Control de posición de una masa con husillo de avance mediante LabVIEW

Caceres Pinzon Brian Sebastian, Gaviria Didier, Lizarazo Nicolas Esteban.

*{u1803245, u1803250 y u1802999}@unimilitar.edu.co*

Profesor: Adriana Riveros.

Como se aprecia en la Figura 1 cada uno de los métodos que se ven a continuación cumplen con un función en específico y

***Resumen*— En esta práctica de laboratorio se busca diseñar reguladores en tiempo continuo y en tiempo discreto para controlar la posición de un sistema masa con husillo de avance para después hacer implementación en LabVIEW .**

***Palabras clave*—reguladores, continuo, discreto, LabVIEW.**

* 1. INTRODUCCIÓN

E

N la presente practica se mostrar el diseño de reguladores para un sistema trabajado en prácticas anteriores el cual es sistema masa con husillo, tanto en tiempo continuo como en tiempo discreto que permitan controlar la posición ante referencia escalón y rampa para después discretizarlos y hacer

una implementación por medio del software de LabVIEW.

tiene cada uno un reemplazo de S distinto, el primer método es el denominado euler en adelanto, el segundo euler en atraso, el tercero llamado tustin.

* ¿Cómo se selecciona el tiempo de muestreo? Presentar 2 métodos diferentes y sus ejemplos respectivos (sistemas reales)

El tiempo de muestreo tiene una regla muy importante

al momento de realizar la selección del tiempo es la regla de NyQuist donde la frecuencia de muestreo fs debe ser dos veces mayor a la frecuencia máxima del sistema, también se tiene que tener en cuenta que el ancho de banda debe ser de 6 a 25 veces el tiempo de muestreo en lazo cerrado.

𝑓𝑠 = 2𝑓𝑚𝑎𝑥

1. método de selección tiempo de muestreo sistemas primer orden.
   1. TRABAJO PREVIO
   * ¿Cuántos métodos de discretización se encuentran en la literatura? Dar ejemplo de cada uno de ellos (por lo menos 4),aplicado a sistemas reales.

los métodos de discretización son una transformación de convergencia entre S y z , para esto existen diferentes tipos de tipos como lo son tustin,ZOH,FOH,matchet y muchas más que en la teoría vista en clase son las más comunes y las cuales se consideraron como método de estudio, a continuación una imagen que resume tres tipos de discretizaciones y algunas características de cada una

𝑡

4 < 𝑡𝑚 < 𝑟(1)

Ejemplo: diseñado PID, Ess=0:

3

𝐹(𝑠) =

𝑠 + 2

llevando la función a su forma estándar para encontrar el valor

de τ y posterior a ello los intervalos de tiempo de muestreo.

1

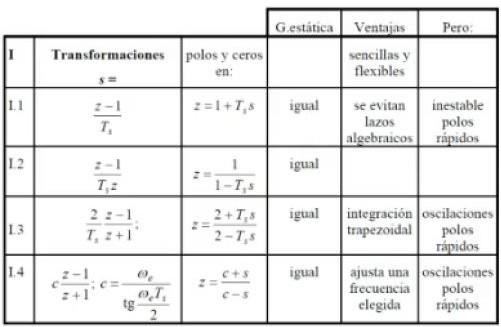
𝑟 = 2 = 0.5

el intervalo de tiempo de muestreo para el ejemplo debe ser reemplazando en la ecuación (1) y escogiendo tm = 0.25 que está dentro del rango

0.5

4 < 𝑡𝑚 < 0.5 = 0.125 < 𝑡𝑚 < 0.5

1. Selección de tiempo de muestreo sistema de segundo orden para los sistemas de segundo orden ası como en primero orden, tiene un tiempo de muestreo que está dado por la multiplicación de la frecuencia natural de la ecuación que describe la forma de la ecuación de segundo orden wn por un tiempo de muestreo y un rango de tiempo de muestreo como se observa a continuación.

0.25 < 𝑤𝑛 ∗ 𝑡𝑚 < 1.5

al ser una función de transferencia de segundo orden se tiene también un paramento importante para el cual los rangos de tiempo de muestreo estipulados se cumplen siempre y cuando

0.7 < 𝜁 < 1

Ejemplo:

3

𝐹(𝑠) = 𝑠2 + 4𝑠 + 4

* + ¿Que es un retenedor de orden cero?

Un retenedor de datos se conoce como retenedor de orden cero, o sujetador generador de la señal de escalera, el circuito retenedor de orden cero suaviza la señal muestreada para

asumiendo un tsd = 2 y un = 0.8 realizando el cálculo

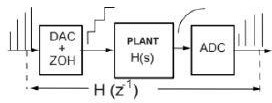
que se hace para hallar la frecuencia natural de la planta se encuentra que wn = 2.63 entonces el tm es:

0.25 < 2𝑡𝑚 < 1.5 = 0.125 < 𝑡𝑚 < 0.75

teniendo en cuenta los rangos en los que debe estar el tiempo de muestreo se escoge un tiempo de muestreo de 0.5 lo cual está dentro de los rangos que cumplen la regla, adicionalmente también se guiaran por el libro guía [1] que referencia una serie de tiempos de muestreo dependiendo el tipo de sistema

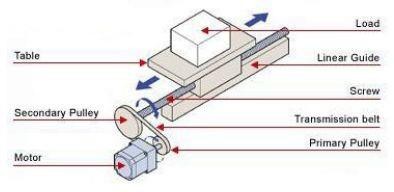
a tener en cuenta para diseños a futuro.

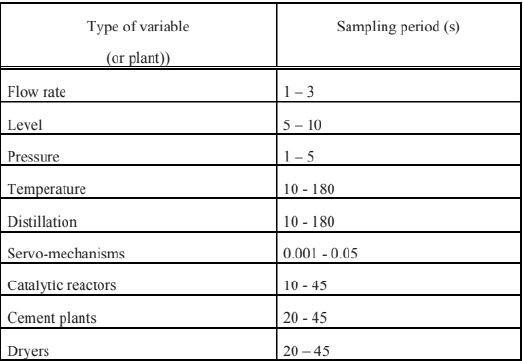
producir la señal h(t), la cual es constante desde el último valor muestreado hasta que se pueda disponer de la siguiente muestra.



en ese orden de ideas lo que es un retenedor de orden cero es un método de muestreo el cual su tiempo de duración es mucho más rápido que el de la planta, lo cual en un sistema embebido es la forma de conversión de una señal continua discreta mediante un tren de pulsos.

* 1. DESARROLLO DE LA PRÁCTICA



* + ¿Qué es una ecuación en diferencias? ¿Como se programa en un sistema de procesado tipo tarjetas de adquisición o microcontroladores?

Las ecuaciones en diferencia es una expresión del tipo:

𝐺(𝑛, 𝑓 (𝑛), 𝑓 (𝑛 + 1), 𝑓 (𝑛 + 2), . . . , 𝑓 (𝑛 + 𝑘)) = 0

Donde f es una función definida en Z.

El orden de una ecuación en diferencia está definido por

𝐾 = 𝐾𝑀𝐴𝑋 − 𝐾𝑀𝐼𝑁

Se puede decir que la ecuación en diferencia de orden K es lineal si se puede expresar de la siguiente forma:

𝑝0(𝑛) 𝑓 (𝑛 + 𝑘) + 𝑝1(𝑛) 𝑓 (𝑛 + 𝑘

− 1) + . . . + 𝑝𝑘 (𝑛)𝑓 (𝑛) = 𝑔(𝑛)

Para la implementación de una ecuación en diferencias en un microcontrolador se debe de tomar en cuenta que:

(𝑛 − 1) = 𝑣𝑎𝑙𝑜𝑟𝑎𝑛𝑡𝑒𝑟𝑖𝑜𝑟 (𝑛 + 1)

= 𝑣𝑎𝑙𝑜𝑟𝑠𝑖𝑔𝑢𝑖𝑒𝑛𝑡𝑒

Aunque el valor siguiente se puede estimar es preferible dejar la ecuación en terminar de 𝑍−1

Planta

Figura1:diagrama del sistema

0.5

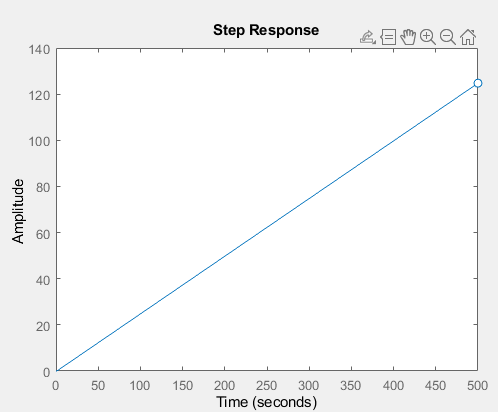
𝑠3 + 2𝑠2 + 2𝑠

Figura2: comportamiento de la planta

igualando la función anterior a U( z−1 ) / E( z−1 ):

## Planta

**Control continuo-discreto**

Diseñamos un PID para el sistema

𝑃𝐼𝐷 = 𝐾𝑑𝑠 + 𝐾𝑝 +

𝐾𝑖

𝑠

𝑈(𝑧−1)

𝐸(𝑧−1)

0.01298 + 0.03894𝑧−1 + 0.03894𝑧−2 − 0.01298𝑧−3

= 1 − 1.708𝑧−1 + 0.9692𝑧−2 − 0.2615𝑧−3

## Control

Valor de las constantes para el PID son las siguientes:

𝐾𝑑2 = 0.0608 𝐾𝑑 = −1.3527 𝐾𝑝 = 0.6040

𝐾𝑖 = 0.0506

Teniendo los valores de las constantes del PID se procede

𝑈(𝑧−1)

𝐸(𝑧−1) =

−2.552 + 2.971𝑧−1 + 4.357𝑧−2 − 4.625𝑧−3

1 + 𝑧−1 − 𝑧−2 − 𝑧−3

hacer la discretización para el controlador.

Luego de esto se multiplica en cruz y se despeja U (z−1 ) que

𝐾𝑑2

𝑠3 + 𝐾𝑑1

𝑠2 + 𝐾𝑝 𝑠 + 𝐾𝑖

𝑠

se convierte en U (n)

## Planta

0.0608 𝑠3 − 1.353 𝑠2 + 0.604 𝑠 + 0.05059

𝑠

Ya remplazadas las constantes se discretiza el controlador para

esto hacemos uso del comando c2d en Matlab para hacer la discretización de manera rápida.

𝑡𝑠 = 30𝑠 𝑡𝑚 = 𝑡𝑠

40

Se tiene el valor del tiempo de establecimiento para definir el tiempo de muestreo deseado con esto el control discretizado es el siguiente.

## Control

−2.552𝑧3 + 2.971𝑧2 + 4.357𝑧 − 4.625

𝑧3 + 𝑧2 − 𝑧 − 1

Con esto pasamos a definir las ecuaciones en diferencia de la

planta y del control en escalón para poder hacer la simulación correspondiente en LabVIEW.

Antes de obtener las ecuaciones en diferencia se procede a discretizar la planta utilizando el comando c2d de matlab obteniendo la siguiente función de transferencia en el dominio z:

## Planta

0.01298𝑧3 + 0.03894𝑧2 + 0.03894𝑧 − 0.01298

𝑧3 − 1.708𝑧2 + 0.9692𝑧 − 0.2615

Ahora para la obtención de la ecuación se divide toda la función por la z de mayor orden para que nuestra función quede en términos de valores pasados.

## Planta

0.01298 + 0.03894𝑧−1 + 0.03894𝑧−2 − 0.01298𝑧−3

1 − 1.708𝑧−1 + 0.9692𝑧−2 − 0.2615𝑧−3

𝑈(𝑛) = 0.01298 𝐸(𝑛 ) − 0.03894 𝐸(𝑛 − 1)

+ 0.03894𝐸(𝑛 − 2) − 0.01298𝐸(𝑛 − 3)

+ 1.708𝑈(𝑛 − 1) − 0.9692𝑈(𝑛 − 2)

+ 0.2615𝑈(𝑛 − 3)

## Control

𝑈(𝑛) = −2.552𝐸(𝑛 ) + 2.971 𝐸(𝑛 − 1)

+ 4.357𝐸(𝑛 − 2) − 4.625𝐸(𝑛 − 3)

− 𝑈(𝑛 − 1) + 𝑈(𝑛 − 2) + 𝑈(𝑛 − 3)

Ya definidas las ecuaciones en diferencias se procede a hacer la simulación en LabVIEW

## Planta LabVIEW

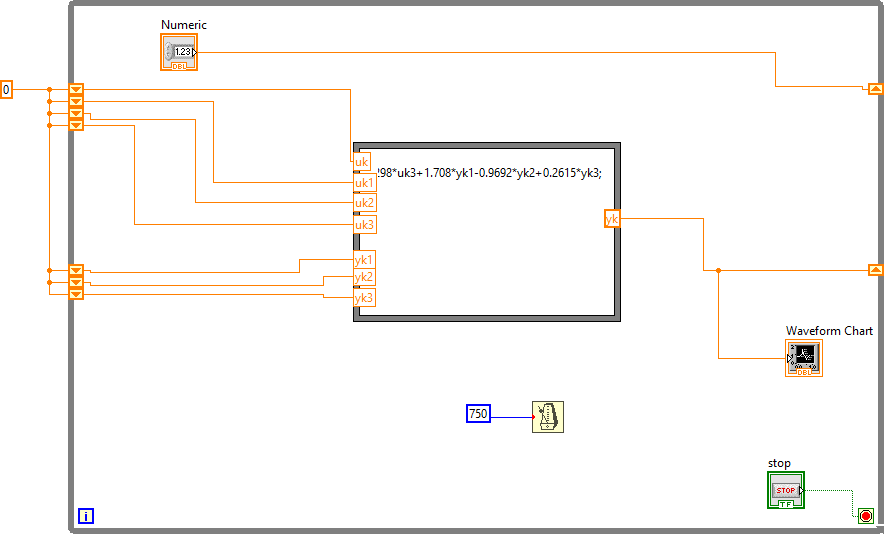


Figura3:Diseño de la planta LabVIEW

## Control

−2.552 + 2.971𝑧−1 + 4.357𝑧−2 − 4.625𝑧−3

1 + 𝑧−1 − 𝑧−2 − 𝑧−3

Ya con esto se procede a obtener la ecuación en diferencia

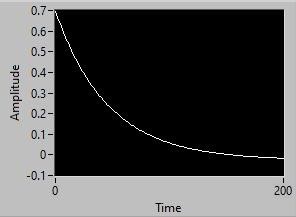
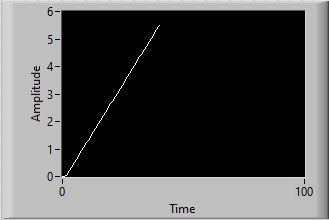


Figura4: respuesta de la planta LabVIEW

## Planta con control LabVIEW

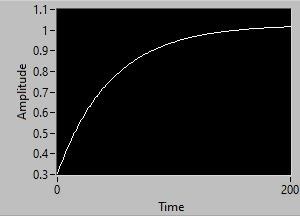


Figura4: Respuesta de la planta con control LabVIEW

## Figura6: Error del sistema LabVIEW

Como se puede apreciar el control funciona correctamente, pero también se demora bastante en estabilizarse.

## Control discreto-discreto

En esta parte para hacer el cálculo de los controles PID se toma la planta discretizada y se sacan los controles a partir de esta.

## Escalón:

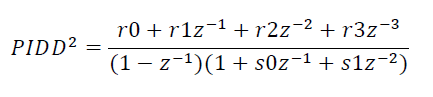
Primero se definen las condiciones del controlador:

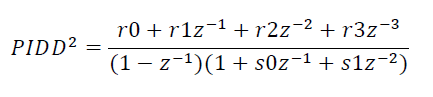
𝑡𝑠 = 3.8

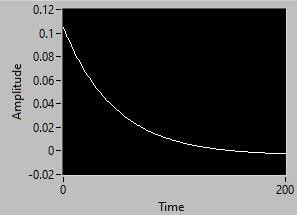
𝑧𝑒𝑡𝑎 = 0.800

𝑤𝑛 = 1.3158

Luego se procede a identificar la cantidad de constantes y la forma que tendrá el controlador para el caso de escalón se obtuvo la siguiente función:







Ya con nuestro controlador definido se pasó a obtener el valor de las constantes, para esto se obtuvo el polinomio deseado dando los siguientes coeficientes del polinomio:

𝑝𝑑

= 1, −1.994, 1.1841, −1.8945𝑒−1, 1.3056𝑒−2, −4.162𝑒−4, 5.0635𝑒−6

Ya con nuestro controlador definido se pasó a obtener los

valores de BR y AS donde 𝐵

𝐴

equivalen al numerador y

Figura5: Respuesta del control PID LabVIEW

Figura5: Respuesta del control PID LabVIEW

denominador de mi planta discreta respectivamente y 𝑅

𝑆

equivalen al numerador y denominador de mi controlador respectivamente entonces para obtener el valor de BR seria multiplicar el numerador de mi planta por el denominador de mi controlador y AS se obtiene de multiplicar el denominador con el numerador del controlador, ya con eso podemos calcular el polinomio característico de controlador y así hallar las siguientes constantes.

𝑠𝑜 = 1.2863

𝑠1 = 2.6352𝑒−1

𝑟0 = 5.49𝑒3

𝑟1 = −1.34𝑒4

𝑟2 = 1.09𝑒4

𝑟3 = −2.9346𝑒3

## -Rampa

Para rampa se utilizan las mismas condiciones del sistema y se obtiene la siguiente función que representa el controlador:

𝑟0 + 𝑟1𝑧−1 + 𝑟2𝑧−2 + 𝑟3𝑧−3 + 𝑟4𝑧−4

𝑃𝐼𝐼2𝐷𝐷2 =

(1 − 𝑧−1)(1 + 𝑠0𝑧−1 + 𝑠1𝑧−2)

Se obtienen las constantes del polinomio deseado:

𝑝𝑑

= (1), (−2.044), (1.2838), (−2.486𝑒−1), (2.252𝑒−2),

(−1.069𝑒−3), (5.0635𝑒−5), (−2.531𝑒−7)

Se obtienen los valores de BR y AS se suman y se igualan al polinomio deseado y despejamos las constantes definidas anteriormente, obteniendo las siguientes constantes:

𝑠𝑜𝑟 = 1.8014

𝑠1𝑟 = 3.8745𝑒−1

𝑟0𝑟 = 1.1120𝑒−4

𝑟1𝑟 = −3.4602𝑒−4

𝑟2𝑟 = 4.0169𝑒−4

𝑟3𝑟 = −2.1423𝑒4

𝑟3𝑟 = 4.3148𝑒3

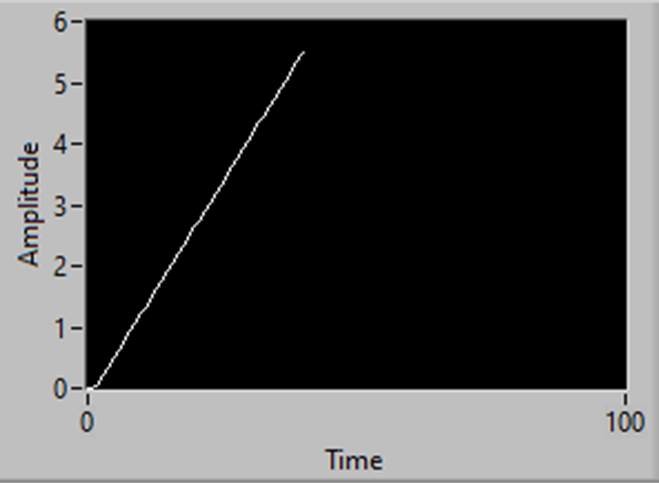


Figura7: Salida del sistema para rampa en LabVIEW.

* 1. CONCLUSIONES
  + A la hora de implementar esto en un microcontrolador se hace uso de las ecuaciones en diferencia definidas, pero además de eso se deben tener en cuenta otros parámetros como el voltaje máximo que alcanza el controlador.
  + Se concluye que el control discreto por medio de implementación en labview funciona correctamente para las entradas escalón y rampa.