Simulación de Sistemas Lineales Utilizando Labview

Bruno Vargas Tamani

Facultad de Ingeniería electrónica Eléctrica, Universidad Nacional Mayor de San Marcos, Lima, Perú

RESUMEN: Mostramos la manera en que se puede utilizar el lenguaje de programación LabView de National Instruments, para la simulación de sistemas lineales. Se justifica numéricamente los algoritmos a utilizar para poder lograr mediante un sistema discreto obtener respuestas convergentes al de un sistema continuo. Se describe la estrategia para programar sistemas de primer orden y de segundo orden. Se mostrarán los paneles de control (o interfase usuario), así como los diagramas de los sistemas diseñados.

ABSTRACT: We show how to use the programming language from National Instruments LabView, for the simulation of linear systems. It justifies using numerical algorithms to achieve through a unobtrusive system answers converging to a continuous system. It describes the strategy for scheduling systems of the first order and second order. There will be shown the control panels (or user interface), as well as diagrams of the designed systems.

PALABRAS CLAVES: simulación, LabView, panel, diagrama, muestreo, integración.

I. INTRODUCCIÓN

Los sistemas lineales representan la base para el diseño de controladores de procesos, para su posterior implementación. Con LabView podemos construir y probar el controlador interactuando directamente con el proceso; sin embargo luego del diseño del controlador, la verificación del desempeño del sistema de lazo cerrado no se puede realizar directamente por la falta de herramientas para representar los modelos que representan a los procesos. Sin embargo LabView nos ofrece todas las ventajas de un lenguaje de programación y podemos construir nuestros modelos con LabView, para poder simular sistemas de lazo cerrado [1], [2].

Nuestro objetivo es lograr simular en LabView las respuestas a excitaciones conocidas como la señal tipo escalón, de los sistemas lineales básicos, como lo son los sistemas de primer y segundo orden. A partir de estos sistemas, como es sabido podemos obtener representaciones lineales de cualquier orden.

Como LabView permite la implementación del controlador, podríamos verificar el comportamiento del sistema de lazo cerrado a nivel de simulación, utilizando los modelos de los procesos a controlar; en esa etapa obtendremos conclusiones respecto a los resultados a esperar en la realidad, teniendo la oportunidad de realizar ajustes del controlador de ser necesarios.

Para ese fin necesitamos establecer una estrategia de programación que nos permita obtener respuestas del sistema continuo, mediante la convergencia de un sistema discreto aproximado.

Justificaremos previamente esa estrategia, para luego describir como se implementó con LabView la simulación de los sistemas lineales básicos.

II. SISTEMAS DISCRETOS EQUIVALENTES SISTEMAS CONTINUOS

La simulación de sistemas continuos al realizarse en computadoras (de operación discreta), requiere que esto se realice mediante un algoritmo discreto, con lo cual en realidad lo que tratamos es el de lograr una buena aproximación de un sistema discreto al sistema continuo al cual trata de representar. Iniciaremos el planteamiento de tal aproximación, a partir de un sistema de primer orden.

A. *Sistema de Primer Orden*

Un sistema de primer orden, se representa mediante su función de transferencia $G(s)$:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K}{\tau s + 1} \qquad (1)$$

donde $y(t)$ es la salida del sistema y $Y(s)$ es su transformada de Laplace, asimismo $u(t)$ es la entrada del sistema y $U(s)$ es su transformada de Laplace; K es la ganancia del sistema y τ es la constante de tiempo del sistema.

A partir de (1), podemos obtener el modelo temporal del sistema, dado por:

$$\frac{dy(t)}{dt} + \frac{1}{\tau} y(t) = \frac{K}{\tau} u(t) \qquad (2)$$

Según (2), podríamos conocer la derivada de la salida a partir del valor actual de salida y de la entrada, así tenemos que :

$$\frac{dy(t)}{dt} = \frac{1}{\tau}(Ku(t) - y(t)) \qquad (3)$$

El resultado (3) es muy importante pues nos permitirá conocer la proyección de la salida en los instantes siguientes. Una aproximación discreta del significado de la derivada sería :

$$\frac{dy}{dt}(k) = \dot{y}(k) = \frac{y(k+1) - y(k)}{dt} \qquad (4)$$

donde $\dot{y}(k)$ es la derivada de la salida en el instante de muestreo k, $y(k)$ es la salida del sistema en el instante de muestreo k, $y(k+1)$ es la salida del sistema en el instante de muestreo $k+1$ (siguiente muestra), y dt es tiempo lo suficientemente pequeño de tal manera que permita la aproximación (4). El tiempo dt, se convierte luego de ser determinado adecuadamente en el período de muestreo T del sistema, para su aproximación discreta.

Si aceptamos la aproximación (4), entonces significa que si conocemos la derivada de la salida en el instante de muestreo k, entonces podremos conocer la salida del sistema en el instante de muestreo $k+1$, utilizando:

$$y(k+1) = \dot{y}(k)dt + y(k) \qquad (5)$$

La obtención de $\dot{y}(k)$ en el instante de muestreo k, se realiza a partir de (3), según :

$$\frac{dy}{dt}(k) = \dot{y}(k) = \frac{1}{\tau}(Ku(k) - y(k)) \qquad (6)$$

Las ecuaciones (5) y (6) son las que permiten lograr implementar un programa de simulación de un sistema continuo de primer orden. La segunda nos permite conocer la derivada de la salida con la información actual de entrada y salida; y la primera nos permite conocer la salida en el siguiente instante de muestreo. Estas relaciones son las que se implementarán mediante un programa en LabView para la simulación de un sistema de primer orden.

1).- Sistema de Segundo Orden

Un sistema de segundo orden, se representa mediante su función de transferencia $G(s)$:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \qquad (7)$$

donde K es la ganancia del sistema, ξ es el factor de amortiguamiento del sistema, y ω_n es la frecuencia natural no amortiguada del sistema.

Según (7), le corresponde un modelo temporal dado por:

$$\frac{d^2y(t)}{dt^2} + 2\xi\omega_n \frac{dy(t)}{dt} + \omega_n^2 y(t) = K\omega_n^2 u(t) \qquad (8)$$

Según (8), la segunda derivada temporal de la salida, se puede conocer a partir de la primera derivada de la salida y a partir del valor actual de salida y de la entrada, así tenemos que :

$$\frac{d^2y(t)}{dt^2} = \omega_n^2(Ku(t) - \frac{2\xi}{\omega_n}\frac{dy(t)}{dt} - y(t)) \qquad (9)$$

Una aproximación discreta de la segunda derivada temporal de la salida sería :

$$\frac{d^2y}{dt^2}(k) = \ddot{y}(k) = \frac{\dot{y}(k) - \dot{y}(k-1)}{dt} \qquad (10)$$

donde $\ddot{y}(k)$ es la segunda derivada de la salida en el instante de muestreo k, $\dot{y}(k)$ es la primera derivada de la salida del sistema en el instante de muestreo k, $\dot{y}(k-1)$ es la primera derivada de la salida del sistema en el instante de muestreo $k-1$, y dt es tiempo lo suficientemente pequeño de tal manera que permita la aproximación (10).

Conocida la segunda derivada en el instante de instante de muestreo k; a partir de (10) podemos encontrar la primera derivada en el instante de muestreo k :

$$\dot{y}(k) = \ddot{y}(k)dt + \dot{y}(k-1) \qquad (11)$$

Ahora aplicando (5) podremos conocer la salida del sistema en el instante de muestreo $k+1$, utilizando :

$$y(k+1) = \dot{y}(k)dt + y(k) \qquad (12)$$

La obtención de $\ddot{y}(k)$ en el instante de muestreo k, se realiza a partir de (9), según :

$$\frac{d^2y}{dt^2}(k) = \ddot{y}(k) = \omega_n^2\left(Ku(k) - \frac{2\xi}{\omega_n}\frac{dy}{dt}(k) - y(k)\right)$$
(13)

Las ecuaciones (11), (12) y (13), permiten diseñar un algoritmo para la simulación de un sistema continuo de segundo orden. La tercera calcula la segunda derivada de la salida con la información de la primera derivada de la salida actual y de la entrada y salida actual; la primera nos permite conocer la primera derivada de la salida actual con la información de la segunda derivada de la salida actual y la primera derivada de la salida del sistema en el instante de muestreo anterior; y la segunda nos permite conocer la salida en el siguiente instante de muestreo. Estas ecuaciones serán utilizadas para programar con Labview la simulación de un sistema de segundo orden.

III. PROGRAMAS DE SIMULACIÓN CON LABVIEW

Mostramos a continuación los programas desarrollados utilizando el lenguaje de programación gráfica, LabView de National Instruments versión 6.0, que nos permiten simular sistemas de primer y segundo orden. Se muestran los paneles de control (o interfase usuario) así como los diagramas de programación de los sistemas diseñados.

A. Sistema de Primer Orden

Mostramos en la figura 1, el panel de control a través del cual podemos obtener la simulación de un sistema de primer orden para una señal escalón.

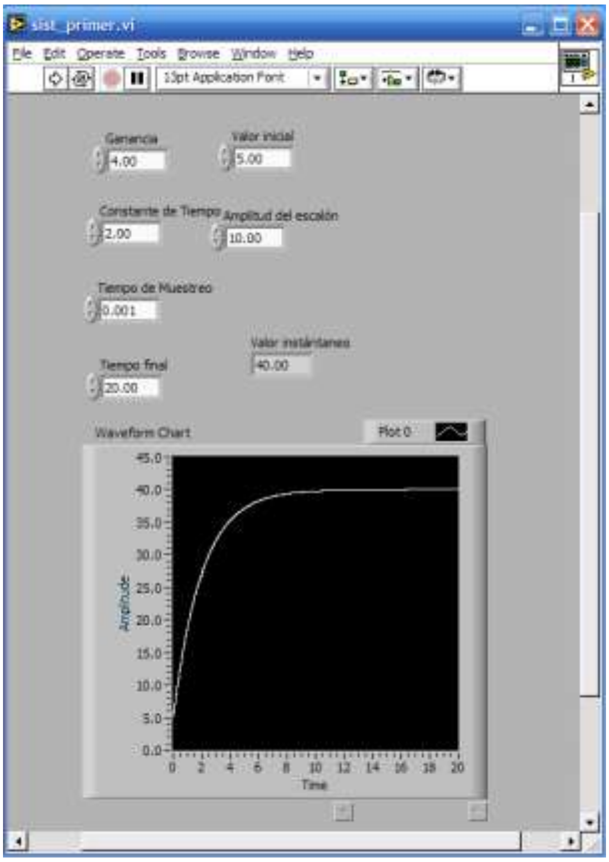
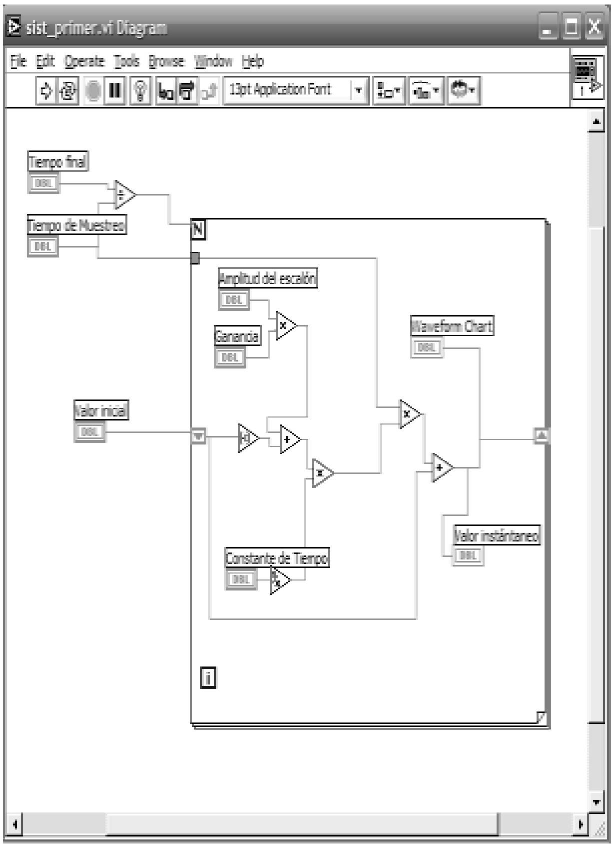


Fig. 1. Panel para la simulación de un sistema de primer orden.

Este panel de control, nos permite ingresar las parámetros del sistema, como son la ganancia del sistema como su constante de tiempo.

Podemos además ajustar el valor de la amplitud del escalón, el valor inicial de salida, el tiempo final de simulación, como el período de muestreo de simulación. El programa muestra en un presentador gráfico el resultado de la simulación y en un control el valor instantáneo de la salida.

La figura 2, muestra el diagrama de programación para la simulación del sistema de primer orden.



Fig, 2. Diagrama de programación para la simulación del sistema de primer orden.

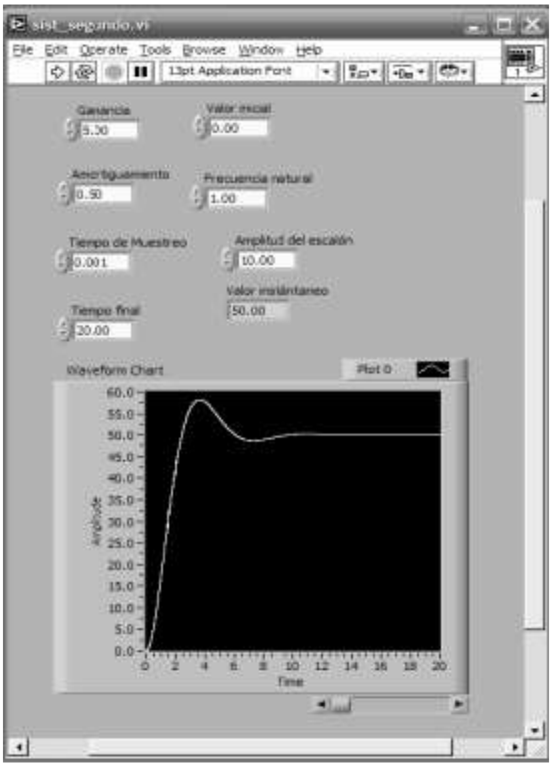
B. Sistema de Segundo Orden

Mostramos en la figura 3, el panel de control para la simulación de un sistema de segundo orden del tipo subamortiguado para una señal escalón. Se debe ingresar los parámetros del sistema, como son la ganancia, el factor de amortiguamiento y la frecuencia natural no amortiguada del sistema. También podemos ajustar la amplitud del escalón, el valor inicial de salida, el tiempo final de simulación, como el período de muestreo de simulación. En un presentador gráfico se obtiene el resultado de la simulación y en un control el valor instantáneo de la salida.

La figura 4, muestra el diagrama de programación para la simulación del sistema de segundo orden.

La figura 5, muestra el panel de control de simulación de un sistema de segundo orden del tipo sobreamortiguado para una señal escalón.

La figura 6, muestra el panel de control de simulación de un sistema de segundo orden del tipo oscilatorio para una señal escalón.



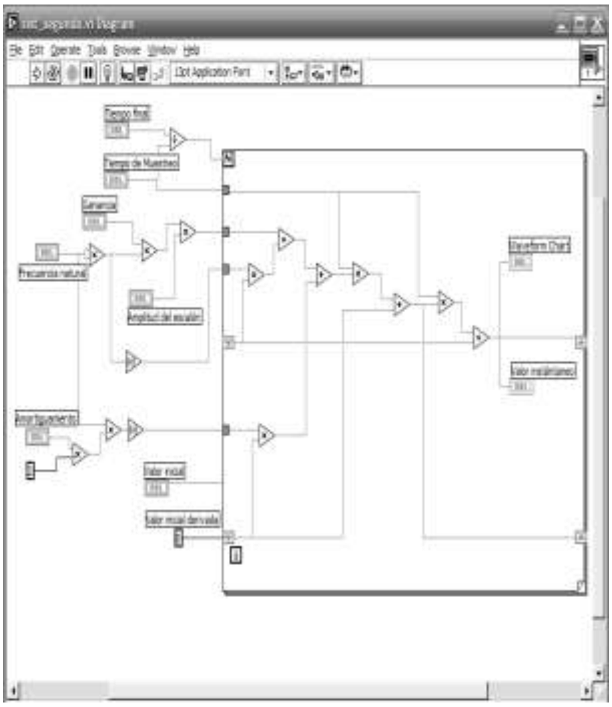
Fig, 3. Panel para la simulación de un sistema de segundo orden subamortiguado.

C. Aplicación: Simulación de la respuesta del nivel de un tanque

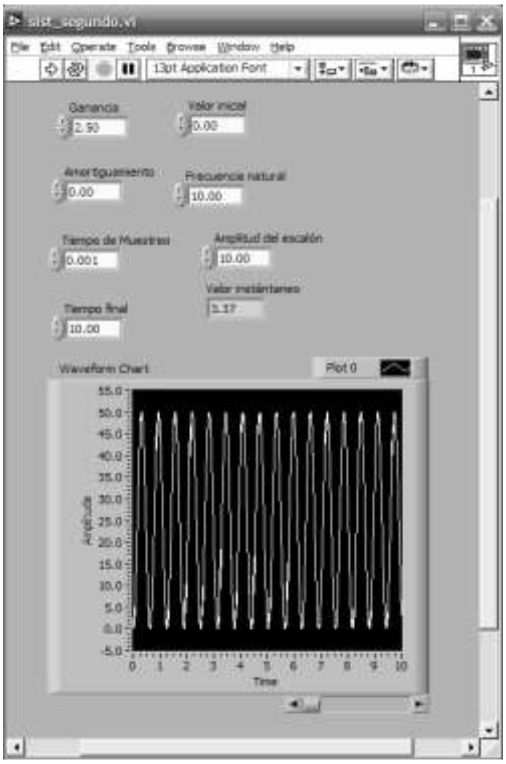
Los sistemas continuos simulados pueden servir de base para obtener la respuesta de sistemas de físicos que representan por ejemplos plantas de procesos, las cuales requieren ser simulados para posteriormente comprobar el diseño de sus controladores. La figura 7, muestra el panel de control para la simulación de la respuesta de nivel de un tanque.

Se pueden ingresar los parámetros del tanque, flujo de entrada, tiempo de muestreo y el tiempo de simulación. El resultado de la variación del nivel se muestra en un presentador gráfico y en un control tipo tanque disponible en LabView.

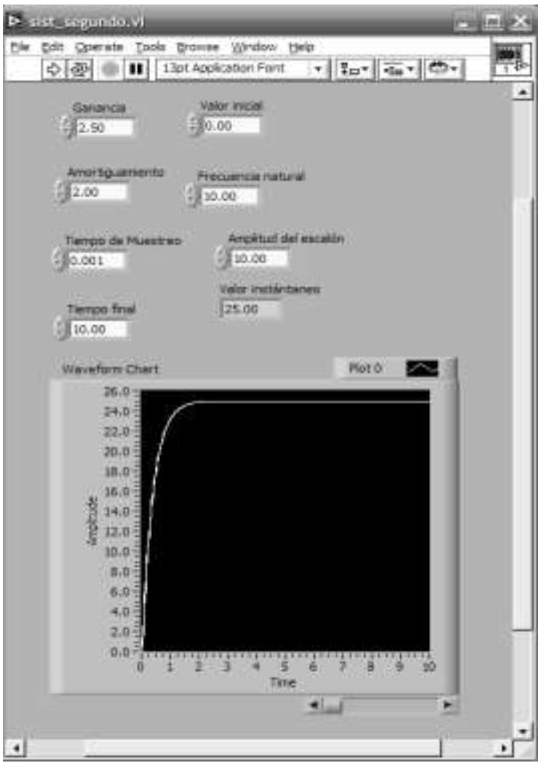
En la figura 8 se muestra el diagrama de programación para obtener la simulación de la respuesta de nivel del tanque. Como se observa, se ha utilizado la librería de programación de secuencias como una variación respecto a los diagramas anteriores; en la figura 9, se muestra la secuencia para el manejo del período de muestreo.



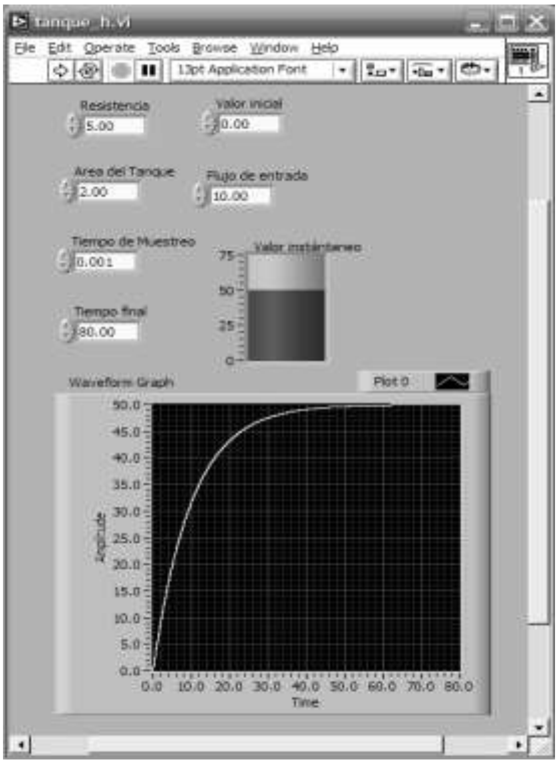
Fig, 4. Diagrama de programación para la simulación del sistema de segundo orden.



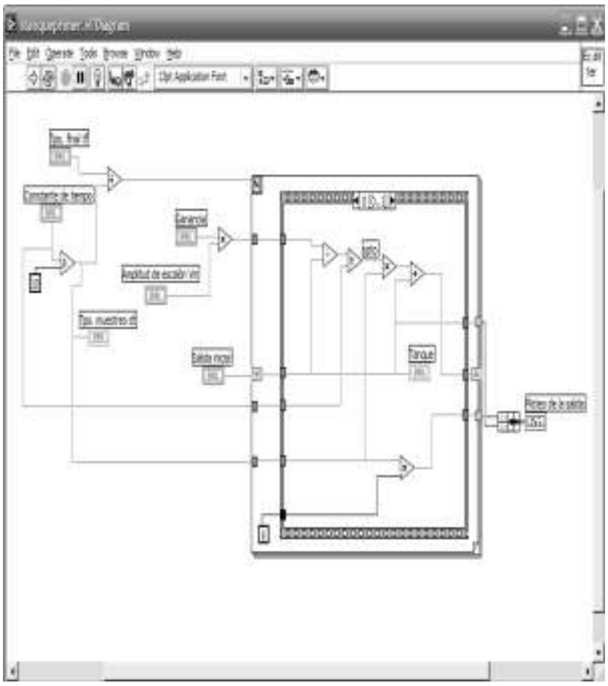
Fig, 6. Panel para la simulación de un sistema de segundo orden oscilatorio.



Fig, 5. Panel para la simulación de un sistema de segundo orden sobreamortiguado.



Fig,7. Panel para la simulación de la respuesta de nivel de un tanque.



Fig, 8. Diagrama de programación para la simulación de la respuesta de nivel de un tanque.

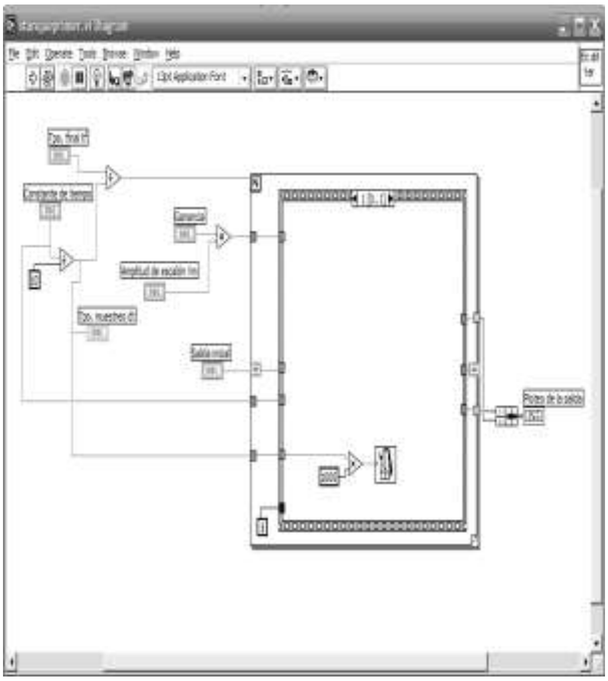


Fig. 9. Secuencia para programar el control del período de muestreo.

IV. CONCLUSIONES

El lenguaje de programación gráfica LabView, ha sido diseñado para la implementación de sistemas de instrumentación, permitiendo conectarse con dispositivos externos a través de librerías fáciles de configurar. Además, con LabView podemos programar modelos de sistemas lineales, los cuales son básicos para verificar nuestros diseños de controladores lineales, los cuales son generalmente utilizados en el control de procesos.

Es muy importante la elección del período de muestreo para lograr una buena simulación de sistemas lineales continuos, mediante algoritmos discretos.

Los sistemas de primer y segundo orden, son los sistemas elementales a partir de los cuales se pueden obtener sistemas de mayor orden, estos sistemas han sido programados y mediante su adecuada combinación se pueden simular sistemas de mayor orden.

REFERENCIAS

[1] Manuel Antoni, Biel Domingo, Olivé Joaquín, Prat Jordi, Sánchez Francesc. Instrumentación Virtual. Alfaomega 2002.

[2] Manuel Antoni. LabView. Programación Gráfica para el Control de Instrumentación. ITP Paraninfo 1997.