CURSO COMPLETO: MODELADO DE SISTEMAS EN MECATRÓNICA

Video 08 — Parte 4B: Sistemas en Serie, Lazos de Realimentación y Ubicación de Polos — Álgebra de Diagramas de Bloques, Cascada vs. Efecto de Carga, Fórmula de Lazo Cerrado T(s) y Polos en el Plano Complejo

Parte del Curso: Parte 4B — Sistemas en Serie, Lazos de Realimentación y Ubicación de Polos
Planta / Sistema Modelado: Lazo Cerrado de Control T(s) = G(s)/(1 + G(s)H(s)) y Geometría de Polos en el Plano Complejo s

$$T(s) = G(s) / (1 + G(s)\cdot H(s)) \mid 1 + G(s)\cdot H(s) = 0 \mid s_{1,2} = -\sigma \pm j\cdot\omega_d = -\zeta\cdot\omega_n \pm j\cdot\omega_n\cdot\sqrt{(1-\zeta^2)}$$

1. Síntesis de Conceptos Clave y Fórmulas de Ingeniería

- Álgebra de diagramas de bloques: conexión en serie o cascada G_eq(s) = G_1(s)·G_2(s) (evitando efecto de carga mediante buffers), conexión en paralelo G_1(s) ± G_2(s).
- Fórmula fundamental de realimentación negativa T(s) = G(s) / (1 + G(s)·H(s)), rechazo de perturbaciones D(s) e insensibilidad a variaciones de la planta.
- Interpretación geométrica del plano complejo s = σ + j·ω: líneas verticales (igual t_s = 4/σ), líneas horizontales (igual t_p = π/ω_d) y rayos radiales (igual ζ = cos θ e igual M_p).
- Criterio de estabilidad BIBO: todos los polos de lazo cerrado deben residir estrictamente en el Semiplano Izquierdo abierto (σ < 0); concepto de polos dominantes (regla 5x).

2. Parámetros Nominales del Modelo Matemático (Video 08)

Parámetro Físico / Canónico	Símbolo	Valor Nominal	Significado en el Modelado Dinámico
Masa / Momento de Inercia Equivalente	`m_eq / J_eq`	1	Almacenador de energía cinética opositor a cambios de velocidad
Amortiguamiento / Resistencia Equivalente	`c_eq / R_eq`	11.31	Elemento disipador de energía que gobierna la tasa de atenuación
Rigidez / Elasticidad Equivalente	`k_eq / 1/C`	64	Almacenador de energía potencial restauradora
Frecuencia Natural No Amortiguada	`ω_n [rad/s]`	8 rad/s	Distancia radial de los polos al origen en el plano complejo s
Factor de Amortiguamiento Relativo	`ζ [-]`	0.707	Determina el régimen dinámico y el sobreimpulso máximo M_p
Ganancia Estática / Constante de Tiempo	`K / τ [s]`	K = 1, τ = 0.177 s	Sensibilidad estacionaria y rapidez del modo exponencial e^{-t/τ}

3. Desarrollo Académico Diapositiva por Diapositiva

Diapositiva 01: Diagramas de Bloques en Ingeniería de Control: Elementos Fundamentales

Bloque Funcional G(s), Punto de Suma / Comparador y Punto de Ramificación o Toma

- **Turno 1 (Masculina):** En la Parte cuatro B aprenderemos a interconectar funciones de transferencia mediante diagramas de bloques y a predecir cómo la posición de los polos en el plano complejo determina la estabilidad y la respuesta transitoria.
- **Turno 2 (Femenina):** Un diagrama de bloques es la representación gráfica de las relaciones de causa y efecto en el dominio de Laplace, construida con tan solo tres símbolos elementales.
- **Turno 3 (Masculina):** El primer símbolo es el bloque rectangular que contiene una función de transferencia ge de ese: cumple una propiedad direccional estricta según la cual la señal que sale es igual al producto de ge de ese por la señal que entra.
- **Turno 4 (Femenina):** El segundo símbolo es el punto de suma o comparador circular, que suma o resta señales según el signo más o menos de cada flecha; y el tercer símbolo es el punto de ramificación o toma, desde el cual una misma señal se envía intacta hacia varios bloques simultáneamente.

Diapositiva 02: Sistemas en Serie o Cascada y el Efecto de Carga entre Etapas

Multiplicación G_eq(s) = G_1(s)·G_2(s) bajo Aislamiento de Impedancia vs. Redes Interactuantes

- **Turno 1 (Masculina):** Cuando la salida de un primer subsistema ge uno de ese se conecta directamente a la entrada de un segundo subsistema ge dos de ese, decimos que ambos bloques están conectados en serie o en cascada.
- **Turno 2 (Femenina):** Si el segundo bloque no altera ni drena corriente o fuerza significativa del primero, la función de transferencia equivalente de toda la cadena es simplemente el producto algebraico: ge uno de ese multiplicado por ge dos de ese.
- **Turno 3 (Masculina):** Pero debemos tener muchísimo cuidado con el llamado efecto de carga: si conectamos dos filtros pasivos erre ce directamente uno tras otro sin aislamiento, el segundo filtro extrae corriente del primero y modifica su ecuación diferencial.
- **Turno 4 (Femenina):** Para eliminar el efecto de carga y garantizar que podamos multiplicar legítimamente ge uno por ge dos, en electrónica intercalamos un amplificador operacional en configuración seguidor de tensión, cuya altísima impedancia de entrada aísla ambas etapas.

Diapositiva 03: Conexión de Subsistemas en Paralelo (G_eq(s) = G_1(s) ± G_2(s))

Bifurcación en Punto de Toma, Suma de Salidas y Estructura del Controlador PID

- **Turno 1 (Masculina):** La segunda topología básica es la conexión en paralelo: una misma señal de entrada u de ese se divide en un punto de toma, atraviesa simultáneamente dos o más bloques independientes y sus salidas convergen en un punto de suma.
- **Turno 2 (Femenina):** Como la salida del primer camino es ge uno de ese por u de ese, y la del segundo camino es ge dos de ese por u de ese, al sumarse y sacar factor común u de ese, la función de transferencia equivalente es la suma ge uno de ese más o menos ge dos de ese.
- **Turno 3 (Masculina):** Obsérvese que al sumar dos fracciones racionales con distinto denominador, el nuevo denominador común es el producto de los denominadores individuales, por lo que el sistema en paralelo conserva los polos de ambas ramas pero modifica la ubicación de los ceros.
- **Turno 4 (Femenina):** El ejemplo por excelencia de bloques en paralelo en mecatrónica es el controlador pe i de clásico, donde la señal de error atraviesa tres ramas paralelas: la rama proporcional ka sub pe, la rama integral ka sub i sobre ese, y la rama derivativa ka sub de por ese.

Diapositiva 04: Sistemas con Lazos de Realimentación Cerrada (Closed-Loop)

Deducción de la Ecuación Fundamental T(s) = G(s) / (1 + G(s)·H(s)) y Ecuación Característica

- **Turno 1 (Masculina):** Llegamos ahora a la estructura más importante de toda la ingeniería mecatrónica: el lazo de control con realimentación negativa, donde medimos la salida real para corregir continuamente la acción de mando.
- **Turno 2 (Femenina):** Sea erre de ese la señal de referencia deseada, ge de ese la función de transferencia de la trayectoria directa que agrupa controlador y planta, y hache de ese la función de transferencia del sensor en el camino de realimentación.
- **Turno 3 (Masculina):** Del comparador sale el error e de ese igual a la referencia erre de ese menos la señal realimentada hache de ese por ye de ese; y como la salida ye de ese es ge de ese multiplicado por ese error, sustituimos y agrupamos todos los términos con ye de ese a la izquierda.
- **Turno 4 (Femenina):** Al despejar el cociente ye de ese sobre erre de ese obtenemos la fórmula inmortal del lazo cerrado te de ese: es igual a la ganancia directa ge de ese, dividida entre uno más el producto ge de ese por hache de ese.

Diapositiva 05: Álgebra de Reducción de Diagramas de Bloques Múltiples

Movimiento de Puntos de Suma y Puntos de Toma y Reducción de Lazos Anidados en Cascada

- **Turno 1 (Masculina):** En sistemas complejos como un servomotor industrial con lazo interno de corriente, lazo intermedio de velocidad y lazo externo de posición, necesitamos aplicar las reglas del álgebra de bloques para hallar la función de transferencia global.
- **Turno 2 (Femenina):** La estrategia más limpia es reducir de adentro hacia afuera: primero reemplazamos el lazo interno de velocidad por su bloque equivalente ge sobre uno más ge por hache, luego multiplicamos en serie por el integrador uno sobre ese, y finalmente cerramos el lazo externo.
- **Turno 3 (Masculina):** Cuando dos lazos se entrecruzan, utilizamos las reglas de desplazamiento de nodos: si movemos un punto de toma desde la entrada hacia la salida de un bloque ge, debemos dividir la rama desplazada entre ge para no alterar la señal.
- **Turno 4 (Femenina):** Y de forma recíproca, si movemos un punto de toma desde la salida hacia la entrada de un bloque ge, debemos multiplicar la rama desplazada por ge, deshaciendo el cruce y dejando los lazos listos para su reducción.

Diapositiva 06: Efecto de la Realimentación sobre Perturbaciones y Variaciones de Planta

Atenuación de Perturbaciones de Carga D(s) por Factor 1/(1 + G·H) y Robustez Paramétrica

- **Turno 1 (Masculina):** ¿Por qué cerramos lazos de realimentación en lugar de operar simplemente en lazo abierto calibrando un comando fijo? La ecuación uno más ge de ese por hache de ese en el denominador nos da la respuesta matemática exacta.
- **Turno 2 (Femenina):** En primer lugar, si una perturbación de carga externa de de ese golpea la entrada de la planta, la función de transferencia desde esa perturbación hasta la salida queda dividida entre uno más el producto del controlador por la planta.
- **Turno 3 (Masculina):** Por tanto, si diseñamos un controlador con ganancia elevada en bajas frecuencias de modo que ge por hache sea mucho mayor que uno, el efecto de las perturbaciones externas sobre la salida se reduce casi a cero.
- **Turno 4 (Femenina):** En segundo lugar, cuando ge por hache es muy grande en comparación con uno, la transferencia de lazo cerrado ge sobre uno más ge por hache se aproxima a uno sobre hache de ese: el sistema se vuelve insensible al desgaste o calentamiento de la planta y depende solo de la precisión del sensor.

Diapositiva 07: El Plano Complejo s = σ + j·ω y el Mapa de Polos y Ceros

Geometría de los Polos de Lazo Cerrado: Distancia Radial ω_n y Ángulo cos(θ) = ζ

- **Turno 1 (Masculina):** Una vez obtenida la función de transferencia en lazo cerrado te de ese, las raíces de su denominador uno más ge de ese por hache de ese igual a cero son los polos de lazo cerrado, los cuales graficamos con una cruz sobre el plano complejo ese.
- **Turno 2 (Femenina):** En el plano complejo ese, el eje horizontal representa la parte real sigma en segundos a la menos uno, y el eje vertical representa la parte imaginaria jota omega en radianes por segundo.
- **Turno 3 (Masculina):** Para un par de polos complejos conjugados ubicados en menos dseta por omega sub ene más o menos jota omega sub de, su geometría en el plano complejo revela instantáneamente todos los parámetros dinámicos.
- **Turno 4 (Femenina):** Por el teorema de Pitágoras, la distancia radial desde el origen hasta el polo es exactamente la frecuencia natural no amortiguada omega sub ene; y el coseno del ángulo teta que forma ese radio con el semieje real negativo es exactamente el factor de amortiguamiento dseta.

Diapositiva 08: Efecto de la Ubicación de los Polos en la Respuesta Transitoria

Líneas Verticales (Igual t_s), Líneas Horizontales (Igual t_p) y Líneas Radiales (Igual M_p)

- **Turno 1 (Masculina):** Saber leer el mapa de polos en el plano complejo permite a un ingeniero mecatrónico visualizar la forma exacta de la respuesta en el tiempo con un simple vistazo a la ubicación de las cruces.
- **Turno 2 (Femenina):** Primera regla geométrica: como el tiempo de establecimiento es cuatro dividido entre la parte real sigma, cuanto más a la izquierda se ubiquen los polos en el plano complejo, más rápido decae la exponencial y menor es el tiempo de establecimiento; las rectas verticales son líneas de igual tiempo de establecimiento.
- **Turno 3 (Masculina):** Segunda regla: como el tiempo pico es pi dividido entre la ordenada imaginaria omega sub de, cuanto más arriba y abajo se alejen los polos del eje real, más rápido oscila el sistema; las rectas horizontales son líneas de igual tiempo pico.
- **Turno 4 (Femenina):** Tercera regla: como el sobreimpulso máximo depende únicamente de dseta y dseta es el coseno del ángulo radial teta, todas las rectas que nacen del origen con el mismo ángulo producen exactamente el mismo porcentaje de sobreimpulso.

Diapositiva 09: Estabilidad en el Plano Complejo s: Semiplano Izquierdo vs. Semiplano Derecho

Criterio de Estabilidad Asintótica (σ < 0), Estabilidad Marginal (σ = 0) e Inestabilidad (σ > 0)

- **Turno 1 (Masculina):** La propiedad más crítica que revela la ubicación de los polos en el plano ese es la estabilidad absoluta del sistema mecatrónico.
- **Turno 2 (Femenina):** Cada polo con parte real sigma aporta a la respuesta temporal un término multiplicado por e elevado a sigma por te. Si todos los polos se encuentran estrictamente en el semiplano izquierdo, donde sigma es negativo, todas las exponenciales decaen a cero y el sistema es estable.
- **Turno 3 (Masculina):** Basta con que un solo polo cruce hacia el semiplano derecho, teniendo parte real sigma positiva, para que su término e elevado a más sigma por te crezca exponencialmente sin límite, haciendo que el sistema sea inestable.
- **Turno 4 (Femenina):** Y si existe un par de polos simples exactamente sobre el eje imaginario vertical con parte real sigma igual a cero y ninguno en el semiplano derecho, el sistema ni decae ni explota: mantiene una oscilación senoidal pura y decimos que es marginalmente estable.

Diapositiva 10: Polos Dominantes, Efecto de Polos Adicionales y Ceros de Fase No Mínima

Regla de Separación 5x en el Semiplano Izquierdo y Respuesta Inversa por Cero Derecho

- **Turno 1 (Masculina):** Muchos sistemas reales son de tercer, cuarto o quinto orden, pero en la práctica se comportan casi igual que un sistema de segundo orden gracias al concepto de polos dominantes.
- **Turno 2 (Femenina):** Los polos dominantes son aquellos situados más cerca del eje imaginario: si los demás polos secundarios están al menos cinco veces más lejos hacia la izquierda, sus exponenciales rápidas se apagan en los primeros instantes y la respuesta queda gobernada por los polos dominantes.
- **Turno 3 (Masculina):** Por su parte, la presencia de un cero cercano a los polos dominantes en el semiplano izquierdo añade una acción derivativa que acelera el tiempo de subida pero incrementa notablemente el pico de sobreimpulso.
- **Turno 4 (Femenina):** Y si un sistema posee un cero en el semiplano derecho, llamado cero de fase no mínima, al aplicar un escalón positivo la salida arranca inicialmente en sentido contrario hacia valores negativos antes de rectificar y subir hacia la consigna.

4. Firmware C++/Arduino: Integrador RK4 y Filtro de Tustin (simulacion_hil_y_adquisicion_video_08.ino)

```
/*
 * ========================================================================
 * CURSO COMPLETO: MODELADO DE SISTEMAS EN MECATRÓNICA
 * VIDEO 08: Video 08 — Parte 4B: Sistemas en Serie, Lazos de Realimentación y Ubicación de Polos
 * Archivo: simulacion_hil_y_adquisicion_video_08.ino
 * ------------------------------------------------------------------------
 * Descripción Técnica:
 *   Firmware en tiempo real (Ts = 2 ms -> 500 Hz) para simulación Hardware-in-
 *   the-Loop (HIL), integración numérica Runge-Kutta de 4.º orden (RK4) de la
 *   planta mecatrónica "Lazo Cerrado de Control T(s) = G(s)/(1 + G(s)H(s)) y Geometría de Polos en el Plano
Complejo s",
 *   filtro discreto por Transformada Bilineal de Tustin y adquisición por ADC
 *   para identificación experimental de parámetros (K, tau, wn, zeta).
 *
 * Ecuación Modelo:
 *   T(s) = G(s) / (1 + G(s)·H(s))   |   1 + G(s)·H(s) = 0   |   s_{1,2} = -σ ± j·ω_d = -ζ·ω_n ± j·ω_n·√(1-ζ²)
 * ========================================================================
 */

#include <Arduino.h>

// --- Definición de Pines de Hardware ---
const uint8_t PIN_ADC_PLANTA    = A0;  // Entrada analógica desde sensor real (0..5 V)
const uint8_t PIN_ADC_PERTURB   = A1;  // Perturbación externa / carga medida
const uint8_t PIN_PWM_ACTUADOR  = 9;   // Salida PWM hacia driver / puente H / válvula
const uint8_t PIN_SYNC_SCOPE    = 8;   // Pulso de sincronismo para osciloscopio

// --- Parámetros Físicos del Modelo (Video 08) ---
const float M_EQ       = 1.0000f;   // Masa / Inercia equivalente
const float C_EQ       = 11.3100f;   // Amortiguamiento / Resistencia equivalente
const float K_EQ       = 64.0000f;   // Rigidez / Elasticidad equivalente
const float WN_RAD_S   = 8.000f;  // Frecuencia natural no amortiguada [rad/s]
const float ZETA_REL   = 0.707f; // Factor de amortiguamiento relativo [-]
const float K_GAIN     = 1.000f;   // Ganancia estática de la planta
const float TAU_SEC    = 0.177f; // Constante de tiempo equivalente [s]

// --- Configuración del Reloj de Muestreo en Tiempo Real ---
const float TS_SEC              = 0.002f; // Periodo de integración Ts = 2.0 ms (500 Hz)
const unsigned long TS_MICROS   = 2000UL;
unsigned long prevMicros        = 0;
unsigned long stepCounter       = 0;

// --- Vector de Estado del Integrador RK4: x1 = y(t), x2 = dy(t)/dt ---
float x1_pos = 0.0f;
float x2_vel = 0.0f;
float u_cmd  = 0.0f;

// --- Filtro Digital de 1.er Orden Discretizado por Transformada de Tustin ---
// G_f(s) = K / (tau*s + 1)  -->  y[k] = b0*u[k] + b1*u[k-1] - a1*y[k-1]
float b0_tustin = 0.0f;
float b1_tustin = 0.0f;
float a1_tustin = 0.0f;
float u_prev1   = 0.0f;
float y_tustin  = 0.0f;

// --- Derivadas del Espacio de Estados Continuo: dx1/dt = x2, dx2/dt = f(x1,x2,u) ---
inline void evaluarDerivadasEstado(float x1, float x2, float u, float &dx1, float &dx2) {
  dx1 = x2;
  // Forma canónica de 2.º orden: x2' = K*wn^2*u - 2*zeta*wn*x2 - wn^2*x1
  dx2 = (K_GAIN * WN_RAD_S * WN_RAD_S * u)
      - (2.0f * ZETA_REL * WN_RAD_S * x2)
      - (WN_RAD_S * WN_RAD_S * x1);
}

// --- Paso de Integración Numérica Runge-Kutta de 4.º Orden (RK4) ---
void integrarPasoRK4(float u_in, float dt) {
  float k1_x1, k1_x2, k2_x1, k2_x2, k3_x1, k3_x2, k4_x1, k4_x2;
```

```
  evaluarDerivadasEstado(x1_pos, x2_vel, u_in, k1_x1, k1_x2);
  evaluarDerivadasEstado(x1_pos + 0.5f * dt * k1_x1, x2_vel + 0.5f * dt * k1_x2, u_in, k2_x1, k2_x2);
  evaluarDerivadasEstado(x1_pos + 0.5f * dt * k2_x1, x2_vel + 0.5f * dt * k2_x2, u_in, k3_x1, k3_x2);
  evaluarDerivadasEstado(x1_pos + dt * k3_x1,        x2_vel + dt * k3_x2,        u_in, k4_x1, k4_x2);

  x1_pos += (dt / 6.0f) * (k1_x1 + 2.0f * k2_x1 + 2.0f * k3_x1 + k4_x1);
  x2_vel += (dt / 6.0f) * (k1_x2 + 2.0f * k2_x2 + 2.0f * k3_x2 + k4_x2);
}

// --- Inicialización de Coeficientes Discretos por Regla Trapezoidal (Tustin) ---
void inicializarFiltroTustin() {
  float denom = 2.0f * TAU_SEC + TS_SEC;
  b0_tustin   = (K_GAIN * TS_SEC) / denom;
  b1_tustin   = b0_tustin;
  a1_tustin   = (TS_SEC - 2.0f * TAU_SEC) / denom;
}

void setup() {
  Serial.begin(115200);
  pinMode(PIN_PWM_ACTUADOR, OUTPUT);
  pinMode(PIN_SYNC_SCOPE, OUTPUT);
  inicializarFiltroTustin();
  prevMicros = micros();

  Serial.println(F("# ============================================================"));
  Serial.println(F("# CURSO MODELADO DE SISTEMAS EN MECATRONICA - VIDEO 08"));
  Serial.println(F("# Columnas CSV: t_s,u_cmd,y_rk4_2do,dy_rk4,y_tustin_1ro,adc_V"));
  Serial.println(F("# ============================================================"));
}

void loop() {
  unsigned long nowMicros = micros();
  if (nowMicros - prevMicros >= TS_MICROS) {
    prevMicros += TS_MICROS;
    float t_sec = stepCounter * TS_SEC;

    // 1. Generar entrada de prueba: Escalón unitario + barrido senoidal suave
    if (t_sec < 0.2f) {
      u_cmd = 0.0f;
    } else if (t_sec < 3.0f) {
      u_cmd = 1.0f; // Respuesta al escalón unitario u_s(t)
      digitalWrite(PIN_SYNC_SCOPE, HIGH);
    } else {
      // Excitación armónica para medir respuesta en frecuencia G(jw)
      u_cmd = 1.0f + 0.5f * sinf(WN_RAD_S * (t_sec - 3.0f));
      digitalWrite(PIN_SYNC_SCOPE, LOW);
    }

    // 2. Actualizar gemelo digital de 2.º orden por RK4
    integrarPasoRK4(u_cmd, TS_SEC);

    // 3. Actualizar modelo de 1.er orden por Transformada de Tustin
    y_tustin = b0_tustin * u_cmd + b1_tustin * u_prev1 - a1_tustin * y_tustin;
    u_prev1  = u_cmd;

    // 4. Leer planta física real en el ADC y comandar salida PWM
    float adcVolt = analogRead(PIN_ADC_PLANTA) * (5.0f / 1023.0f);
    int pwmVal    = constrain((int)(x1_pos * 127.0f + 128.0f), 0, 255);
    analogWrite(PIN_PWM_ACTUADOR, pwmVal);

    // 5. Transmitir telemetría cada 5 pasos (10 ms -> 100 Hz)
    if (stepCounter % 5 == 0) {
      Serial.print(t_sec, 3);      Serial.print(',');
      Serial.print(u_cmd, 4);      Serial.print(',');
      Serial.print(x1_pos, 4);     Serial.print(',');
      Serial.print(x2_vel, 4);     Serial.print(',');
      Serial.print(y_tustin, 4);   Serial.print(',');
```

```
      Serial.println(adcVolt, 3);
    }
    stepCounter++;
  }
}
```

5. Script Python: Polos, Respuesta Temporal y Bode (modelado_laplace_y_bode_video_08.py)

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
================================================================================
CURSO COMPLETO: MODELADO DE SISTEMAS EN MECATRÓNICA
VIDEO 08: Video 08 – Parte 4B: Sistemas en Serie, Lazos de Realimentación y Ubicación de Polos
Archivo: modelado_laplace_y_bode_video_08.py
--------------------------------------------------------------------------------
Descripción:
  Script integral en Python para simulación de ecuaciones diferenciales (RK4),
  análisis de sistemas de 1.er y 2.º orden, cálculo exacto de polos y ceros
  en el plano complejo s, especificaciones transitorias (tr, tp, Mp, ts),
  evaluación fasorial G(jw), trazado de Diagramas de Bode y cálculo de
  Márgenes de Ganancia (MG) y de Fase (MF).
================================================================================
"""

import math
from typing import Dict, List, Tuple

# Parámetros nominales del Video 08: Lazo Cerrado de Control T(s) = G(s)/(1 + G(s)H(s)) y Geometría de Polos en el
Plano Complejo s
M_EQ     = 1
C_EQ     = 11.31
K_EQ     = 64
WN_RAD   = 8
ZETA     = 0.707
K_STATIC = 1
TAU_SEC  = 0.177


def metricas_primer_orden(K: float, tau: float, escalon_A: float = 1.0) -> Dict[str, float]:
    """Calcula las medidas de desempeño exactas de G(s) = K / (tau*s + 1)."""
    y_ss = K * escalon_A
    y_1tau = y_ss * (1.0 - math.exp(-1.0))
    t_d = tau * math.log(2.0)          # 50% del valor final
    t_r = tau * math.log(9.0)          # 10% al 90% (~2.197 * tau)
    t_s_5pct = 3.0 * tau               # Banda del +-5%
    t_s_2pct = 4.0 * tau               # Banda del +-2%
    omega_c = 1.0 / tau                # Frecuencia de esquina (-3 dB, -45 deg)
    return {
        "y_ss": round(y_ss, 4),
        "y_en_1tau_63pct": round(y_1tau, 4),
        "tiempo_retardo_td_s": round(t_d, 4),
        "tiempo_subida_tr_s": round(t_r, 4),
        "tiempo_establecimiento_5pct_s": round(t_s_5pct, 4),
        "tiempo_establecimiento_2pct_s": round(t_s_2pct, 4),
        "frecuencia_esquina_wc_rads": round(omega_c, 4),
    }


def metricas_segundo_orden(K: float, wn: float, zeta: float) -> Dict[str, object]:
    """Calcula polos en el plano s y métricas (wd, tr, tp, Mp, ts, wr, Mr, w_BW)."""
    sigma = zeta * wn
    if zeta < 1.0:
        wd = wn * math.sqrt(1.0 - zeta ** 2)
        polos = f"-{sigma:.3f} +- j*{wd:.3f}"
        regimen = "Subamortiguado (0 < zeta < 1)" if zeta > 0 else "No amortiguado"
        tr = (math.pi - math.acos(zeta)) / wd
        tp = math.pi / wd
        mp = math.exp(-math.pi * zeta / math.sqrt(1.0 - zeta ** 2))
    elif abs(zeta - 1.0) < 1e-6:
        wd = 0.0
        polos = f"-{sigma:.3f} (Raíz real doble)"
        regimen = "Críticamente amortiguado (zeta = 1)"
        tr = 3.35 / wn
        tp = float("inf")
```

```
        mp = 0.0
    else:
        wd = 0.0
        r1 = -sigma + wn * math.sqrt(zeta ** 2 - 1.0)
        r2 = -sigma - wn * math.sqrt(zeta ** 2 - 1.0)
        polos = f"s1 = {r1:.3f}, s2 = {r2:.3f}"
        regimen = "Sobreamortiguado (zeta > 1)"
        tr = 2.2 / abs(r1)
        tp = float("inf")
        mp = 0.0

    ts_2pct = 4.0 / sigma if sigma > 0 else float("inf")

    # Resonancia y Ancho de Banda en frecuencia
    if zeta < 1.0 / math.sqrt(2.0):
        wr = wn * math.sqrt(1.0 - 2.0 * (zeta ** 2))
        mr = 1.0 / (2.0 * zeta * math.sqrt(1.0 - zeta ** 2))
        mr_db = 20.0 * math.log10(mr)
    else:
        wr = 0.0
        mr = 1.0
        mr_db = 0.0

    w_bw = wn * math.sqrt((1.0 - 2.0 * zeta ** 2) + math.sqrt(4.0 * zeta ** 4 - 4.0 * zeta ** 2 + 2.0))

    return {
        "regimen": regimen,
        "polos_plano_s": polos,
        "atenuacion_sigma": round(sigma, 4),
        "frecuencia_amortiguada_wd_rads": round(wd, 4),
        "tiempo_subida_tr_s": round(tr, 4),
        "tiempo_pico_tp_s": round(tp, 4) if math.isfinite(tp) else None,
        "sobreimpulso_Mp_pct": round(mp * 100.0, 2),
        "tiempo_establecimiento_ts_2pct_s": round(ts_2pct, 4),
        "frecuencia_resonancia_wr_rads": round(wr, 4),
        "pico_resonancia_Mr_dB": round(mr_db, 2),
        "ancho_banda_wBW_rads": round(w_bw, 4),
    }


def evaluar_respuesta_frecuencia_bode(
    K: float, wn: float, zeta: float, frecuencias_rads: List[float]
) -> List[Dict[str, float]]:
    """Evalúa G(jw) = K / ((1 - u^2) + j*(2*zeta*u)) con u = w / wn."""
    tabla_bode = []
    for w in frecuencias_rads:
        u = w / wn
        parte_real = 1.0 - (u ** 2)
        parte_imag = 2.0 * zeta * u
        mag_abs = K / math.sqrt(parte_real ** 2 + parte_imag ** 2)
        mag_db = 20.0 * math.log10(max(mag_abs, 1e-12))
        fase_deg = -math.degrees(math.atan2(parte_imag, parte_real))
        tabla_bode.append({
            "w_rads": round(w, 3),
            "u_norm": round(u, 3),
            "mag_abs": round(mag_abs, 4),
            "mag_dB": round(mag_db, 2),
            "fase_deg": round(fase_deg, 2),
        })
    return tabla_bode


def identificar_segundo_orden_desde_escalon(
    u_escalon: float, y_estacionario: float, y_pico_max: float, t_pico_s: float
) -> Dict[str, float]:
    """Identifica experimentalmente (K, Mp, zeta, wn) a partir de la curva medida."""
    K_id = y_estacionario / u_escalon
    mp_id = max(0.0, (y_pico_max - y_estacionario) / y_estacionario)
```

```
    if mp_id > 0.0:
        ln_mp = abs(math.log(mp_id))
        zeta_id = ln_mp / math.sqrt(math.pi ** 2 + ln_mp ** 2)
    else:
        zeta_id = 1.0
    wn_id = math.pi / (t_pico_s * math.sqrt(max(1e-6, 1.0 - zeta_id ** 2)))
    return {
        "K_identificada": round(K_id, 4),
        "Mp_medido": round(mp_id, 4),
        "zeta_identificado": round(zeta_id, 4),
        "wn_identificada_rads": round(wn_id, 4),
    }


if __name__ == "__main__":
    print("=" * 76)
    print(" CURSO: MODELADO DE SISTEMAS EN MECATRÓNICA — VIDEO 08")
    print(" Planta:", "Lazo Cerrado de Control T(s) = G(s)/(1 + G(s)H(s)) y Geometría de Polos en el Plano
Complejo s")
    print("=" * 76)

    m1 = metricas_primer_orden(K_STATIC, TAU_SEC)
    print("\n[1] Análisis de Sistema de Primer Orden (K, tau):", m1)

    m2 = metricas_segundo_orden(K_STATIC, WN_RAD, ZETA)
    print("\n[2] Análisis de Sistema de Segundo Orden (K, wn, zeta):", m2)

    freqs = [0.1 * WN_RAD, 0.5 * WN_RAD, WN_RAD, 2.0 * WN_RAD, 10.0 * WN_RAD]
    bode_pts = evaluar_respuesta_frecuencia_bode(K_STATIC, WN_RAD, ZETA, freqs)
    print("\n[3] Tabla de Respuesta en Frecuencia y Diagrama de Bode G(jw):")
    for row in bode_pts:
        print(f"    w = {row['w_rads']:7.2f} rad/s | |G| = {row['mag_abs']:6.4f} ({row['mag_dB']:6.2f} dB) | fase
= {row['fase_deg']:7.2f} deg")

    id_res = identificar_segundo_orden_desde_escalon(1.0, 1.0, 1.254, 0.343)
    print("\n[4] Ejemplo de Identificación Experimental desde Escalón:", id_res)
```

6. Script MATLAB/Octave: Funciones de Transferencia y Estabilidad (analisis_control_mecatronico_video_08.m)

```
% ========================================================================
% CURSO COMPLETO: MODELADO DE SISTEMAS EN MECATRÓNICA
% VIDEO 08: Video 08 — Parte 4B: Sistemas en Serie, Lazos de Realimentación y Ubicación de Polos
% Archivo: analisis_control_mecatronico_video_08.m
% ------------------------------------------------------------------------
% Descripción:
%   Script para MATLAB / GNU Octave que construye las funciones de
%   transferencia en Laplace con tf(), calcula polos, ceros y factor de
%   amortiguamiento con damp(), simula la respuesta al escalón step() y al
%   impulso impulse(), cierra el lazo de realimentación con feedback(),
%   traza el Diagrama de Bode con márgenes de estabilidad margin() y el
%   Lugar Geométrico de las Raíces rlocus().
% ========================================================================

clear; clc; close all;
fprintf('========================================================================\n');
fprintf(' CURSO: MODELADO DE SISTEMAS EN MECATRONICA - VIDEO 08\n');
fprintf(' Planta: Lazo Cerrado de Control T(s) = G(s)/(1 + G(s)H(s)) y Geometría de Polos en el Plano Complejo
s\n');
fprintf('========================================================================\n\n');

%% 1. Definicion de Parametros Fisicos y Canónicos
m_eq   = 1;
c_eq   = 11.31;
k_eq   = 64;
wn     = 8;
zeta   = 0.707;
K_gain = 1;
tau    = 0.177;

s = tf('s');

%% 2. Funcion de Transferencia de Primer Orden: G1(s) = K / (tau*s + 1)
G1 = K_gain / (tau * s + 1);
fprintf('--- Sistema de Primer Orden G1(s) ---\n');
disp(G1);

%% 3. Funcion de Transferencia de Segundo Orden: G2(s) = K*wn^2 / (s^2 + 2*zeta*wn*s + wn^2)
num_G2 = [64.0];
den_G2 = [1.0, 11.31, 64.0];
G2 = tf(num_G2, den_G2);
fprintf('--- Sistema Principal del Video 08: G2(s) ---\n');
disp(G2);

fprintf('--- Polos, Factor de Amortiguamiento (zeta) y Frecuencia Natural (wn) ---\n');
damp(G2);
polos_G2 = pole(G2);
ceros_G2 = zero(G2);

%% 4. Reduccion de Diagramas de Bloques: Serie, Paralelo y Lazo Cerrado T(s)
Kp_ctrl = 4.0;
Ki_ctrl = 8.0;
Gc = Kp_ctrl + Ki_ctrl / s;          % Controlador PI en paralelo
L_abierto = series(Gc, G2);          % Sistemas en serie L(s) = Gc(s)*G2(s)
H_sensor  = 1.0;                     % Realimentacion unitaria H(s) = 1
T_cerrado = feedback(L_abierto, H_sensor, -1); % T(s) = G(s) / (1 + G(s)H(s))

fprintf('--- Funcion de Transferencia en Lazo Cerrado T(s) ---\n');
disp(T_cerrado);

%% 5. Graficas en el Dominio del Tiempo (Escalon e Impulso) y Plano Complejo s
figure('Name', 'Video 08 - Respuestas Dinamicas y Mapa de Polos', 'Color', 'w', 'Position', [80 80 1100 720]);

subplot(2,2,1);
step(G1, 'b--', G2, 'r-', T_cerrado, 'g-.');
grid on;
title('Respuesta al Escalon Unitario u_s(t)');
```

```
legend('1.er Orden G_1(s)', 'Planta G_2(s)', 'Lazo Cerrado T(s)', 'Location', 'best');

subplot(2,2,2);
impulse(G2, 'r-', T_cerrado, 'g-.');
grid on;
title('Respuesta al Impulso Unitario \delta(t)');
legend('Planta G_2(s)', 'Lazo Cerrado T(s)', 'Location', 'best');

subplot(2,2,3);
pzmap(G2, 'r', T_cerrado, 'g');
grid on;
title('Ubicacion de Polos (x) y Ceros (o) en el Plano Complejo s');

subplot(2,2,4);
rlocus(L_abierto);
grid on;
title('Lugar Geometrico de las Raices (Root Locus)');

%% 6. Respuesta en Frecuencia: Diagrama de Bode y Margenes de Estabilidad (MG, MF)
figure('Name', 'Video 08 - Diagrama de Bode y Estabilidad Relativa', 'Color', 'w', 'Position', [140 100 980 680]);
margin(L_abierto);
grid on;

[Gm, Pm, Wcg, Wcp] = margin(L_abierto);
Gm_dB = 20 * log10(Gm);
fprintf('\n=== ESPECIFICACIONES DE ESTABILIDAD EN FRECUENCIA (VIDEO 08) ===\n');
fprintf('  Margen de Ganancia (MG)            : %.2f dB (en w_pc = %.3f rad/s)\n', Gm_dB, Wcp);
fprintf('  Margen de Fase (MF)                : %.2f deg (en w_gc = %.3f rad/s)\n', Pm, Wcg);
fprintf('  Amortiguamiento equivalente (MF/100): %.3f\n', Pm / 100.0);
```