

Práctica de Laboratorio 8: Control Óptimo con Integrador

Elías Álvarez

Carrera de Ing. Electrónica

Universidad Católica Nuestra Señora de la Asunción

Asunción, Paraguay

Email: elias.alvarez@universidadcatolica.edu.py

Docente: Lic. Montserrat González

Facultad de Ingeniería

Universidad Católica Nuestra Señora de la Asunción
Asunción, Paraguay

Tania Romero

Carrera de Ing. Electrónica

Universidad Católica Nuestra Señora de la Asunción

Asunción, Paraguay

Email: tania.romero@universidadcatolica.edu.py

Docente: PhD. Enrique Vargas

Facultad de Ingeniería

Universidad Católica Nuestra Señora de la Asunción
Asunción, Paraguay

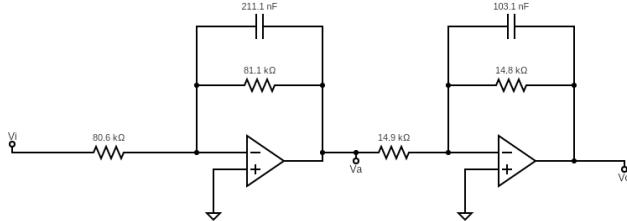


Figura 1. Circuito de la planta.

I. DESARROLLO

I-A. Modelado del Sistema

I-A1. Obtención de las matrices del sistema F, G, H y J: Para el modelado del sistema se parte del circuito mostrado en la Figura 1, a partir del cual se determinan las ecuaciones de estado mediante el análisis de los lazos de realimentación y las relaciones de tensión en los componentes.

El sistema se describe mediante las siguientes ecuaciones en espacio de estados:

$$\dot{x}(t) = F x(t) + G V_i(t) \quad (1)$$

$$y(t) = H x(t) + J V_i(t) \quad (2)$$

Para obtener las expresiones de las variables de estado, se parte del equivalente del paralelo entre un resistor y un capacitor:

$$R \parallel \frac{1}{sC} = \frac{R}{1 + sRC}$$

Considerando que ambos amplificadores operacionales se encuentran en configuración no inversora, se obtienen las siguientes relaciones:

$$V_a = \frac{-R_2}{1 + sR_2C_1} \frac{V_i}{R_1} \Rightarrow sV_a = -\frac{1}{R_1C_1} V_i - \frac{1}{R_2C_1} V_a$$

$$V_o = -\frac{R_4}{1 + sR_4C_2} \frac{V_a}{R_3} \Rightarrow sV_o = -\frac{1}{R_3C_2} V_a - \frac{1}{R_4C_2} V_o$$

Definiendo como variables de estado $x_1(t) = V_a$ y $x_2(t) = V_o$, las ecuaciones anteriores se expresan en forma matricial como:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} -\frac{1}{R_2C_1} & 0 \\ -\frac{1}{R_3C_2} & -\frac{1}{R_4C_2} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} -\frac{1}{R_1C_1} \\ 0 \end{bmatrix} V_i(t)$$

y la ecuación de salida queda definida como:

$$y(t) = [0 \ 1] \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + 0 \cdot V_i(t)$$

Sustituyendo los valores de los componentes $C_1 = 211.1 \times 10^{-9} \text{ F}$, $R_1 = 80.55 \times 10^3 \Omega$, $R_2 = 81.09 \times 10^3 \Omega$, $C_2 = 103.07 \times 10^{-9} \text{ F}$, $R_3 = 14.878 \times 10^3 \Omega$ y $R_4 = 14.76 \times 10^3 \Omega$, se obtienen las siguientes matrices numéricas:

$$F = \begin{bmatrix} -58.42 & 0 \\ -652.11 & -657.37 \end{bmatrix}, \quad G = \begin{bmatrix} -58.81 \\ 0 \end{bmatrix}, \quad H = [0 \ 1] \quad y \quad J = 0 \quad (3)$$

I-A2. Mostrar el diagrama de bloques del sistema:

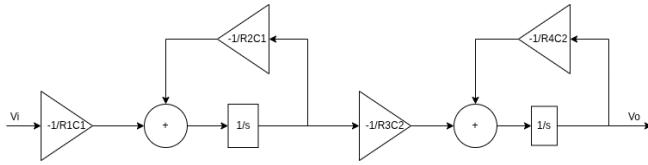


Figura 2. Diagrama de bloques del sistema continuo.

II. DISCRETIZACIÓN DEL SISTEMA

II-A. Elección del tiempo de muestreo $T_s = 1.1948 \text{ ms}$, usado en laboratorios anteriores

Para la discretización del sistema continuo descrito por las ecuaciones (1) y (2), se busca obtener un modelo equivalente en tiempo discreto que relacione las variables de estado y la señal de entrada en instantes de muestreo definidos. Las ecuaciones del sistema discreto se expresan como:

$$X(k+1) = A X(k) + B u(k) \quad (4)$$

$$Y(k) = C X(k) + D u(k) \quad (5)$$

Usando las matrices continuas F , G , H y J obtenidas previamente, las matrices discretas se determinan mediante las siguientes expresiones:

$$A = e^{FT_s}, \quad B = F^{-1}(e^{FT_s} - I) G, \quad C = H$$

$$\& \quad D = J$$

II-B. Obtención de las matrices discretas A , B , C y D

Con un tiempo de muestreo $T_s = 1 \text{ ms}$, se obtienen las siguientes matrices discretizadas:

$$A = \begin{bmatrix} 0.9226 & 0 \\ -0.5189 & 0.456 \end{bmatrix}, \quad B = \begin{bmatrix} -0.0678 \\ 0.02089 \end{bmatrix},$$

$$C = [0 \ 1], \quad \& \quad D = 0 \quad (6)$$

Estas matrices representan el modelo digital equivalente del sistema continuo, y serán utilizadas posteriormente para el diseño del controlador e implementación en el PSOC.

II-C. Diagrama de bloques del sistema discretizado

En la Figura 3 se presenta el diagrama de bloques correspondiente al sistema discretizado, donde se observa la relación entre las variables de estado, la entrada $u(k)$ y la salida $Y(k)$.

II-D. Comparar los resultados obtenidos con las simulaciones realizadas en MATLAB

Resultados de las matrices discretas:

$$A = \begin{bmatrix} 0.9433 & 0 \\ -0.4628 & 0.5182 \end{bmatrix}, \quad B = \begin{bmatrix} -0.05712 \\ 0.01527 \end{bmatrix}, \quad C = [0 \ 1]$$

$$\& \quad D = [0]$$

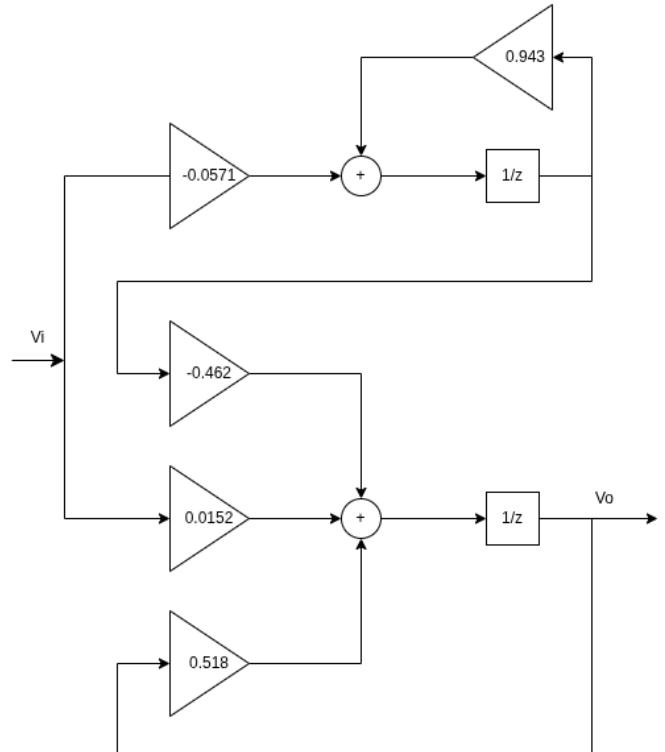


Figura 3. Diagrama de bloques del sistema discretizado.

III. MODIFICACIÓN DE LA DINÁMICA DE LA PLANTA

III-A. Modificar la dinámica de la planta con integrador considerando 3 matrices de pesos distintas

Para asegurar error nulo en régimen permanente ante referencias tipo escalón, se emplea un regulador LQI (LQR con integrador del error). Esto se logra aumentando el modelo con el estado integrador ξ que acumula $e = r - y$, y diseñando la realimentación óptima sobre el par aumentado. En discreto, con período T_s , el modelo queda:

$$A_{ad} = \begin{bmatrix} A_d & 0 \\ -T_s C & I \end{bmatrix}, \quad B_{ad} = \begin{bmatrix} B_d \\ 0 \end{bmatrix},$$

$$K_a = [K \ K_i] = \text{dlqr}(A_{ad}, B_{ad}, Q_a, R),$$

y la ley de control es $u_k = -K \hat{x}_k - K_i \xi_k$ (no se requiere pre-filtro para escalones). Para mantener la notación del enunciado, tomamos $Q_1 \equiv Q_a$ y $Q_2 \equiv R$.

III-B. Modificar la dinámica de la planta con integrador considerando las siguientes matrices de pesos:

$$Q_1 = \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,001 \end{bmatrix}, \quad Q_2 = 1$$

III-C. Modificar la dinámica de la planta con integrador considerando:

$$Q_1 = \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,1 \end{bmatrix}, \quad Q_2 = 0,1$$

III-D. Modificar la dinámica de la planta con integrador considerando:

$$Q_1 = \begin{bmatrix} 0,01 & 0 & 0 \\ 0 & 0,01 & 0 \\ 0 & 0 & 0,001 \end{bmatrix}, \quad Q_2 = 0,01$$

III-E. Barrido de ponderaciones y síntesis LQR (discreto) con integrador

Con el fin de comparar el compromiso rapidez/esfuerzo/robustez, se sintetizaron tres controladores LQR sobre el modelo aumentado empleado en los laboratorios anteriores. Para automatizar el proceso, se organizaron los pesos en arreglos de celdas y se computaron en lote las ganancias, los polos y las matrices de lazo cerrado asociadas a cada caso.

III-E0a. Conjuntos de pesos.: Se definieron tres configuraciones de ponderaciones (Q_1, Q_2), donde Q_1 penaliza los estados del modelo aumentado y Q_2 penaliza el esfuerzo de control:

$$Q_1^{(1)} = \text{diag}(0,01, 0,01, 0,001), \quad Q_2^{(1)} = 1,$$

$$Q_1^{(2)} = \text{diag}(0,01, 0,01, 0,1), \quad Q_2^{(2)} = 0,1,$$

$$Q_1^{(3)} = \text{diag}(0,01, 0,01, 0,001), \quad Q_2^{(3)} = 0,01.$$

III-E0b. Estructura de datos y cálculo en lote.: Para facilitar el barrido se utilizaron celdas de MATLAB que almacenan, para cada caso i , la ganancia $K^{(i)}$, su partición sobre estados e integrador ($K_2^{(i)}, K_1^{(i)}$), la solución de Riccati $P^{(i)}$ y las matrices de lazo cerrado.

III-E0c. Observador (predictivo y actual).: Para la estimación de estado se empleó, por sencillez, un mismo par de observadores en los tres casos: uno predictivo y otro actual. Los polos del observador se seleccionaron como un par complejo conjugado amortiguado en el interior del círculo unidad.

III-E0d. Simulación y graficación.: El esquema de simulación reutiliza la infraestructura de prácticas anteriores: para cada configuración ($Q_1^{(i)}, Q_2^{(i)}$) se cierra el lazo con $K^{(i)}$, se integra la dinámica y se registran respuesta temporal, esfuerzo de control y ubicación de polos.

En dichas imágenes (4, 5 y 6), los triángulos rosados indican los polos del sistema en lazo cerrado y los cuadrados azules indican los polos del estimador.

IV. IMPLEMENTACIÓN DEL SISTEMA

IV-A. Implementar el estimador en el PSoC

Esta sección describe el montaje hardware-software utilizado para ejecutar el estimador y el controlador en tiempo real sobre la plataforma PSoC, así como la lógica de temporización y los recursos auxiliares empleados.

IV-A0a. Planta y periféricos auxiliares.: La planta bajo control se muestra en la Fig. 7. Los circuitos auxiliares empleados para referencia, adquisición y conmutación de modos se ilustran en la Fig. 8.

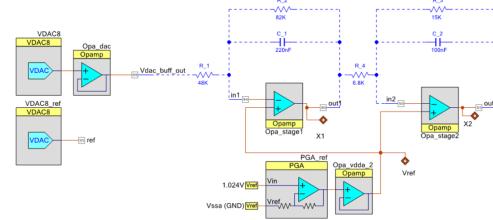


Figura 7. Planta bajo control.

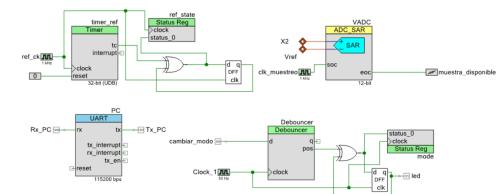


Figura 8. Periféricos y lógica auxiliar implementados en el PSoC.

IV-A0b. Generación de referencia (timer_ref).: Un temporizador `timer_ref` genera una referencia tipo escalón. Para permitir que el procesador lea su estado, se utiliza un registro `ref_state`. La combinación de una compuerta XOR y un biestable tipo D implementa un *flip-flop* tipo T: a cada evento TC (terminal count) del temporizador, el T-FF comuta y, por ende, alterna el nivel lógico de la referencia (útil para secuencias de ensayo).

IV-A0c. Interfaz con PC (PC por UART).: Se dispone de un puerto UART (PC) para comunicación por USB. Habitualmente permite ajustar amplitud y período de la referencia, así como abrir/cerrar el lazo para pruebas. En este laboratorio no se utilizó funcionalmente, pero permanece disponible.

IV-A0d. Adquisición (ADC SAR).: La señal de salida de la planta se digitaliza con el ADC SAR (VADC) en **modo diferencial** para mejorar el rechazo del offset de continua. Un pulso `soc` (start of conversion) inicia cada conversión a 1 kHz. Al finalizar, el ADC activa `eoc` (end of conversion), que dispara la ISR de control.

IV-A0e. Conmutación de estimador (botón + Debouncer).: El botón del kit conmuta el modo de estimación entre *predictivo* y *actual*. Para evitar rebotes se emplea Debouncer y, de forma análoga a la referencia, un

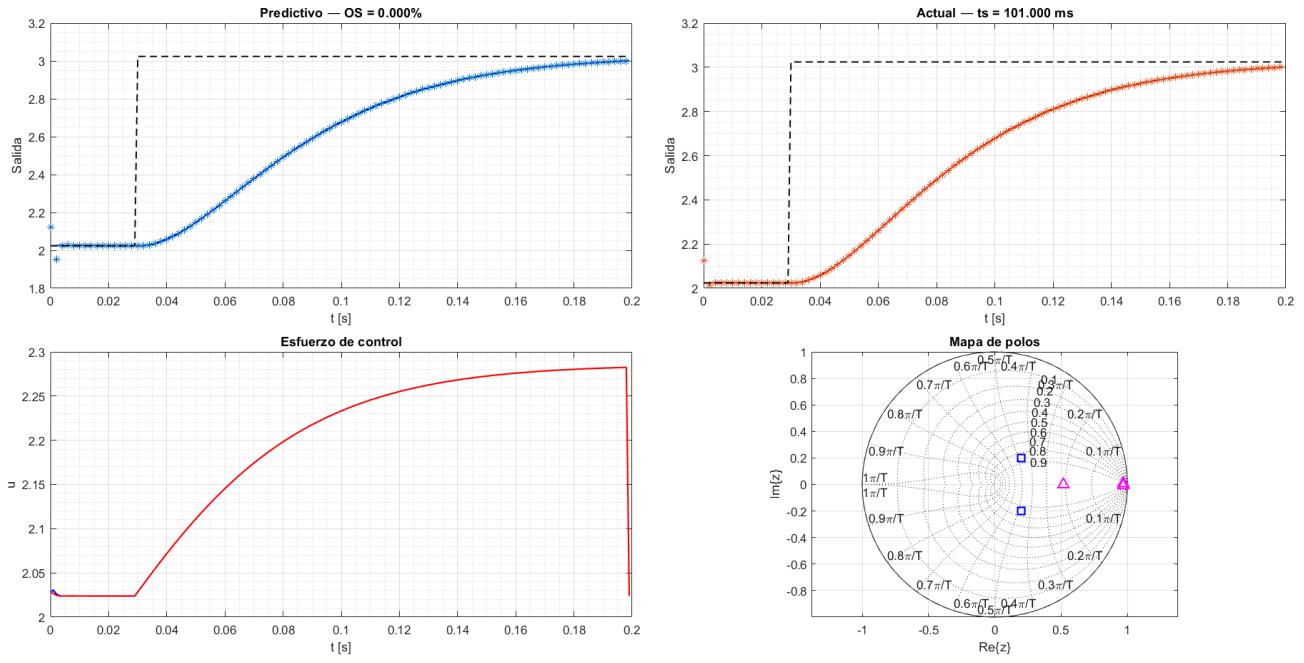


Figura 4. Simulación en MATLAB: predictivo vs. actual; esfuerzo de control y círculo unitario (C1).

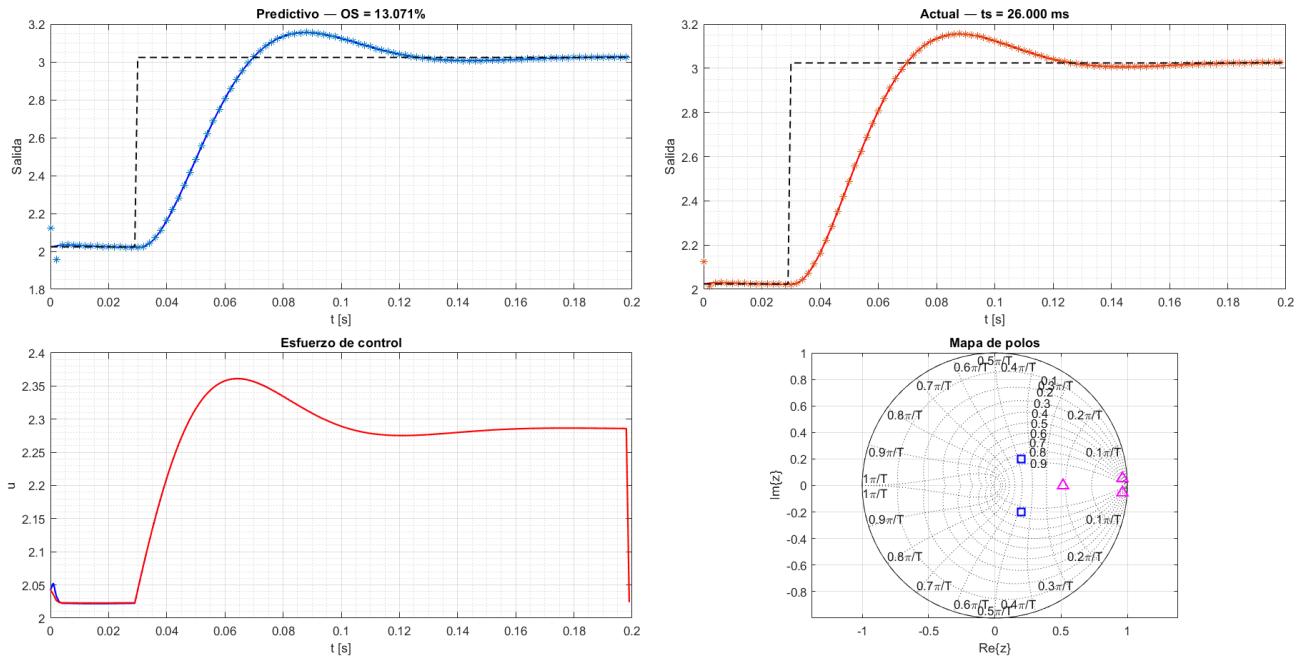


Figura 5. Simulación en MATLAB: predictivo vs. actual; esfuerzo de control y círculo unitario (C2).

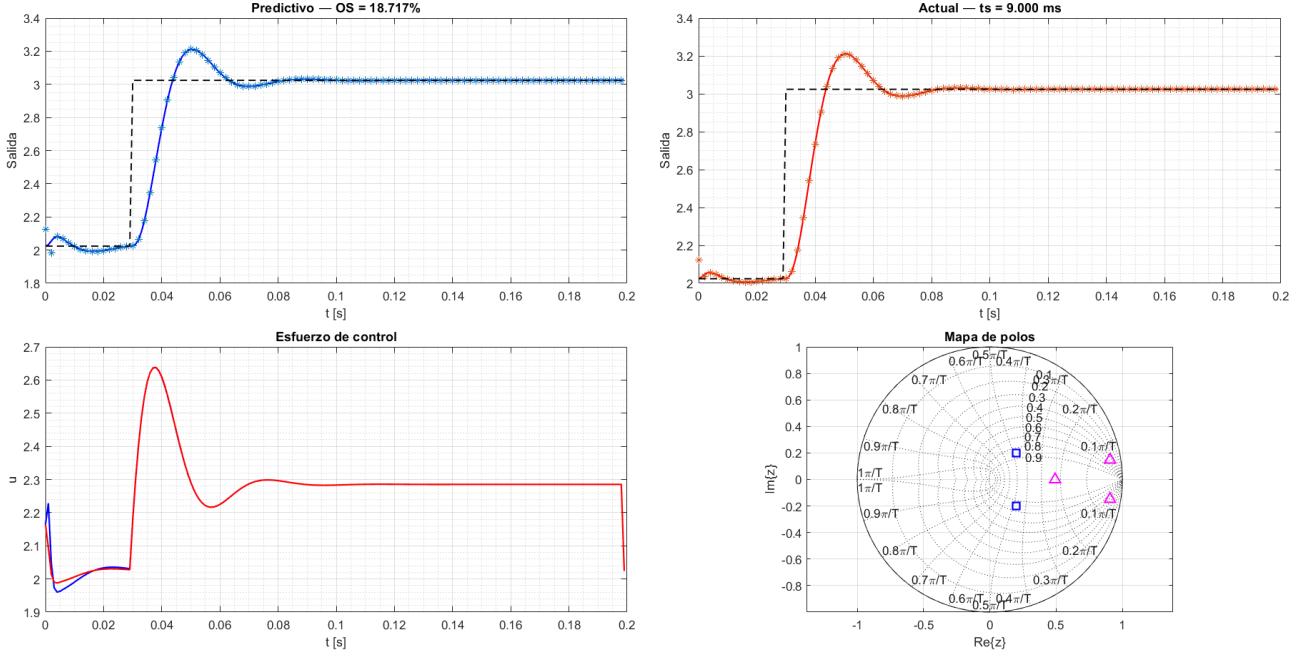


Figura 6. Simulación en MATLAB: predictivo vs. actual; esfuerzo de control y círculo unitario (C3).

biestable mode que opera como T-FF. El firmware lee modo en cada ciclo y selecciona el estimador correspondiente.

IV-A0f. Ciclo de control (ISR a 1 kHz).: Cada interrupción eoc ejecuta la siguiente secuencia:

1. Lectura de y_k desde VADC.
2. Actualización del estimador (*predictivo* o *actual*, según modo) para obtener \hat{x}_k .
3. Actualización del integrador del error ξ_k (servo con LQI).
4. Cálculo del control $u_k = -K \hat{x}_k - K_i \xi_k$ y aplicación de saturaciones físicas.
5. Escritura de u_k al DAC (esfuerzo aplicado a la planta).
6. Registro de variables para trazas/osciloscopio si corresponde.

IV-A0g. Trazas experimentales.: En las figuras (9, 10 y 11), la **curva superior** (sonda 1) corresponde a la *salida de la planta* y la **curva inferior** (sonda 2) al *esfuerzo de control*. Se muestran fotografías del estimador *actual*; las diferencias visuales con el *predictivo* resultaron poco apreciables en estas condiciones de prueba.

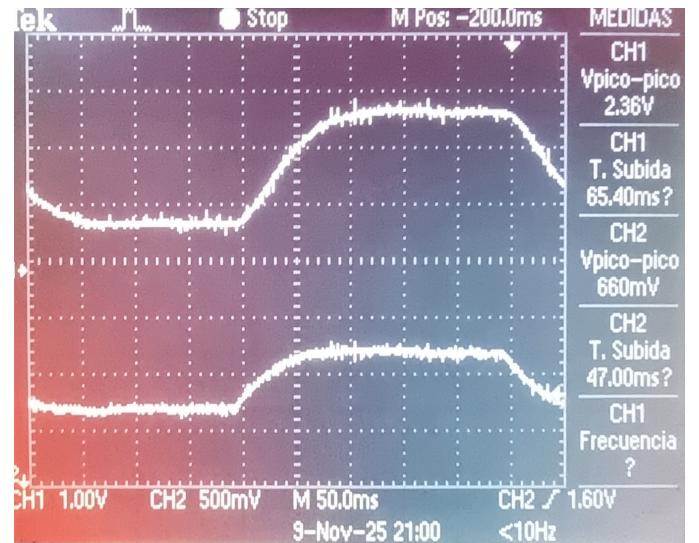


Figura 9. Experimento — Estimador actual (Caso C1).

V. RESULTADOS

V-A. Comparar los resultados obtenidos con Matlab.

Los resultados del laboratorio confirman que el control óptimo con integrador (LQI) cumple el objetivo de **seguimiento con error estacionario nulo**, manteniendo una dinámica global coherente con la simulación. Tanto el *observador predictivo* como el *actual* reprodujeron la tendencia teórica: al pasar de C1, C2 a C3 y tornar el diseño más agresivo (mayor peso al integrador y/o menor R), el **tiempo de subida** disminuye y el **sobreimpulso** aumenta. En la Tabla I se observa, por ejemplo, que t_r en PSoC es sistemáticamente menor que en

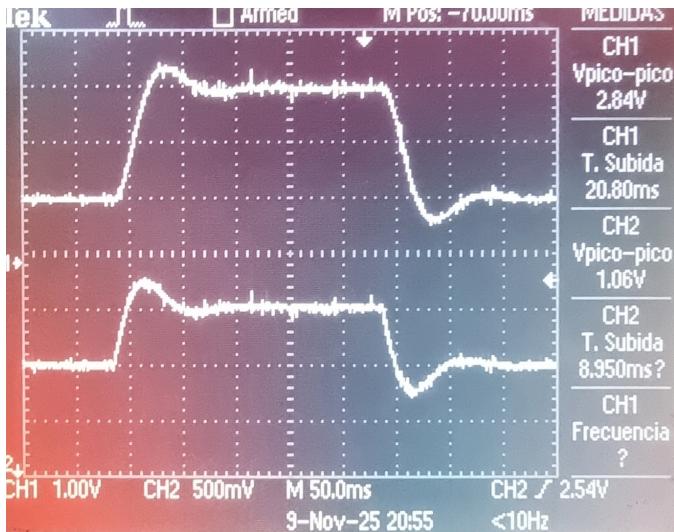


Figura 10. Experimento — Estimador actual (Caso C2).

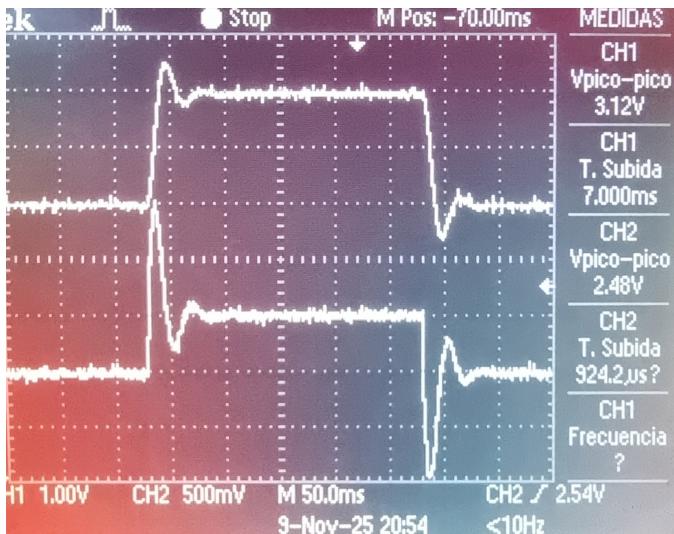


Figura 11. Experimento — Estimador actual (Caso C3).

Cuadro I

COMPARACIÓN DE OVERSHOOT (OS) Y TIEMPO DE SUBIDA t_r ENTRE EXPERIMENTO (PSOC) Y SIMULACIÓN.

Caso	OS [%]			t_r [ms]		
	PSOC	Sim.	$ \Delta $	PSOC	Sim.	$ \Delta $
C1	0	0	0	65.4	101	35.6
C2	20	13.071	6.929	20.8	26	5.2
C3	25	18.717	6.28	7	9	2

simulación (C1: 65.4 ms vs. 101 ms; C3: 7 ms vs. 9 ms), mientras que el OS medido resulta mayor en los casos más exigentes (C2: 20 % vs. 13.071 %; C3: 25 % vs. 18.717 %). Estas discrepancias se explican por *tolerancias de componentes, desajustes de ganancia, cuantización y latencias DAC/ADC, detalles de discretización* (T_s efectivo y retardo de ZOH), *saturación* (y la gestión de *anti-windup*) y ligeros *desvíos de*

polos respecto a los ubicados por diseño. Aun así, el LQI mostró **estabilidad robusta** y esfuerzo de control acotado en los tres casos; en particular, C2 ofrece un compromiso práctico entre rapidez y OS para esta planta.