Curso de Sistemas y Control en Mecatrónica — Guía de Estudio Video 06

Módulo del Curso: Módulo 6 — Redes Neuronales, Sistemas Difusos, Control de Eje Electrohidráulico y Optimización
Enfoque de Ingeniería: Lógica Difusa Mamdani/Takagi-Sugeno, Neuro-Control RBF y ANFIS, Control No Lineal de Servoválvula y Cilindro Electrohidráulico, y Co-Diseño Multiobjetivo Pareto
Plataforma Oficial: `cursosmecatronica.com`

```
Q_L = C_d w x_v √((P_s - sgn(x_v)P_L)/ρ) | (V_0/(4β_e)) dP_L/dt = Q_L - A_p dx_p/dt - C_tp P_L | min_p [J_1(p), J_2(p), J_3(p)] s.t. g(p) ≤ 0
```

1. Síntesis de Conceptos Clave y Reglas de Diseño

- Sistemas de Lógica Difusa (inferencia Mamdani y Takagi-Sugeno), superficies de control no lineal y auto-sintonía difusa en línea de ganancias PID.
- Redes Neuronales Artificiales (MLP, RBF y modelos NARX), compensación neuronal adaptativa estable por Lyapunov y arquitectura híbrida Neuro-Difusa ANFIS de 5 capas.
- Modelado físico y control avanzado de un eje electrohidráulico: ecuación de caudal por orificio de Bernoulli, compresibilidad volumétrica βe, resonancia hidráulica poco amortiguada, retroalimentación activa de presión de carga PL, inversión de válvula y Backstepping/SMC.
- Optimización del diseño de sistemas mecatrónicos: co-diseño concurrente planta-controlador, frontera multiobjetivo de Pareto y algoritmos SQP, NSGA-II y PSO.

2. Parámetros Nominales de Sistemas y Control (Video 06)

Parámetro de Ingeniería	Símbolo	Valor Nominal	Función en el Sistema de Control Mecatrónico
Periodo de Muestreo Digital	`Ts [ms]`	0.5 ms	Intervalo determinista de discretización ZOH y ejecución de Kalman/LQR

Parámetro de Ingeniería	Símbolo	Valor Nominal	Función en el Sistema de Control Mecatrónico
Frecuencia Natural y Amortiguamiento	ω_n / ζ	180 rad/s / 0.7	Ubicación de polos dominantes en lazo cerrado
Márgenes de Fase y Ganancia	PM / GM	58° / 12 dB	Márgenes de estabilidad robusta en respuesta en frecuencia
Covarianzas del Filtro de Kalman	Q / R	Q=0.00001, R=0.0004	Matrices de incertidumbre de proceso y de ruido de sensores

3. Desarrollo Académico Diapositiva por Diapositiva

Diapositiva 01: Sistemas de Lógica Difusa (Fuzzy Systems): Conjuntos, Inferencia Mamdani y Takagi-Sugeno

Variables Lingüísticas, Funciones de Pertenencia Triangulares/Gaussianas, Base de Reglas IF-THEN y Defuzzificación

Turno 1 (Masculina): Introducida por Lotfi Zadeh, la Lógica Difusa permite incorporar conocimiento experto humano y reglas heurísticas de operación directamente dentro de un controlador matemático sin requerir un modelo diferencial exacto.

Turno 2 (Femenina): En la etapa de Fuzzificación, variables continuas precisas como el error y la derivada del error se convierten en grados de pertenencia entre cero y uno sobre conjuntos lingüísticos como Negativo Grande, Cero o Positivo Pequeño.

Turno 3 (Masculina): El motor de inferencia evalúa en paralelo una base de reglas condicionales Si-Entonces; en el método de Mamdani el consecuente es otro conjunto difuso que luego se convierte en un comando numérico mediante el centro de gravedad.

Turno 4 (Femenina): Por su parte, en los sistemas difusos de Takagi-Sugeno el consecuente de cada regla es un modelo lineal local o una ley de control en espacio de estados, permitiendo demostrar estabilidad global mediante desigualdades matriciales lineales.

Diapositiva 02: Controladores Difusos en Mecatrónica y Auto-Sintonía Difusa de Ganancias PID

Superficie No Lineal de Control PD/PID Difuso y Adaptación en Tiempo Real de Kp, Ki y Kd según el Punto de Operación

Turno 1 (Masculina): Un controlador difuso con entradas de error y cambio de error genera una superficie de control tridimensional intrínsecamente no lineal, cuya ganancia equivalente varía suavemente según la región del espacio de fase.

Turno 2 (Femenina): Así, podemos programar una pendiente suave cerca del origen para evitar amplificar ruido o excitar el juego mecánico, y una ganancia muy enérgica lejos de la referencia para lograr tiempos de subida ultrarrápidos.

Turno 3 (Masculina): En la industria es sumamente popular la arquitectura híbrida de P I D auto-sintonizado por lógica difusa, donde el lazo interno de alta velocidad sigue siendo un P I D digital clásico y determinista...

Turno 4 (Femenina): ...mientras un supervisor difuso ajusta continuamente en línea los incrementos de las ganancias proporcional, integral y derivativa para compensar cambios de carga o fricción sin riesgo de inestabilidad.

Diapositiva 03: Redes Neuronales Artificiales (MLP y RBF) en Identificación y Control Mecatrónico

Teorema de Aproximación Universal, Modelos NARX de Planta, Control por Modelo Inverso y Compensación Neuronal en Línea

Turno 1 (Masculina): Las Redes Neuronales Artificiales, en particular el Perceptrón Multicapa M L P y las redes de Funciones de Base Radial R B F, gozan de la propiedad matemática de aproximación universal sobre cualquier función continua no lineal.

Turno 2 (Femenina): En identificación de sistemas mecatrónicos, entrenamos redes neuronales autorregresivas no lineales con entradas exógenas, conocidas como N A R X, para capturar dinámicas complejas de actuadores térmicos, neumáticos o magnéticos.

Turno 3 (Masculina): En control de movimiento, la arquitectura más robusta combina un controlador lineal o de par calculado nominal en paralelo con una red neuronal de base radial cuyos pesos de salida se actualizan en tiempo real.

Turno 4 (Femenina): Derivando la ley de actualización de pesos directamente de una función de Lyapunov del error de seguimiento, la red neuronal aprende y cancela al vuelo las perturbaciones e incertidumbres no modeladas con estabilidad garantizada.

Diapositiva 04: Sistemas Neuro-Difusos Adaptativos (ANFIS): Sinergia de Interpretabilidad y Aprendizaje

Arquitectura de 5 Capas Equivalente a Takagi-Sugeno y Entrenamiento Híbrido (Gradiente Descendente + Mínimos Cuadrados)

Turno 1 (Masculina): Mientras la lógica difusa ofrece reglas transparentes e interpretables pero carece de ajuste automático, y las redes neuronales aprenden de los datos pero operan

como cajas negras, los sistemas Neuro-Difusos unen lo mejor de ambos mundos.

Turno 2 (Femenina): La arquitectura A N F I S, Sistema de Inferencia Difusa Basado en Redes Adaptativas, representa un sistema difuso de Takagi-Sugeno como una red prealimentada estructurada en cinco capas funcionales.

Turno 3 (Masculina): La primera capa contiene los parámetros no lineales de las funciones de pertenencia de entrada, las capas intermedias evalúan y normalizan el grado de disparo de cada regla, y la cuarta capa calcula los polinomios lineales de salida.

Turno 4 (Femenina): Gracias a esta estructura, A N F I S emplea un algoritmo de aprendizaje híbrido sumamente veloz: mínimos cuadrados en el paso hacia adelante para optimizar los parámetros consecuentes y retropropagación del gradiente hacia atrás para afinar las funciones de pertenencia.

Diapositiva 05: Física y Modelado No Lineal de un Eje Electrohidráulico Industrial

Dinámica de la Servoválvula, Ecuación de Caudal por Orificio de Bernoulli y Continuidad de Presión con Módulo Volumétrico βe

Turno 1 (Masculina): Los ejes electrohidráulicos son insustituibles en prensas pesadas, simuladores de vuelo, maquinaria de construcción y bancos de fatiga estructural, pues entregan una densidad de fuerza por unidad de peso diez veces superior a los motores eléctricos.

Turno 2 (Femenina): Sin embargo, su modelado revela que un eje electrohidráulico es una de las plantas más severamente no lineales de la ingeniería mecatrónica, gobernada por el acoplamiento entre mecánica, electromagnetismo y dinámica de fluidos.

Turno 3 (Masculina): En primer lugar, el caudal volumétrico Q que atraviesa la corredera de la servoválvula no es proporcional a la apertura x sub v, sino que varía con la raíz cuadrada de la caída de presión a través del orificio según la ley de Bernoulli.

Turno 4 (Femenina): En segundo lugar, la derivada de la presión en las cámaras del cilindro depende del módulo de compresibilidad efectivo del aceite beta sub e dividido por el volumen instantáneo de cada cámara, el cual cambia continuamente con la posición del pistón.

Diapositiva 06: Desafíos de Control en Ejes Electrohidráulicos: Resonancia Hidráulica y Fricción

Frecuencia Natural Hidráulica ω_h, Bajo Amortiguamiento Intrínseco (ζ_h ≈ 0.05–0.15), Zona Muerta de Corredera y Fugas

Turno 1 (Masculina): La columna de aceite atrapada en las dos cámaras del cilindro actúa físicamente como un resorte hidromecánico acoplado a la masa de la carga, originando la frecuencia natural hidráulica omega sub h.

Turno 2 (Femenina): Dicha frecuencia alcanza su valor mínimo crítico cerca de la mitad de la carrera del cilindro y presenta un factor de amortiguamiento natural extremadamente bajo, a menudo inferior a cero punto uno debido a las bajas fugas laminares.

Turno 3 (Masculina): A esto se suma la asimetría de áreas en cilindros de simple vástago, la zona muerta por recubrimiento positivo del carrete de la válvula y la pronunciada fricción de Stribeck de los sellos de alta presión.

Turno 4 (Femenina): Como consecuencia, un controlador P I D convencional se ve forzado a operar con ganancias muy reducidas para no desestabilizar la resonancia hidráulica, sufriendo errores de seguimiento inaceptables.

Diapositiva 07: Control Avanzado de un Eje Electrohidráulico: Cascada, Presión Dinámica y Backstepping

Retroalimentación de Presión de Carga P_L (Amortiguamiento Activo), Inversión de Ecuación de Orificio y Control Adaptativo/SMC

Turno 1 (Masculina): El control avanzado de un eje electrohidráulico resuelve estas limitaciones combinando instrumentación multivariada con leyes de compensación no lineal y observadores en tiempo real.

Turno 2 (Femenina): Primero incorporamos transductores de presión en ambas cámaras para realimentar la derivada o el filtro pasaaltos de la presión de carga P sub L, inyectando amortiguamiento hidromecánico virtual sin disipar potencia en estrangulamientos físicos.

Turno 3 (Masculina): Segundo, implementamos en el controlador la inversión matemática de la ecuación de raíz cuadrada del orificio junto con una acción prealimentada o Feedforward de velocidad que calcula el caudal exacto requerido por la trayectoria.

Turno 4 (Femenina): Finalmente, cerramos el lazo exterior mediante Control Adaptativo Robusto o Backstepping con Modos Deslizantes, logrando seguimientos de posición submilimétricos y transiciones suaves entre control de posición y control de fuerza.

Diapositiva 08: Optimización del Diseño de Sistemas Mecatrónicos: Formulación Matemática

Vector de Variables de Diseño p (Mecánicas + Eléctricas + Control), Funciones Objetivo J(p) y Restricciones g_i(p) ≤ 0

Turno 1 (Masculina): Culminamos el curso abordando la Optimización del Diseño de Sistemas Mecatrónicos: el marco matemático que determina simultáneamente las dimensiones físicas de la máquina y los parámetros de su controlador.

Turno 2 (Femenina): Definimos el vector de variables de diseño p, el cual integra cotas geométricas como longitudes de eslabones y relaciones de transmisión, parámetros de actuadores y las ganancias del controlador.

Turno 3 (Masculina): Sobre este vector formulamos una o varias funciones objetivo a minimizar, tales como el tiempo de ciclo de una operación, la energía térmica disipada por efecto Joule en los devanados o el error cuadrático de seguimiento.

Turno 4 (Femenina): El problema queda acotado por restricciones de desigualdad e igualdad estrictas que garantizan la factibilidad física: esfuerzo mecánico menor al límite de fluencia, par pico inferior al máximo del motor y márgenes de estabilidad garantizados.

Diapositiva 09: Co-Diseño Planta-Controlador y Optimización Multiobjetivo: Frontera de Pareto

Superación del Diseño Secuencial Clásico, Compromiso Masa-Energía-Ancho de Banda y Selección sobre el Frente de Pareto

Turno 1 (Masculina): Estudios industriales demuestran que el co-diseño concurrente de planta y controlador alcanza desempeños entre un veinte y un cuarenta por ciento superiores al diseño secuencial tradicional donde la mecánica se fija de antemano.

Turno 2 (Femenina): Como en mecatrónica los objetivos suelen estar en conflicto, por ejemplo minimizar la masa total de un robot mientras se maximiza su rigidez estructural y su ancho de banda de control, recurrimos a la optimización multiobjetivo.

Turno 3 (Masculina): En lugar de una solución única, obtenemos el conjunto de soluciones no dominadas conocido como Frontera de Pareto, donde ningún objetivo puede mejorarse sin deteriorar al menos otro de los criterios en competencia.

Turno 4 (Femenina): La curva de Pareto entrega al equipo de ingeniería el mapa exacto de compromisos físicos del sistema, permitiendo elegir el diseño óptimo según el costo comercial, la vida útil o la precisión requerida.

Diapositiva 10: Algoritmos de Optimización Mecatrónica y Síntesis Final del Módulo 6

Programación Cuadrática Secuencial (SQP), Algoritmos Genéticos (NSGA-II), Enjambre de Partículas (PSO) y Validación

Turno 1 (Masculina): Para resolver numéricamente el problema de optimización mecatrónica empleamos dos familias de algoritmos complementarios en función de la suavidad y convexidad del espacio de búsqueda.

Turno 2 (Femenina): Cuando las funciones objetivo y restricciones son diferenciables, los métodos basados en gradiente como la Programación Cuadrática Secuencial S Q P y el método de Punto Interior convergen con extraordinaria rapidez al óptimo local.

Turno 3 (Masculina): Cuando el diseño involucra catálogos discretos de motores, simulaciones no lineales discontinuas o múltiples óptimos locales, aplicamos algoritmos metaheurísticos globales como el algoritmo genético multiobjetivo N S G A dos o el Enjambre de Partículas P S O.

Turno 4 (Femenina): Acompáñanos al video final de cierre para sintetizar todo el recorrido del Curso de Sistemas y Control en Mecatrónica.

4. Firmware C++/MCU: Filtro de Kalman, Biquad DSP, LQR (H2) y Modos Deslizantes (control_sistemas_mecatronica_video_06.ino)

```
/*
 * ========================================================================
 * CURSO: SISTEMAS Y CONTROL EN MECATRÓNICA (GEMINI — cursosmecatronica.com)
 * VIDEO 06: Video 06 — Redes Neuronales, Sistemas Difusos, Control de Eje Electrohidráulico y
Optimización Mecatrónica
 * Enfoque: Lógica Difusa Mamdani/Takagi-Sugeno, Neuro-Control RBF y ANFIS, Control No Lineal de
Servoválvula y Cilindro Electrohidráulico, y Co-Diseño Multiobjetivo Pareto
 * Ecuación Clave: Q_L = C_d w x_v √((P_s - sgn(x_v)P_L)/ρ)  |  (V_0/(4β_e)) dP_L/dt = Q_L - A_p dx_p/dt
- C_tp P_L  |  min_p [J_1(p), J_2(p), J_3(p)] s.t. g(p) ≤ 0
 * ========================================================================
 */
#include <Arduino.h>

// Parámetros de tiempo real y control en espacio de estados discreto
static const float TS_SEC = 0.0005f;
static const uint32_t TS_US = 500UL;

// 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 = 0.00001f;
static const float R_MEAS = 0.0004f;
```

```
// Ganancias óptimas LQR (H2) + Superficie de Modos Deslizantes (SMC)
static const float K_LQR_POS = 81.00f;
static const float K_LQR_VEL = 9.10f;
static const float LAMBDA_SMC = 63.00f;
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_06.py)

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
================================================================================
CURSO: SISTEMAS Y CONTROL EN MECATRÓNICA (GEMINI — cursosmecatronica.com)
VIDEO 06: Video 06 — Redes Neuronales, Sistemas Difusos, Control de Eje Electrohidráulico y Optimización Mecatrónica
Simulación de Espacio de Estados, LQR/H2, Filtro de Kalman, Eje Electrohidráulico
```

```
y Optimización Multiobjetivo de Pareto.
================================================================================
"""
import math

TS = 0.0005
WN = 180
ZETA = 0.7
Q_KALMAN = 0.00001
R_KALMAN = 0.0004

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 06 ===")
```

```
    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_06.st)

```
(* ================================================================================
 * CURSO: SISTEMAS Y CONTROL EN MECATRÓNICA (GEMINI — cursosmecatronica.com)
 * VIDEO 06: Video 06 — Redes Neuronales, Sistemas Difusos, Control de Eje Electrohidráulico y
Optimización 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_V06
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 := 0.0005;
    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
```