

# Informe de Laboratorio 1

Control Automático

1<sup>st</sup> Juan José Restrepo Rosero  
Pontificia Universidad Javeriana Cali  
Cali, Colombia  
juanjorestrepo@javerianacali.edu.co

2<sup>nd</sup> Isabella Ceballos Sánchez  
Pontificia Universidad Javeriana Cali  
Cali, Colombia  
isaceballos517@javerianacali.edu.co

3<sup>th</sup> Francisco Jose Pantoja Zapata  
Pontificia Universidad Javeriana Cali  
Cali, Colombia  
pantojazapata@javerianacali.edu.co

4<sup>th</sup> Alejandra Gómez Parra  
Pontificia Universidad Javeriana Cali  
Cali, Colombia  
alejandrag10@javerianacali.edu.co

## I. INTRODUCCIÓN

En esta práctica se estudiarán los efectos de las acciones de control relativas a la posición angular de un servomotor. Se identificarán los parámetros del modelo de la planta, se diseñará e implementará un controlador de acuerdo con las especificaciones dadas y posteriormente se evaluará el desempeño del sistema controlado.

## II. EQUIPOS E INSTRUMENTOS

- Quanser Qube Servo 2 con disco de inercia.
- Arduino UNO.
- Cables de conexión.
- Computador.

## III. DESARROLLO DEL LABORATORIO

### A. Estabilidad planta representación velocidad - voltaje

$$\frac{\Omega_m(s)}{V_m(s)} = \frac{K}{(\tau s + 1)} \quad (1)$$

A partir de la ecuación 1, se logra apreciar que es de primer orden, y que a su vez, esta cuenta con un polo en:

$$s = \frac{-1}{\tau} \quad (2)$$

Dado esto, se cuenta con un polo real negativo, permitiendo caracterizar a la planta como un sistema con respuesta estable.

### B. Estabilidad planta representación posición - voltaje

$$\frac{\Theta_m(s)}{V_m(s)} = \frac{K}{s(\tau s + 1)} \quad (3)$$

A partir de la ecuación 3, se aprecia que en este caso se presenta una planta de segundo orden, y por ende, se contará con dos polos en:

$$s = \frac{-1}{\tau} \quad (4)$$

$$s = 0 \quad (5)$$

Por ende, aunque el primer polo es real y con parte negativa, se cuenta con un segundo polo igual a cero, y esto provoca que la planta tenga un comportamiento **marginalmente estable**.

### C. Pasos para determinar la ganancia de estado estacionario y la constante de tiempo

Para determinar la ganancia de estado estacionario y la constante de tiempo, de forma experimental, se realizaron los siguientes pasos:

- Acudiendo a arduino, se toman las muestras de velocidad según el voltaje aplicado, este voltaje se varió de 0 a 5 V.
- A partir de los datos obtenidos, se exportaron a la herramienta excel.
- Se realiza una gráfica por cada cambio de voltaje.
- A partir de la gráfica generada, se mide el tiempo de establecimiento de la gráfica para determinar la ganancia de estado estacionario.
- A partir de la gráfica generada y conociendo la ganancia, se mide el tiempo en el que se toma en llegar al 65% de la ganancia.
- A partir de los resultados, se realiza un promedio, para determinar la ganancia y constante de tiempo en promedio.

Después de haber realizado los pasos anteriores, los resultados obtenidos fueron los siguientes:

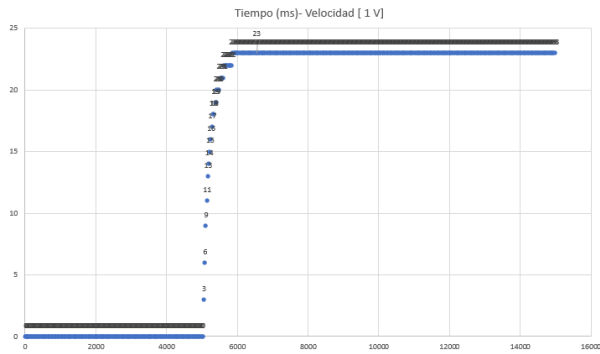


Fig. 1: Diagrama de datos de 0 a 1 V

K	23
Tao	0.199 s

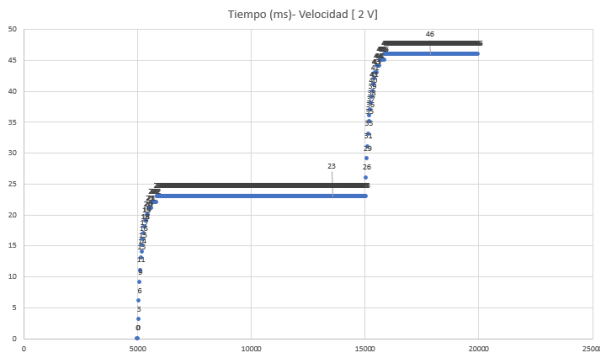


Fig. 2: Diagrama de datos de 1 a 2 V

K	23
Tao	0.091 s

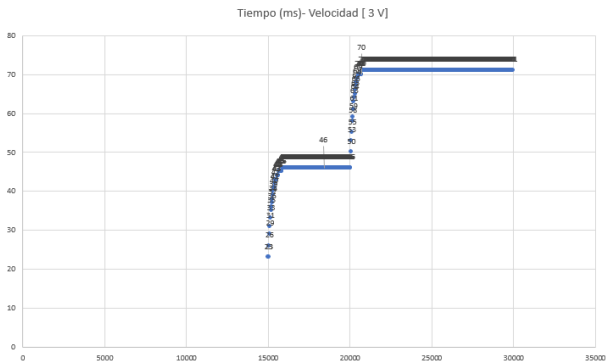


Fig. 3: Diagrama de datos de 2 a 3 V

K	24
Tao	0.06 s

A partir de esto, el promedio obtenido de las variables es:

K	23.3
Tao	0.12 s

#### D. Uso de simulink

Este punto se desarrolló partir de la función transferencia de posición con respecto al voltaje:

$$\frac{\theta m(s)}{Vm(s)} = \frac{23.3}{0.12s^2 + s} \quad (6)$$

1)  $Kd = 0$ ,  $Kp$  entre 1 y 4 en pasos de 0.5: Acudiendo a matlab, se desarrolló un bloque correspondiente al control PV con las variaciones de kp:

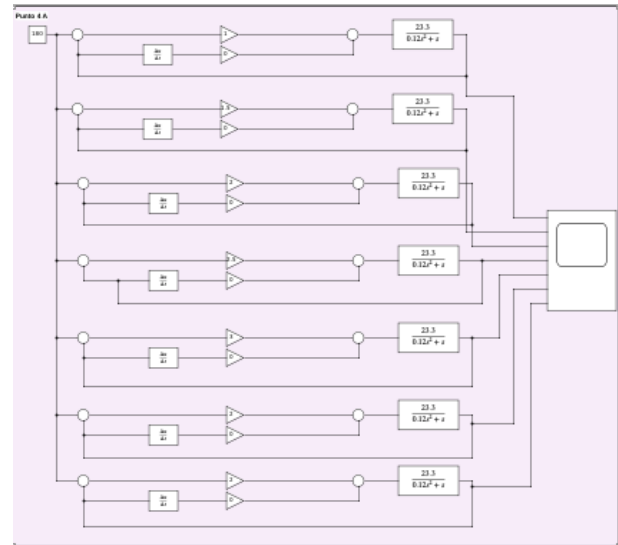


Fig. 4: Bloque punto 4 A

Al correr con un tiempo de 2 segundos, se obtuvo:

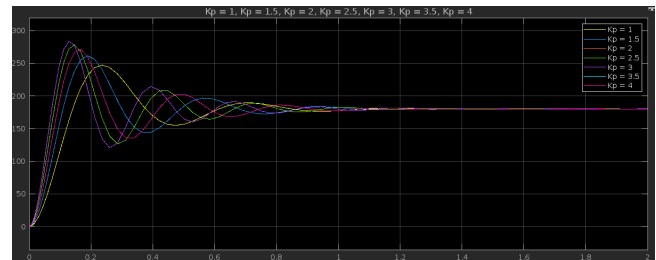


Fig. 5: Bloque punto 4 A resultados

En la figura 5, se puede observar que a medida que aumenta el valor de  $Kp$  se presentan más oscilaciones. Esto se debe a que el controlador  $Kp$  (proporcional) determina la relación entre la salida del sistema y el error, lo que implica que, al incrementar el  $k_p$ , aumenta la velocidad de estabilización del sistema; pero sin embargo, se ve afectado ya que cada vez presenta más oscilaciones antes de lograr el tiempo de estabilización.

2)  $K_p = 2.5$ ,  $K_d$  entre 0 y 0.15 en pasos de 0.03:  
Del mismo modo, se planteó un bloque de simulación que permitiera variar el valor de kd:

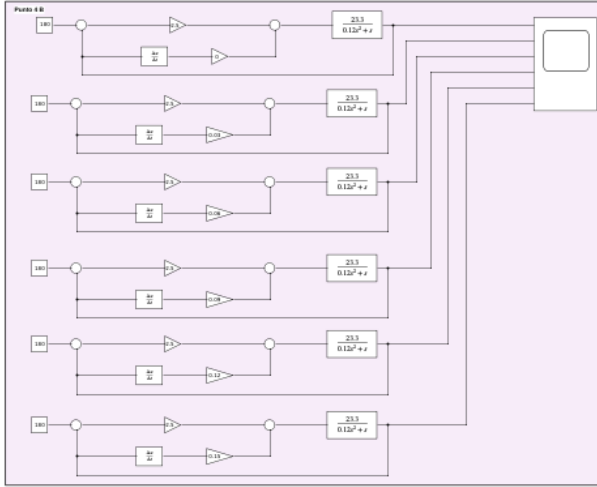


Fig. 6: Bloque punto 4 B

Al correr con un tiempo de 2 segundos, se obtuvo:

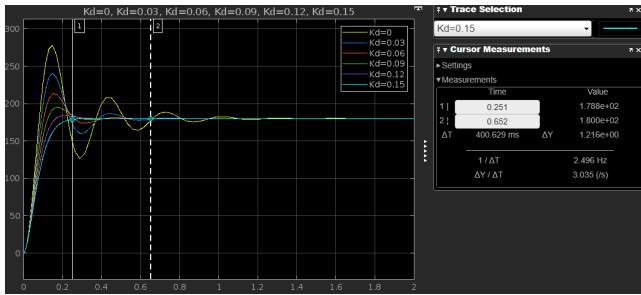


Fig. 7: Bloque punto 4 B resultados

En la figura 7, se concluye que, a medida que aumenta la constante derivativa ( $K_d$ ), el motor toma menos tiempo en estabilizarse, esto se presenta dado que la ganancia  $K_d$ , evita el sobre pico y además, entre mayor sea el valor de  $K_d$ , las oscilaciones que se presentarán serán menor.

### E. Filtro pasa bajos

El filtro pasa-bajos planteado en el laboratorio, está dado por la siguiente ecuación:

$$\frac{100}{s + 100} \quad (7)$$

A partir de esto, se tomaron los mismos bloques desarrollados en el punto 4, y se añadió el filtro pasa-bajas. Dado esto, el resultado obtenido fue el siguiente:

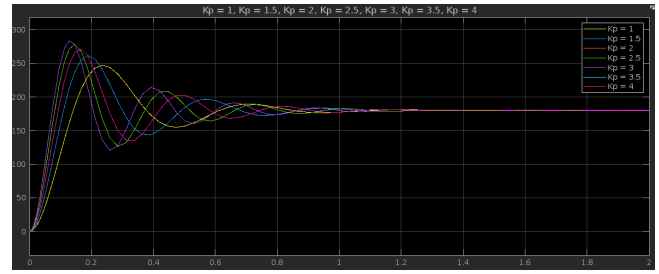


Fig. 8: Resultados al añadir filtro pasa-bajo al variar  $K_p$

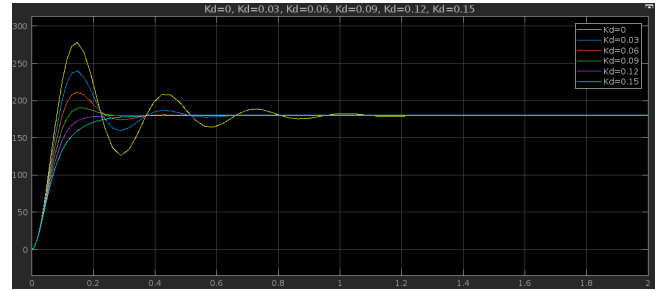


Fig. 9: Resultados al añadir filtro pasa-bajo al variar  $K_d$

Al comparar los resultados con el punto 4, se puede apreciar que al generar la implementación del filtro, este no provoca un gran cambio en el resultado de la salida. Esto nos hace pensar, que el polo que añade el filtro pasa-baja, puede estar ubicado en una zona muy lejana, logrando que su efecto en la planta sea el mínimo.

### F. Exploración y entendimiento programa Arduino

En este punto, se estudió y revisó el código brindado, de tal manera que, se consiguió un buen entendimiento de éste.

### G. Planteamiento del control para posición angular

Para el desarrollo del control, se decidió diseñar un controlador a partir del control PV, esto dado que este tipo de control tiene un gran uso para las aplicaciones en la industria. Además, se quiso probar con un nuevo tipo de control, teniendo en cuenta que su funcionamiento se basa en un control aprendido anteriormente.

Teniendo en cuenta esto, el tiempo de establecimiento y máximo sobre impulso tomados para el diseño fueron:

$$t_s = 900 \text{ ms} \quad M_p = 10\% \quad (8)$$

Primero, se decidió sacar la función de transferencia en lazo cerrado del controlador:

$$\frac{\frac{23.3(K_p + K_d s)}{0.12s^2 + s}}{1 + \frac{23.3(K_p + K_d s)}{0.12s^2 + s}} = \frac{23.3(K_p + K_d s)}{0.12s^2 + s + 23.3K_p + 23.3K_d s} \quad (9)$$

A partir de esto, la ecuación de control obtenida fue:

$$\begin{aligned} \text{Control} &= 0.12s^2 + s(1 + 23.3K_d) + 23.3K_p \\ \text{Control} &= s^2 + (8.33 + 194.16K_d) + 194.16K_p \end{aligned} \quad (10)$$

Para generar la ecuación deseada, se partió de los parámetros de diseño escogidos:

$$\xi = \sqrt{\frac{\ln(M_p)^2}{\pi^2 + \ln(M_p)^2}} = 0.59 \quad (11)$$

$$W_n = \frac{4}{t_s \xi} = 7.53 \quad (12)$$

Dado esto, la ecuación deseada es:

$$s^2 + 8.8854s + 56.7 \quad (13)$$

Al igualar los términos:

$$\begin{aligned} 8.33 + 194.16K_d &= 8.8854 \\ K_d &= \frac{8.8854 - 8.33}{194.16} = 0.00287 \end{aligned} \quad (14)$$

$$\begin{aligned} 194.16K_p &= 56.7 \\ K_p &= \frac{56.7}{194.16} = 0.292 \end{aligned} \quad (15)$$

Dado esto, se procedió a simular el bloque de control PV en matlab, y así poder comprobar el diseño. Para esta simulación, se simuló tanto con el filtro pasa-bajas, como sin el filtro.

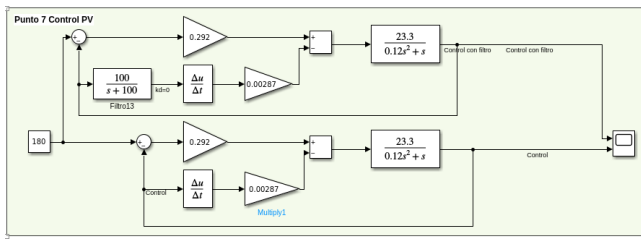


Fig. 10: Bloque de simulación control PV

Al correr la simulación, se obtuvo:

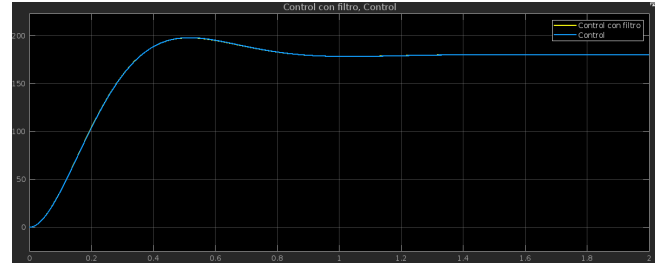


Fig. 11: Respuesta al control PV con y sin filtro

En la figura 12, se logra apreciar que se cumple con el tiempo de establecimiento y máximo sobre impulso especificados. Del mismo modo, se concluye que el efecto del filtro pasa-bajas es mínimo frente a la respuesta obtenida.

Como el control debe de implementarse en arduino, es necesario establecer un tiempo de muestreo, además, de la representación del control en ecuación de diferencias.

Para el tiempo de muestreo, se tuvo en cuenta el valor de tau, obteniendo:

$$t_{\text{muestreo}} = \frac{0.12}{10} = 0.012 \quad (16)$$

Acudiendo a la función de discretización de matlab, se obtuvo:

$$\begin{aligned} \text{SYSD} &= \\ &\frac{0.0003529 \ z^2 + 4.06e-05 \ z - 0.0003123}{z^2 - 1.989 \ z + 0.9894} \end{aligned}$$

Fig. 12: Discretización por método de tustin

Después de realizar el proceso de discretización, la función en diferencias obtenida se aprecia en la ecuación número 18:

$$\frac{U(z)}{E(z)} = \frac{0.0003529 + 4.06 \times 10^{-5}z - 0.0003123z^{-2}}{1 - 1.989z^{-1} + 0.9894z^{-2}} \quad (17)$$

$$\begin{aligned} u(k) &= 3.529 \times 10^{-4}e(k) + 4.06 \times 10^{-5}e(k-1) \\ &\quad - 3.123 \times 10^{-4}e(k-2) + 1.989u(k-1) - 0.9894u(k-2) \end{aligned} \quad (18)$$

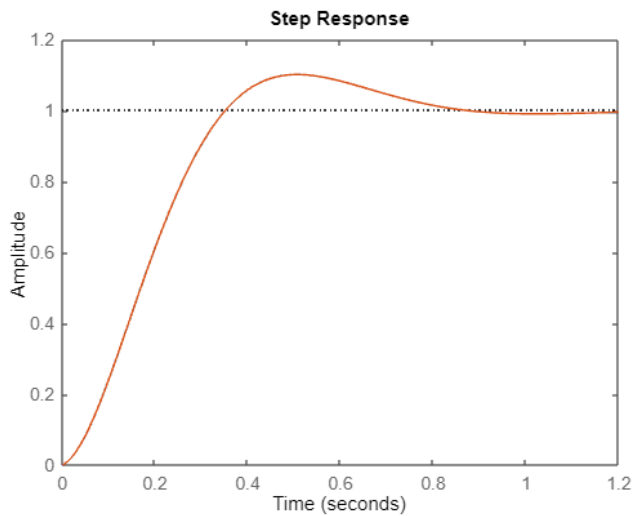


Fig. 13: Respuesta escalón discreto y continuo

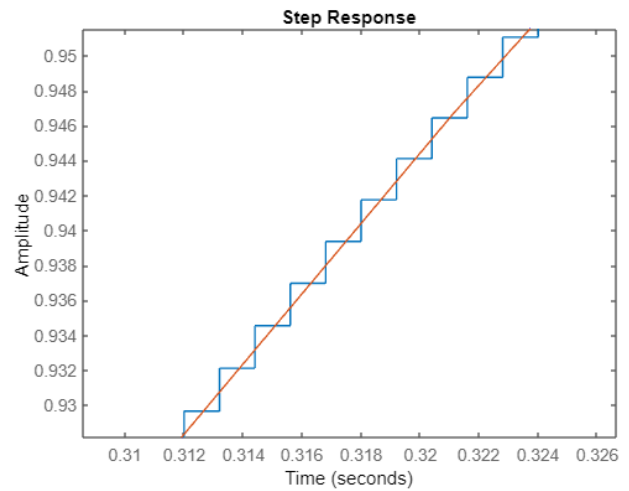


Fig. 15: Respuesta escalón discreto y continuo tiempo de transición

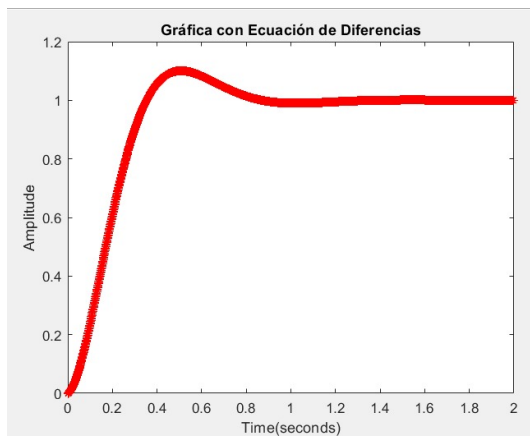


Fig. 14: Respuesta escalón discreto y continuo con ecuación de diferencias

En las figuras 13 y 14 se puede ver la respuesta a un escalón unitario del sistema de control en tiempo discreto, representado por la gráfica de color rojo y el sistema de control en tiempo continuo, representado por el color azul.

A simple vista, en la figura 13, es muy difícil distinguir las diferencias que existen entre el sistema en tiempo discreto y en tiempo continuo. Por ende, se procedió a realizar un zoom y así apreciar los cambios.

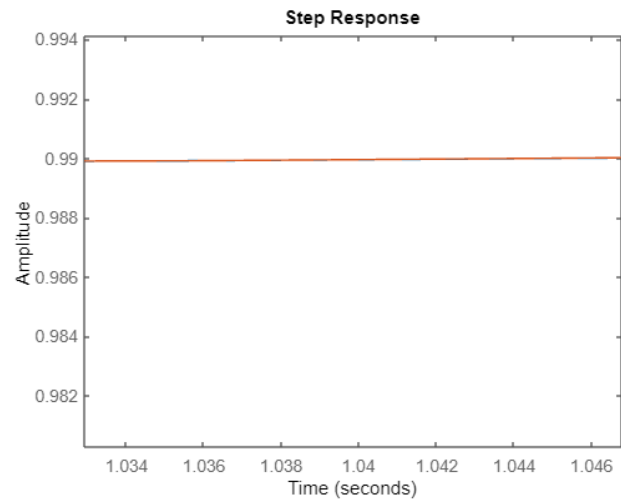


Fig. 16: Respuesta escalón discreto y continuo tiempo final

En la figuras 15 y 16, se puede ver como el sistema discretizado tiene diferencias con el sistema continuo, especialmente en la sección de transición, donde el error de cuantización es más notorio. Por otro lado, conforme avanza el tiempo este error de cuantización se va haciendo menor cada vez, donde al final ambos controles tienen el mismo comportamiento.

#### IV. ANEXOS

- [Muestras de datos excel](#)
- [Cálculos teóricos](#)
- [Simulaciones Matlab](#)