PID

$$PID(s) = \frac{U(s)}{E(s)} = K_p \left(1 + \frac{K_i}{s} + K_d s \right)$$

Red de adelanto atraso

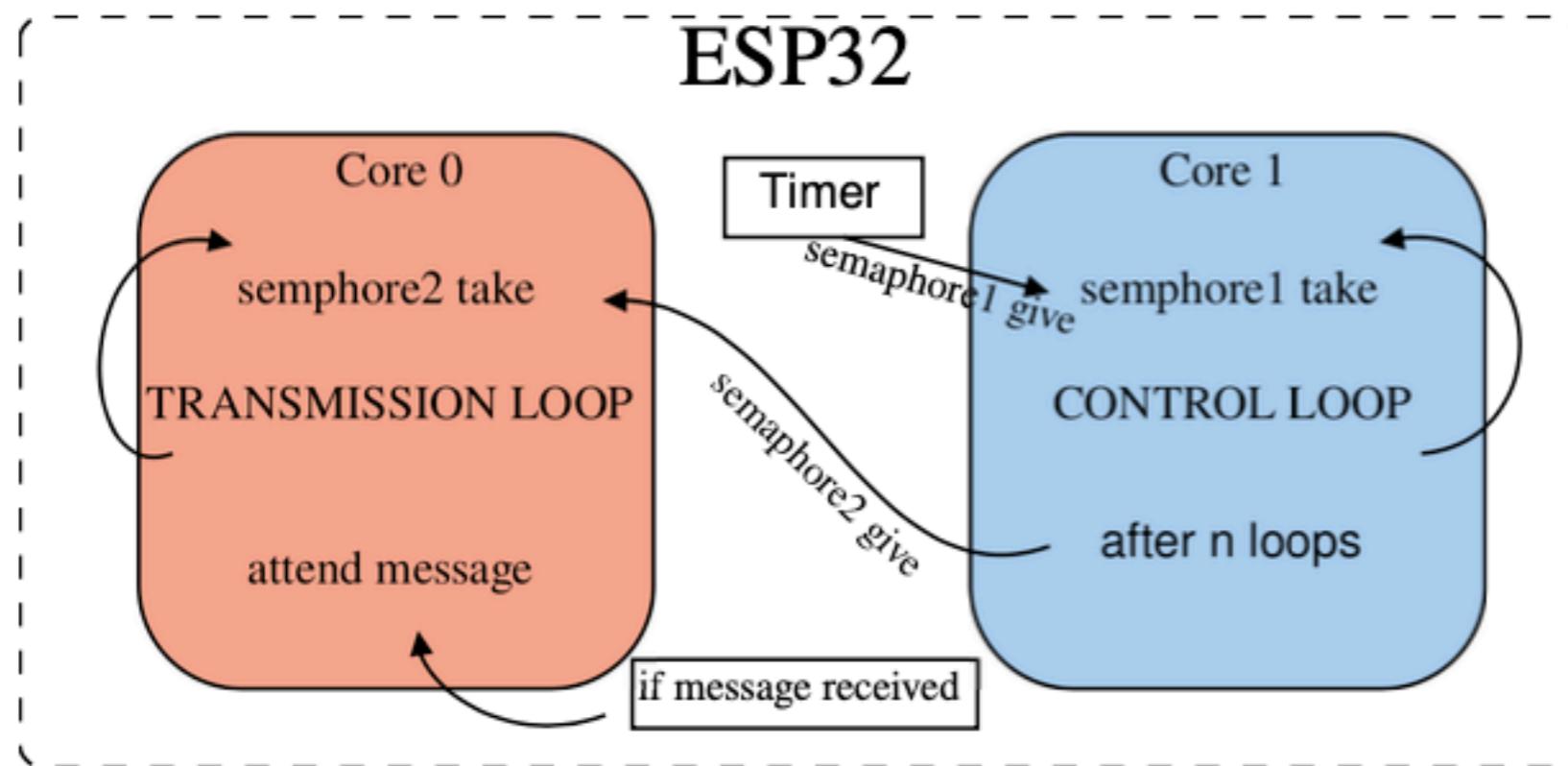
$$LL(s) = \frac{19.139(s + 2)(s + 0.607)}{(s + 8)(s + 0.08)}$$

Calculado con SISOTOOL

Ambos son controles digitales implementados discreteando los reguladores con un $ts = 1$ ms. Se usa tustin como método de discretización.

Programa de control

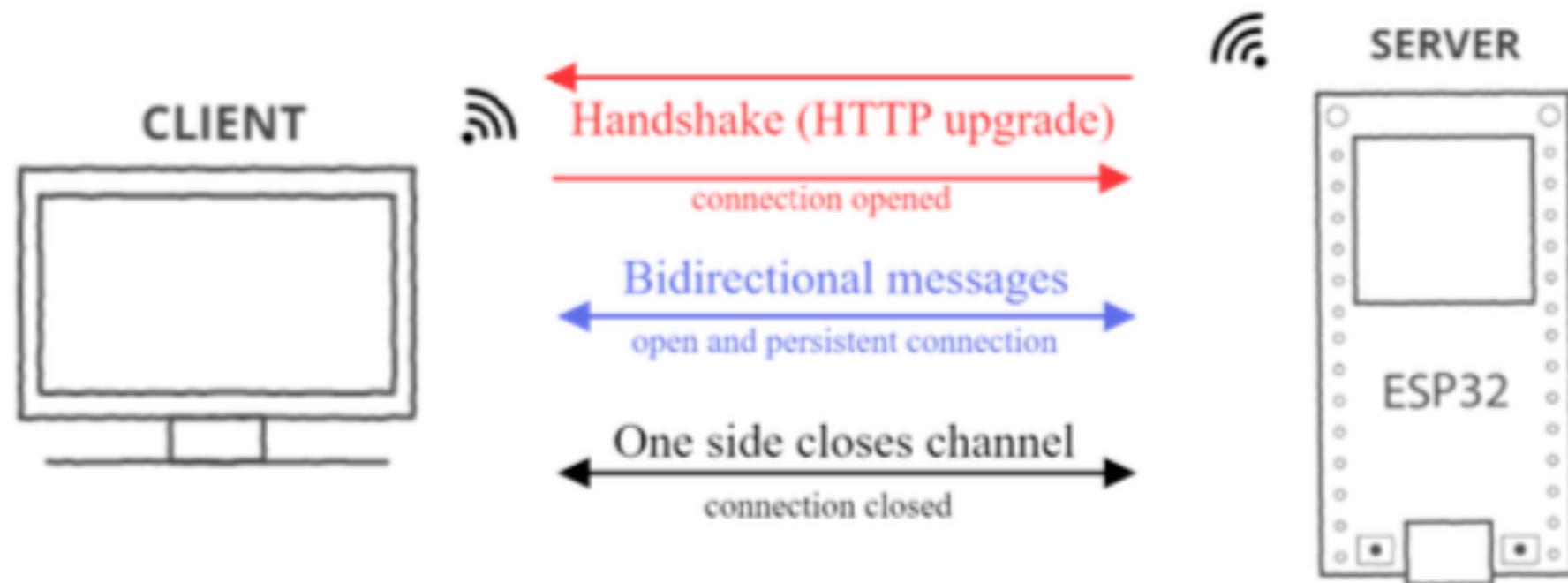
Estructura del programa de control



El periodo de muestro se garantiza con una interrupción de un timer.
 La tarea de control se sincroniza mediante un semáforo que se concede en la interrupción del timer. Esta es prioritaria y se ejecuta en el núcleo 0. El otro núcleo se dedica a las comunicaciones serie y websockets.

Servidor websocket

El ESP32 implementa un servidor websocket que escucha peticiones en un punto de acceso que el propio micro crea



La comunicación websocket permite la comunicación en tiempo real de las señales de control y el envío de consignas desde una aplicación web

Aplicación web

La aplicación web es un html plano que se puede ejecutar en cualquier navegador. El nombre del archivo es: *APPControlAppAeropendulo.html*



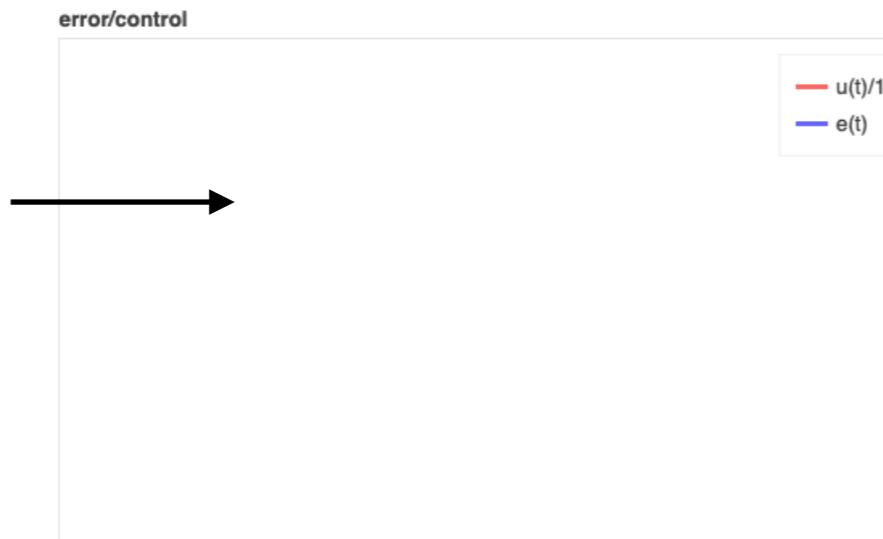
Control del aeropéndulo



Señales de consigna/referencia



Señales de error/acción de control



La planta a controlar presenta la siguiente función de transferencia:

$$G(s) = \frac{K}{s(s + p)}$$

Se implementan dos reguladores:

1) Control PID de la forma:

$$PID(s) = K_p \left(1 + \frac{K_i}{s} + K_d s \right)$$

2) Red de adelante/traso

on/off	
Consigna (°)	
<input type="text" value="0"/>	
<input type="range" value="50"/>	
<input checked="" type="radio"/> PID <input type="radio"/> Red adelante/traso	
Kp [1,7]	
<input type="text" value="7"/>	
Ki [0,0,7]	
<input type="text" value="0"/>	
Kd [0,0,008]	
<input type="text" value="0"/>	
Estado Ref	

Botón inicio del control

Selección del regulador

Parámetros del regulador

Procedimiento de arranque

Alimentar el ESP. Activar interruptor fuente del ESP



Esperar a que se encienda LED blanco en el ESP



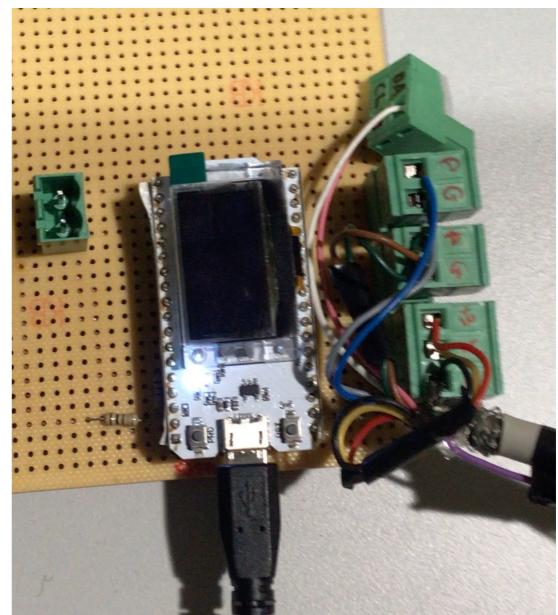
Antes de que se apague (tarda 10s) encender la fuente PC que alimenta los ESC y motores. El motor pitará y los motores empezarán a girar despacio. El control no está activo



Conectarse a la red wifi del ESP:

SSID: ESPAeropendulo

Password: SistemasControl2023



Abrir el archivo: **APPControlAppAeropendulo.html**



Activar el control en la web en el botón “on”



Taparse los oídos

Posibles problemas

Si los motores pitán muchas veces y muy seguido

- Los ESC han entrado en modo calibración.
- Apagar la fuente PC y repetir el procedimiento de arranque.
- Si no funciona hay reconfigurar los ESC. Seguir el pdf: *ConfiguracionESCWindows*

Si el ESP no responde o no se enciende el led blanco ...

- Utilizar el ESP de recambio con el programa.
- Sacar el ESP en fallo del socket
- Insertar el de repuesto y repetir arranque

Si los motores paran pero la app se mantiene activa ...

- La fuente se ha apagado por un pico de corriente. Posiblemente por corrientes inversas.
- Apagar la fuente PC, esperar unos segundos y repetir el procedimiento de arranque.

Si no funciona nada ...

- Llamarle: 985 18 (25 43).
- Explicar el sistema y poner el video del QR a los alumnos.

[https://youtube.com/watch?
v=9cIyRZ8ExLM&feature=shared](https://youtube.com/watch?v=9cIyRZ8ExLM&feature=shared)

