Curso de Sistemas y Control en Mecatrónica — Guía de Estudio Video 01

Módulo del Curso: Módulo 1 — El Rol del Control, el Modelado y Señales y Sistemas (Laplace, Z y Espacio de Estados)
Enfoque de Ingeniería: Muestreo Periódico con ZOH, Función de Transferencia Continua G(s) y Discreta G(z), Transformación Bilineal de Tustin y Modelos en Espacio de Estados
Plataforma Oficial: `cursosmecatronica.com`

```
z = e^(s·Ts) ≈ (1 + s·Ts/2)/(1 - s·Ts/2) | G(s) = C(sI - A)⁻¹B + D |
x[k+1] = e^(A·Ts) x[k] + (∫ e^(Aτ)dτ B) u[k]
```

1. Síntesis de Conceptos Clave y Reglas de Diseño

- El rol del control en mecatrónica: compensación activa de flexibilidad, fricción e inercia variable mediante co-diseño concurrente.
- El rol del modelado en el ciclo en V: modelos físicos de parámetros concentrados orientados al control e identificación experimental.
- Señales en tiempo continuo y discreto, muestreo periódico, retenedor de orden cero (ZOH), linealidad, invarianza temporal (LTI), causalidad y estabilidad BIBO.
- Transformada de Laplace G(s), Transformada Z G(z), discretización por Euler y Tustin con prewarping, y formulación de modelos en espacio de estados continuos (A,B,C,D) y discretos (Φ,Γ,C,D).

2. Parámetros Nominales de Sistemas y Control (Video 01)

Parámetro de Ingeniería	Símbolo	Valor Nominal	Función en el Sistema de Control Mecatrónico
Periodo de Muestreo Digital	`Ts [ms]`	1 ms	Intervalo determinista de discretización ZOH y ejecución de Kalman/LQR
Frecuencia Natural y Amortiguamiento	`ωn / ζ`	50 rad/s / 0.7	Ubicación de polos dominantes en lazo cerrado
Márgenes de Fase y Ganancia	`PM / GM`	52° / 12 dB	Márgenes de estabilidad robusta en respuesta en frecuencia

Parámetro de Ingeniería	Símbolo	Valor Nominal	Función en el Sistema de Control Mecatrónico
Covarianzas del Filtro de Kalman	`Q / R`	Q=0.0001, R=0.005	Matrices de incertidumbre de proceso y de ruido de sensores

3. Desarrollo Académico Diapositiva por Diapositiva

Diapositiva 01: El Rol del Control en Mecatrónica: Sinergia, Precisión, Ancho de Banda y Estabilidad

De la Secuenciación Pasiva al Co-Diseño Activo que Compensa Flexibilidad Estructural, Fricción y Perturbaciones

Turno 1 (Masculina): En la ingeniería tradicional, la estructura mecánica se diseñaba primero como un cuerpo rígido sobredimensionado, luego se añadían los motores y finalmente se acoplaba un controlador tratando de estabilizar una planta ya construida.

Turno 2 (Femenina): En el diseño mecatrónico moderno, el sistema de control cumple un rol fundacional desde el primer boceto: permite aligerar eslabones mecánicos, aumentar el ancho de banda dinámico y estabilizar plantas intrínsecamente inestables.

Turno 3 (Masculina): Gracias a los lazos de realimentación de alta velocidad, un controlador compensa activamente no linealidades severas como la fricción de Coulomb, las variaciones de inercia de la carga y las perturbaciones externas de par.

Turno 4 (Femenina): Así, el control deja de ser un simple accesorio electrónico para convertirse en el cerebro integrador que coordina sensores, actuadores y estructura mecánica bajo métricas estrictas de precisión micrométrica y eficiencia energética.

Diapositiva 02: El Rol del Modelado en el Diseño Mecatrónico: Ciclo en V, Física e Identificación

Modelos de Parámetros Concentrados orientados al Control, Validación Experimental y Gemelos Digitales

Turno 1 (Masculina): Ningún controlador avanzado puede diseñarse a ciegas: el modelado matemático es el lenguaje universal que unifica los dominios mecánico, eléctrico, térmico e hidráulico dentro de un mismo sistema de ecuaciones diferenciales.

Turno 2 (Femenina): Dentro de la metodología de diseño en V, el ingeniero distingue entre modelos de simulación de alta fidelidad y modelos reducidos orientados al control, los cuales capturan los modos dominantes sin exceder la capacidad de cómputo embebido.

Turno 3 (Masculina): Construimos el modelo analítico mediante las leyes físicas de Newton, Lagrange, Kirchhoff y conservación de masa, y posteriormente ajustamos sus parámetros

desconocidos mediante ensayos de identificación experimental sobre el prototipo.

Turno 4 (Femenina): Este modelo validado permite sintonizar observadores, verificar márgenes de estabilidad y ejecutar pruebas exhaustivas con hardware en el lazo antes de arriesgar componentes mecánicos reales en planta.

Diapositiva 03: Señales en Tiempo Continuo y en Tiempo Discreto: Muestreo, ZOH y Cuantificación

Conversión Analógica-Digital Periódica x[k] = x(kTs), Retenedor de Orden Cero y Ruido de Cuantificación

Turno 1 (Masculina): En todo sistema mecatrónico conviven dos naturalezas de señal: las variables físicas de la planta evolucionan en tiempo continuo, mientras que el microcontrolador o procesador digital opera exclusivamente en instantes discretos.

Turno 2 (Femenina): El conversor analógico a digital toma muestras periódicas de la señal continua cada intervalo de muestreo T sub s, generando una secuencia numérica indexada x de k que además está cuantificada en niveles finitos de resolución.

Turno 3 (Masculina): En el camino inverso hacia el actuador, el conversor digital a analógico o el modulador PWM actúa como un Retenedor de Orden Cero, manteniendo constante el comando de control entre dos instantes de muestreo consecutivos.

Turno 4 (Femenina): Este mantenimiento escalonado introduce un retardo efectivo equivalente a medio periodo de muestreo, cuyo desfase debe contabilizarse rigurosamente para no degradar el margen de fase del lazo cerrado.

Diapositiva 04: Propiedades Fundamentales de los Sistemas: Linealidad, Invarianza, Causalidad y BIBO

Principio de Superposición Homogénea, Convolución Continua y Discreta, y Estabilidad Entrada-Salida Acotada

Turno 1 (Masculina): Un sistema mecatrónico se modela como un operador matemático que transforma una señal de excitación de entrada u en una respuesta de salida y, caracterizándose por cuatro propiedades estructurales esenciales.

Turno 2 (Femenina): Un sistema es lineal si obedece simultáneamente los principios de aditividad y homogeneidad, y es invariante en el tiempo cuando un desplazamiento temporal en la entrada produce exactamente el mismo desplazamiento en la salida.

Turno 3 (Masculina): En los sistemas lineales e invariantes en el tiempo, conocidos como L T I, la salida ante cualquier entrada arbitraria queda completamente determinada por la

integral o suma de convolución con la respuesta al impulso del sistema.

Turno 4 (Femenina): Además, todo controlador físico en tiempo real debe ser estrictamente causal, dependiendo solo de entradas presentes y pasadas, y debe cumplir la estabilidad BIBO: toda entrada acotada debe producir siempre una salida acotada.

Diapositiva 05: Funciones de Transferencia y Transformada de Laplace en Sistemas Continuos

Álgebra Operacional en el Plano Complejo s = σ + jω, Polos, Ceros y Dinámica de Ecuaciones Diferenciales

Turno 1 (Masculina): La Transformada de Laplace convierte las ecuaciones diferenciales lineales en el dominio del tiempo continuo en ecuaciones algebraicas polinómicas sobre la variable compleja s igual a sigma más jota omega.

Turno 2 (Femenina): Al asumir condiciones iniciales nulas, definimos la Función de Transferencia continua G de s como el cociente entre la transformada de Laplace de la salida Y de s y la transformada de la entrada U de s.

Turno 3 (Masculina): Las raíces del polinomio denominador son los polos del sistema, los cuales gobiernan la estabilidad y los modos naturales de respuesta, mientras que las raíces del numerador son los ceros, que moldean el peso relativo de cada modo.

Turno 4 (Femenina): Para que un sistema continuo sea asintóticamente estable, absolutamente todos sus polos deben ubicarse en el semiplano izquierdo abierto del plano s, es decir, con parte real sigma estrictamente negativa.

Diapositiva 06: Transformada Z y Sistemas Digitales: Ecuaciones en Diferencias y Plano Complejo z

Operador de Retardo Unitario z^{-1}, Función de Transferencia Pulso G(z) y Estabilidad dentro del Círculo Unitario

Turno 1 (Masculina): Para analizar e implementar controladores digitales en microcontroladores, empleamos la Transformada Z, que cumple para las secuencias en tiempo discreto el mismo papel que la Transformada de Laplace en tiempo continuo.

Turno 2 (Femenina): Su propiedad operacional clave es el desplazamiento temporal: multiplicar una señal transformada por z elevado a la menos uno equivale exactamente a retrasar la muestra un periodo de muestreo T sub s en el código del procesador.

Turno 3 (Masculina): Gracias a ello, cualquier función de transferencia discreta G de z se traduce de manera directa en una ecuación en diferencias recursiva que calcula la salida

actual combinando muestras pasadas de entrada y de salida.

Turno 4 (Femenina): En el plano complejo z, la frontera de estabilidad deja de ser el eje imaginario y pasa a ser el círculo unitario: un sistema digital es estable si y solo si todos sus polos tienen magnitud estrictamente menor que uno.

Diapositiva 07: Discretización de Sistemas Continuos: Mapeo Exacto z = e^(sTs), Euler y Tustin

Equivalente Discreto con Retenedor ZOH, Aproximación Bilineal Trapecial y Pre-Deformación de Frecuencia (Prewarping)

Turno 1 (Masculina): Cuando diseñamos un controlador en tiempo continuo y deseamos programarlo en un sistema embebido, o cuando discretizamos la planta física vista a través de un retenedor de orden cero, debemos mapear el plano s hacia el plano z.

Turno 2 (Femenina): La relación exacta entre ambos dominios está dada por z igual a la exponencial de s por T sub s, la cual transforma el semiplano izquierdo estable de Laplace en el interior del círculo unitario del plano z.

Turno 3 (Masculina): En la práctica numérica, la aproximación de Euler hacia atrás sustituye s por uno menos z a la menos uno sobre T sub s, mientras que la transformación bilineal o regla trapezoidal de Tustin ofrece mucha mayor exactitud en fase.

Turno 4 (Femenina): Al aplicar Tustin con pre-deformación de frecuencia o prewarping, garantizamos que la frecuencia crítica de cruce o de corte de un filtro continuo coincida exactamente con la respuesta del filtro digital resultante.

Diapositiva 08: Modelos en Espacio de Estados en Tiempo Continuo: dx/dt = Ax + Bu, y = Cx + Du

Vector de Estado Físico Interno, Matrices del Sistema (A, B, C, D) y Representación Natural de Sistemas MIMO

Turno 1 (Masculina): Mientras la función de transferencia describe únicamente la relación externa entre una entrada y una salida con condiciones iniciales nulas, el modelo en espacio de estados revela la dinámica interna completa del sistema mecatrónico.

Turno 2 (Femenina): El vector de estado x agrupa el conjunto mínimo de variables físicas, tales como posiciones, velocidades, corrientes de inductancia y tensiones de condensador, cuyo valor actual basta para predecir toda la evolución futura.

Turno 3 (Masculina): Una ecuación diferencial de orden n se reescribe como un sistema matricial de primer orden: la derivada del estado x punto es igual a la matriz dinámica A por x más la matriz de entrada B por el vector de control u.

Turno 4 (Femenina): Simultáneamente, el vector de mediciones de los sensores y es igual a la matriz de salida C por x más la matriz de transmisión directa D por u, permitiendo modelar con idéntica elegancia plantas de múltiples entradas y salidas.

Diapositiva 09: Modelos en Espacio de Estados en Tiempo Discreto y Equivalencia con G(s) y G(z)

Ecuación de Estado Discreta x[k+1] = Φx[k] + Γu[k], Cálculo de Φ = e^(ATs) y Conversión a Función de Transferencia

Turno 1 (Masculina): Al muestrear un sistema continuo en espacio de estados con un retenedor de orden cero de periodo T sub s, obtenemos el modelo de estado discreto exacto: x en k más uno es igual a Fi por x en k más Gamma por u en k.

Turno 2 (Femenina): Aquí la matriz de transición discreta Fi se calcula mediante la exponencial matricial e elevado a la matriz A por T sub s, y la matriz de entrada discreta Gamma es la integral de cero a T sub s de e elevado a A tau por B.

Turno 3 (Masculina): Ambas representaciones, estado y transferencia, están íntimamente conectadas: la función de transferencia continua G de s se obtiene como C por la inversa de s por la identidad menos A, multiplicado por B más D.

Turno 4 (Femenina): Esto demuestra que los polos de la función de transferencia son un subconjunto de los autovalores de la matriz A, salvo cuando ocurren cancelaciones internas entre polos y ceros que el espacio de estados sí permite detectar.

Diapositiva 10: Síntesis del Módulo 1 y Transición al Análisis en Espacio de Estados y Respuesta Dinámica

Integración de Co-Diseño, Señales Muestreadas, Transformadas Operacionales y Formulación Matricial de Estado

Turno 1 (Masculina): En este primer módulo establecimos el rol estratégico del control y del modelado matemático en el diseño mecatrónico, así como las propiedades fundamentales de señales y sistemas continuos y discretos.

Turno 2 (Femenina): Dominamos el álgebra de funciones de transferencia mediante la Transformada de Laplace en el plano s y la Transformada Z en el plano z, junto con los métodos exactos y bilineales de discretización.

Turno 3 (Masculina): Finalmente formulamos los modelos matriciales en espacio de estados tanto en tiempo continuo como en tiempo discreto, sentando las bases de la teoría moderna de control multivariable.

Turno 4 (Femenina): En el Módulo dos profundizaremos en el análisis estructural en espacio de estados, evaluando controlabilidad, observabilidad, estabilidad y la respuesta temporal de sistemas dinámicos.

4. Firmware C++/MCU: Filtro de Kalman, Biquad DSP, LQR (H2) y Modos Deslizantes (control_sistemas_mecatronica_video_01.ino)

```
/*
 * ========================================================================
 * CURSO: SISTEMAS Y CONTROL EN MECATRÓNICA (GEMINI — cursosmecatronica.com)
 * VIDEO 01: Video 01 — El Rol del Control, el Modelado y Señales y Sistemas en Mecatrónica
 * Enfoque: Muestreo Periódico con ZOH, Función de Transferencia Continua G(s) y Discreta G(z),
Transformación Bilineal de Tustin y Modelos en Espacio de Estados
 * Ecuación Clave: z = e^(s·Ts) ≈ (1 + s·Ts/2)/(1 - s·Ts/2)  |  G(s) = C(sI - A)⁻¹B + D  |  x[k+1] =
e^(A·Ts) x[k] + (∫ e^(Aτ)dτ B) u[k]
 * ========================================================================
 */
#include <Arduino.h>

// Parámetros de tiempo real y control en espacio de estados discreto
static const float TS_SEC = 0.0010f;
static const uint32_t TS_US = 1000UL;

// Estado estimado por Filtro de Kalman [posición x1, velocidad x2, perturbación d]
static float x_hat[3] = {0.0f, 0.0f, 0.0f};
static float P_cov[2][2] = {{1.0f, 0.0f}, {0.0f, 1.0f}};
static const float Q_PROC = 0.0001f;
static const float R_MEAS = 0.005f;

// Ganancias óptimas LQR (H2) + Superficie de Modos Deslizantes (SMC)
static const float K_LQR_POS = 22.50f;
static const float K_LQR_VEL = 9.10f;
static const float LAMBDA_SMC = 17.50f;
static const float K_SW_SMC = 2.80f;
static const float PHI_BOUND = 0.05f;

// Filtro digital IIR Biquad Notch (supresión de resonancia mecánica/hidráulica)
struct BiquadFilter {
  float b0, b1, b2, a1, a2;
  float w1, w2;
};
static BiquadFilter notchFilter = {0.94f, -1.78f, 0.94f, -1.78f, 0.88f, 0.0f, 0.0f};

static inline float stepBiquad(BiquadFilter &f, float x) {
  float w0 = x - f.a1 * f.w1 - f.a2 * f.w2;
```

```
  float y  = f.b0 * w0 + f.b1 * f.w1 + f.b2 * f.w2;
  f.w2 = f.w1;
  f.w1 = w0;
  return y;
}

static inline float satBoundary(float s, float phi) {
  if (s > phi) return 1.0f;
  if (s < -phi) return -1.0f;
  return s / phi;
}

void kalmanObserverStep(float u_prev, float y_meas) {
  // 1. Predicción temporal: x_hat^- = Phi * x_hat + Gamma * u
  const float b_fric = 1.25f;
  const float k_gain = 18.5f;
  float x1_pred = x_hat[0] + TS_SEC * x_hat[1];
  float x2_pred = x_hat[1] + TS_SEC * (-b_fric * x_hat[1] + k_gain * u_prev - x_hat[2]);

  // Propagación de covarianza P^- = Phi * P * Phi^T + Q
  P_cov[0][0] += TS_SEC * (2.0f * P_cov[0][1] + TS_SEC * P_cov[1][1]) + Q_PROC;
  P_cov[0][1] += TS_SEC * P_cov[1][1];
  P_cov[1][0]  = P_cov[0][1];
  P_cov[1][1] += Q_PROC * 10.0f;

  // 2. Actualización por medición (Innovación y Ganancia de Kalman K_k)
  float innov = y_meas - x1_pred;
  float S_inv = 1.0f / (P_cov[0][0] + R_MEAS);
  float K0 = P_cov[0][0] * S_inv;
  float K1 = P_cov[1][0] * S_inv;

  x_hat[0] = x1_pred + K0 * innov;
  x_hat[1] = x2_pred + K1 * innov;
  x_hat[2] -= 0.08f * innov; // Estimación adaptativa de carga externa

  float p00 = P_cov[0][0], p01 = P_cov[0][1];
  P_cov[0][0] -= K0 * p00;
  P_cov[0][1] -= K0 * p01;
  P_cov[1][0] -= K1 * p00;
  P_cov[1][1] -= K1 * p01;
}

void setup() {
  Serial.begin(115200);
  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
}

void loop() {
  static uint32_t lastTickUs = 0;
  static float u_cmd = 0.0f;
  uint32_t nowUs = micros();

  if ((uint32_t)(nowUs - lastTickUs) >= TS_US) {
    lastTickUs += TS_US;
```

```
    // Lectura de referencia y posición + filtrado DSP Biquad
    float ref_pos = (analogRead(A0) - 512) * 0.01f;
    float raw_pos = (analogRead(A1) - 512) * 0.01f;
    float y_clean = stepBiquad(notchFilter, raw_pos);

    // Estimación óptima de estados con Filtro de Kalman
    kalmanObserverStep(u_cmd, y_clean);

    // Ley de control híbrida: LQR (H2) + Modo Deslizante Suavizado (SMC)
    float e_pos = x_hat[0] - ref_pos;
    float e_vel = x_hat[1];
    float s_surf = e_vel + LAMBDA_SMC * e_pos;

    float u_lqr = -K_LQR_POS * e_pos - K_LQR_VEL * e_vel;
    float u_smc = -K_SW_SMC * satBoundary(s_surf, PHI_BOUND);
    u_cmd = constrain(u_lqr + u_smc + (x_hat[2] / 18.5f), -12.0f, 12.0f);

    int pwmOut = constrain((int)(fabsf(u_cmd) * (255.0f / 12.0f)), 0, 255);
    digitalWrite(10, (u_cmd >= 0.0f) ? HIGH : LOW);
    analogWrite(9, pwmOut);
  }
}
```

5. Simulación Python: Espacio de Estados, Riccati LQR/H2 y Observador de Kalman (simulacion_sistemas_y_control_video_01.py)

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
================================================================================
CURSO: SISTEMAS Y CONTROL EN MECATRÓNICA (GEMINI — cursosmecatronica.com)
VIDEO 01: Video 01 — El Rol del Control, el Modelado y Señales y Sistemas en Mecatrónica
Simulación de Espacio de Estados, LQR/H2, Filtro de Kalman, Eje Electrohidráulico
y Optimización Multiobjetivo de Pareto.
================================================================================
"""
import math

TS = 0.0010
WN = 50
ZETA = 0.7
Q_KALMAN = 0.0001
R_KALMAN = 0.005

def simulate_state_space_lqr_kalman(steps=400, ref=1.0):
    # Estado real [pos, vel] y estado estimado por Filtro de Kalman
    x = [0.0, 0.0]
    x_hat = [0.0, 0.0]
    p00, p01, p11 = 1.0, 0.0, 1.0

    # Ganancias LQR óptimas (síntesis H2)
```

```
    k1 = WN * 0.55
    k2 = 2.0 * ZETA * math.sqrt(WN)
    history = []

    for k in range(steps):
        y_meas = x[0]  # Medición del sensor
        # Actualización de Kalman
        innov = y_meas - x_hat[0]
        s_cov = p00 + R_KALMAN
        k_kal0 = p00 / s_cov
        k_kal1 = p01 / s_cov
        x_hat[0] += k_kal0 * innov
        x_hat[1] += k_kal1 * innov

        # Ley de control LQR + compensación no lineal de fricción
        e0 = x_hat[0] - ref
        e1 = x_hat[1]
        u = max(-24.0, min(24.0, -k1 * e0 - k2 * e1))

        # Integración de la planta mecatrónica / electrohidráulica
        acc = -2.0 * ZETA * WN * x[1] - (WN ** 2) * (x[0] - ref) * 0.05 + 15.0 * u
        x[0] += TS * x[1]
        x[1] += TS * acc

        # Predicción de Kalman
        x_hat[0] += TS * x_hat[1]
        x_hat[1] += TS * (-2.0 * ZETA * WN * x_hat[1] + 15.0 * u)
        p00 += TS * (2.0 * p01 + TS * p11) + Q_KALMAN
        p01 += TS * p11
        p11 += Q_KALMAN * 5.0

        if k % 50 == 0:
            history.append((round(k * TS, 4), round(x[0], 4), round(x_hat[0], 4), round(u, 3)))
    return history

if __name__ == "__main__":
    print("=== Simulación Sistemas y Control en Mecatrónica - Video 01 ===")
    for t_s, pos, pos_est, u_v in simulate_state_space_lqr_kalman():
        print(f"t={t_s:6.3f}s | Pos={pos:7.4f} | Kalman={pos_est:7.4f} | u={u_v:7.3f}V")
```

6. Programa PLC IEC 61131-3 (ST): Control Avanzado de Eje Electrohidráulico (plc_eje_electrohidraulico_control_video_01.st)

```
(* ================================================================================
 * CURSO: SISTEMAS Y CONTROL EN MECATRÓNICA (GEMINI — cursosmecatronica.com)
 * VIDEO 01: Video 01 — El Rol del Control, el Modelado y Señales y Sistemas en Mecatrónica
 * Norma IEC 61131-3 (Structured Text): Observador de Estado Discreto,
 * Control en Cascada de Eje Electrohidráulico con Amortiguamiento Activo de Presión
 * ================================================================================ *)
PROGRAM PLC_Sistemas_Y_Control_V01
VAR_INPUT
    rRefPos_mm      : REAL := 100.0;   (* Referencia de posición [mm] *)
```

```
    rMeasPos_mm     : REAL := 0.0;     (* Sensor magnetoestrictivo / encoder [mm] *)
    rPresionA_bar   : REAL := 70.0;    (* Presión cámara A [bar] *)
    rPresionB_bar   : REAL := 70.0;    (* Presión cámara B [bar] *)
    bEnableAxis     : BOOL := TRUE;    (* Habilitación de seguridad *)
END_VAR
VAR_OUTPUT
    rValveCmd_V     : REAL := 0.0;     (* Comando a servoválvula [-10..+10 V] *)
    rPosEstimada_mm : REAL := 0.0;     (* Posición filtrada por observador *)
    rVelEstimada_mms: REAL := 0.0;     (* Velocidad estimada por Kalman [mm/s] *)
END_VAR
VAR
    rTs             : REAL := 0.0010;
    rErrorPos       : REAL := 0.0;
    rPresionCarga   : REAL := 0.0;
    rPresionAnt     : REAL := 0.0;
    rDerivPresion   : REAL := 0.0;
    rInnov          : REAL := 0.0;
    rKp_Cascada     : REAL := 4.50;
    rKv_Cascada     : REAL := 0.18;
    rKd_Presion     : REAL := 0.035;   (* Amortiguamiento hidromecánico activo *)
END_VAR

IF NOT bEnableAxis THEN
    rValveCmd_V := 0.0;
    RETURN;
END_IF;

(* 1. Observador de estado alfa-beta (Kalman estacionario) *)
rInnov := rMeasPos_mm - rPosEstimada_mm;
rPosEstimada_mm := rPosEstimada_mm + rTs * rVelEstimada_mms + 0.35 * rInnov;
rVelEstimada_mms := rVelEstimada_mms + (0.06 / rTs) * rInnov;

(* 2. Retroalimentación activa de presión de carga PL = PA - PB *)
rPresionCarga := rPresionA_bar - rPresionB_bar;
rDerivPresion := (rPresionCarga - rPresionAnt) / rTs;
rPresionAnt   := rPresionCarga;

(* 3. Ley de control avanzada con amortiguamiento activo e inversión *)
rErrorPos := rRefPos_mm - rPosEstimada_mm;
rValveCmd_V := (rKp_Cascada * rErrorPos) - (rKv_Cascada * rVelEstimada_mms) - (rKd_Presion *
rDerivPresion);

(* Saturación física de servoválvula proporcional [-10V, +10V] *)
IF rValveCmd_V > 10.0 THEN
    rValveCmd_V := 10.0;
ELSIF rValveCmd_V < -10.0 THEN
    rValveCmd_V := -10.0;
END_IF;

END_PROGRAM
```