

## **Ambiente Multiusuario para Análisis y Diseño de Sistemas de Control**

**C.J. Munaro y M.F. Soares**

Universidad Federal de Espírito Santo, Departamento de Ingeniería Eléctrica  
29060-900 Vitória, ES-Brasil (e-mail: munaro@ele.ufes.br)

---

### **Resumen**

Este artículo presenta la estructura de un sistema multiusuario para la enseñanza y práctica del análisis y diseño de dispositivos de control. El sistema usa el software Matlab como herramienta computacional de apoyo. Varios usuarios pueden tener acceso a los dispositivos físicos conectados a un servidor y la conexión se hace usando puertos seriales. El sistema propuesto usa solamente una tarjeta de adquisición de datos y un sistema dinámico, optimizando así los recursos disponibles. El artículo muestra también algunas características y aplicaciones del sistema propuesto, demostrando su simplicidad y versatilidad.

*Palabras clave: control digital, adquisición de datos, control de dispositivos, enseñanza de la electrónica*

## **Multiuser Environment for the Analysis and Design of Control Systems**

### **Abstract**

This paper presents the structure of a multiuser system for the teaching and practice of analysis and design of control devices. The system uses the software Matlab as a computer tool support. Several users can have access to the physical devices connected to a central computer server and the connection between the users and the server is done using serial ports. The proposed system uses only one data acquisition card and a dynamic system, optimizing in this way the available resources. The paper also presents some characteristics and applications of the proposed system, demonstrating its simplicity and versatility.

*Keywords: digital control, data acquisition, control devices, electronic teaching*

*Nota: este artículo está tomado de "Información Tecnológica" [ISSN 0716-8756], vol. 9(2), 335-339 (1998)*

## INTRODUCCIÓN

La combinación de tarjetas con conversores analógico/digital-A/D y Digital/Analógico-D/A y programas para procesamiento de señales, permiten el desarrollo de herramientas muy poderosas para el análisis y control de sistemas dinámicos. El empleo de estas herramientas en cursos experimentales que involucran la enseñanza de la teoría de control, permite una gran variedad de aplicaciones con un mismo recurso. En Mendes et al. (1995), se propone en laboratorio para análisis y proyecto de sistemas de control basado en microprocesadores. En Soares y Casanova (1995), se construyen estaciones de trabajo compuestas por programas y tarjetas A/D y D/A con finalidad semejante.

En este trabajo, se describe un sistema multiusuario adecuado para uso en cursos experimentales de control. Diferentes estaciones de trabajo, compuestas por programas para comunicación y procesamiento de señales, comparten un único sistema dinámico, una tarjeta A/D y D/A y una impresora para generación de resultados.

Básicamente, el sistema consiste de una PC compuesta por una tarjeta de adquisición de datos funcionando como un servidor de recursos para otros tres PCs clientes localizadas en estaciones donde los grupos de alumnos realizan los experimentos. Con la finalidad de integrar los elementos del laboratorio, los PCs también se componen por programas desarrollados en lenguaje C. El PC servidor posee un programa responsable por la administración de las rutinas de control digital, adquisición y comunicación de datos, además de poder recibir archivos para ser impresos. Los PCs clientes poseen el software Matlab (Ogata, 1994) funcionando como el ambiente de trabajo de los usuarios y un programa que realiza la comunicación entre cliente y servidor transportando datos del ambiente Matlab para el programa del PC servidor y viceversa. Además, existen algunos sistemas físicos que se utilizan en los experimentos.

Esos sistemas son conectados al PC servidor y sus comportamientos pueden ser analizados simplemente utilizando los recursos de adquisición y control desarrollados y funciones del Matlab, todos en el propio ambiente. En las secciones siguientes se presentan con mayor detalle el sistema en cuestión, que se viene utilizando con éxito en la enseñanza de la teoría de control en la Universidad Federal de Espírito Santo.

## ESTRUCTURA Y COMPONENTES

La estructura del sistema propuesto se puede representar por el diagrama descrito en la Fig. 1. Tres PCs clientes están conectados a través de cables al PC servidor que, a su vez, posee el (los) sistema (s) físico (s) y una impresora conectados al mismo. Cada PC cliente está compuesto por el software Matlab y un programa desarrollado para comunicación serial y administración de datos (mayores detalles se verán en la próxima sección). Con el fin de establecer dicha comunicación, el PC servidor está compuesto por tres puertos seriales, un programa que administra las solicitudes de los clientes a través de estas puertas y una tarjeta de adquisición de datos. Esta tarjeta es la AIB-PCt de Sunset Laboratory (USA), que posee conversores A/D y D/A de 12 bits además de temporizadores que son fácilmente accesados por el programa. De la misma forma, se pueden elegir hasta 6 entradas y 2 salidas analógicas diferentes a través de programación de la tarjeta, permitiendo la realización de una gran variedad de tareas, que se describirán posteriormente en este trabajo.

El laboratorio cuenta con algunos sistemas físicos que son utilizados para experimentos y análisis, como un simulador de procesos y un kit para montaje de un servo-posicionador (ambos fabricados por *Feedback*). Además, posee un prototipo de un péndulo rotacional invertido modelado y desarrollado durante el proyecto de graduación de un estudiante. Estos sistemas ofrecen una razonable cantidad de situaciones que representan la realidad, permitiendo la aplicación práctica de los conceptos presentados por la teoría de control. Una impresora conectada al PC servidor completa la estructura del laboratorio. Este recurso puede ser usado por cualquier PC cliente en cualquier instante, puesto que el programa servidor está preparado para administrar impresiones aún cuando se solicitan adquisiciones de datos.

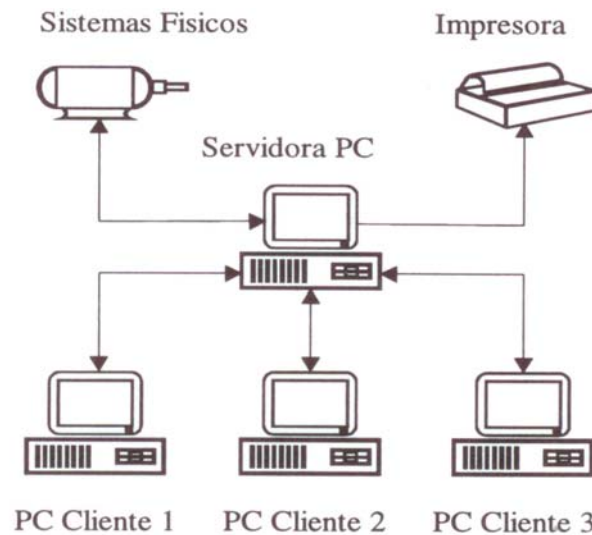


Fig. 1: Estructura del Laboratorio

### DETALLES DE IMPLEMENTACION

Para integrar el ambiente Matlab de un PC cliente con los recursos ofrecidos por el programa servidor, se desarrolló para cada recurso una función correspondiente codificada en archivos \*.m (*function files*). Estas funciones reciben del usuario (en el ambiente Matlab) argumentos que definen el perfil de la tarea a realizarse y revisan la consistencia de estos datos. El programa cliente es ejecutado dentro de este ambiente para administrar el proceso de comunicación enviando este bloque de datos por la puerta serial de la computadora y posteriormente recibiendo los resultados.

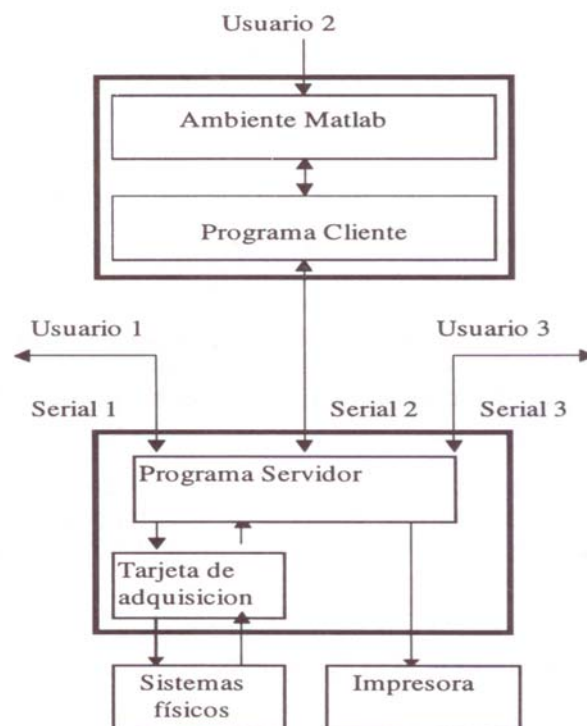


Fig. 2: Integración del Laboratorio propuesto

En el PC servidor, se verifican constantemente las puertas seriales de esta computadora. Al constatar la presencia de un dato específico para llamadas en una determinada puerta, se establece un sincronismo con el respectivo PC cliente que pasa a enviar el bloque de datos del

usuario. Los bytes recibidos correctamente son confirmados por el servidor con el retorno del carácter “\*”, al paso que fallas de comunicación se representan por “!”. La solicitud del cliente se ejecuta, entonces, vía tarjeta de adquisición de datos y tan pronto termina la ejecución, los datos son enviados al cliente y el servidor queda habilitado a atender nuevos pedidos. El cliente entonces deja estos datos disponibles en el ambiente Matlab, donde se pueden realizar muchos análisis a través de sus recursos. La figura 2 ilustra el proceso descrito.

Las funciones implementadas para acceder a recursos del servidor, mencionadas anteriormente se presentan a continuación, y se utilizan de la misma forma que la mayoría de aquellas disponibles en el Matlab. El mantenimiento de este patrón permite que respuestas de sistemas reales a determinadas entradas de excitación sean obtenidas de forma análoga a las respuestas de modelos matemáticos simulados. En todos los casos siguientes, la variable  $y$  recibe los valores resultantes de las operaciones;  $u$  contiene los valores que serán convertidos en tensión a cada  $ts$  milisegundos. Las variables *entrada* y *salida* reciben el valor de las entradas  $u$  y  $y$  analógicas donde serán, respectivamente, leídas y aplicadas las señales eléctricas

La función de excitación y muestreo de señales,  $y = \text{muestreo}(u, ts, \text{entrada}, \text{salida}, \text{malla})$ , realiza el muestreo de la señal de respuesta de un sistema sujeto a una excitación determinada por el vector  $u$ . Si el valor de *malla* es igual a 1, la adquisición de datos se realiza en malla abierta. En caso que sea igual a 2, la señal muestreada en la entrada analógica es realimentada por la computadora para formar, juntamente con la excitación  $u$ , una señal de error que es efectivamente aplicada a la salida analógica escogida, constituyendo así una adquisición en malla cerrada.

En la función de control digital pid,  $y = \text{pid}(u, ts, Kp, Kd, Ki, \text{entrada}, \text{salida})$ , la respuesta del sistema excitado por  $u$  se muestrea para aportar una señal de error que es corregida por el controlador digital PID de constantes  $Kp$ ,  $Ki$  y  $Kd$  y posteriormente aplicada a la entrada del sistema.

La función del control digital leadlag,  $y = \text{leadlag}(u, ts, a, T1, b, T2, \text{entrada}, \text{salida})$ , es análoga a la anterior. Sin embargo, los coeficientes del controlador digital se calculan a partir de una compensación del tipo avance-retraso (lead-lag) (Phillips y Nagle, 1990).

La función de filtro digital,  $y = \text{filtro}(u, ts, a0, a1, a2, b1, b2, \text{entrada}, \text{salida})$ , recibe directamente del usuario los valores de los coeficientes de un filtro digital de 2° orden, y realiza las adquisiciones de la misma forma que la función PID.

El controlador digital implementado por la función control digital vía realimentación de estados,  $y = \text{estados}(u, ts, Ac, Bc, Cc, Dc, p1, p2, \text{entrada}, \text{salida})$ , utiliza observación de estados para sistemas de 2° orden. Los argumentos  $Ac$ ,  $Bc$ ,  $Cc$  y  $Dc$  son matrices del modelo de estados de tipo:

$$\frac{dx(t)}{dt} = Ac \times x(t) + Bc \times u(t) \quad (1)$$

$$y(t) = Cc \times x(t) + Dc \times u(t) \quad (2)$$

$p1$  y  $p2$  son los valores deseados de los polos de la ecuación característica de malla cerrada del sistema. Estos datos son a continuación transformados al dominio discreto siendo calculada la ecuación característica del observador y finalmente los coeficientes del controlador digital que actuará sobre el sistema.

Vale destacar que las funciones sólo pasan parámetros para rutinas ejecutadas en tiempo real en el programa servidor, que retorna a ellas el resultado de la ejecución.

## EJEMPLOS DE APLICACIÓN

Utilizando el conjunto de funciones presentadas en la sección anterior, el usuario podrá realizar una serie de experimentos que envuelven tópicos tales como:

### *Adquisición de datos*

A través de la función “muestra”, se pueden aplicar diversas señales de excitación, formuladas en Matlab, a una de las dos salidas analógicas disponibles, pudiendo leer el comportamiento del sistema bajo la excitación por una de las seis entradas analógicas de la tarjeta de adquisición. Los resultados obtenidos se retornan en una variable del ambiente Matlab. Esta función se utiliza en clases experimentales para la obtención de modelos de primer orden (velocidad de motor dc) y segundo orden (servoposicionador). A través de la comparación de la respuesta típica de un sistema de primer y segundo orden, se obtienen los parámetros del sistema. Los parámetros también se obtienen vía respuesta en frecuencia del sistema. Se aplica una onda senoidal y se establecen las relaciones del módulo y fase entre la señal aplicada y respuesta, resultando en un gráfico de Bode aproximado para estos sistemas.

### *Identificación de Sistemas*

Las herramientas disponibles en el programa Matlab (*toolbox* de identificación) ofrecen diferentes métodos para generar modelos a partir de la respuesta muestreada del sistema.

### *Control Digital de Sistemas Físicos*

A partir del modelo obtenido como el descrito en el apartado anterior o a través de modelado matemático, el estudiante podrá proyectar controladores utilizando técnicas de la teoría de control y realizando las simulaciones con la ayuda de Matlab. A continuación, podrá observar el camino en la dinámica del sistema real ocasionada por la introducción del controlador digital utilizando una función como aquellas descritas.

En clases experimentales, se busca explotar esas funciones para introducir a los alumnos conceptos básicos y avanzados asociados al efecto de controladores sobre una planta. Ajuste de los parámetros de un controlador PID, por ejemplo, son fácilmente asimilables, ya que el resultado de estos ajustes se puede verificar tanto visualmente en el sistema real como por el gráfico de la respuesta muestreada. El efecto del tiempo muestreo sobre la controlabilidad de sistemas también es de fácil demostración experimental. Además, se proponen experimentos que involucran control analógico, utilizando para ello tiempos de muestreo bastantes inferiores a las constantes de tiempo del sistema en estudio. Para demostrar la actuación de los recursos presentados, se realizaron algunos experimentos utilizando procesos conocidos del simulador Feedback PCS327 Mk2. A continuación se muestran las secuencias de comando utilizados en el Matlab con los gráficos de los respectivos resultados.

a) Muestreo en malla cerrada de las respuestas del sistema  $G(s) = 1/s(s+1)$  a un escalón de 3V y una rampa (Figura 3).

```
y1= muestra (u, ts, entrada, salida, malla)
y2= muestra (r, ts, entrada, salida, malla)
u= ones (100,1);% vector con 100elementos
r= ones (100,1);% excitación tipo rampa
ts= 100; %ms
entrada= 1; salida= 1; malla= 2
```

b) Muestreo de la respuesta a un escalón de 3V del sistema  $G(s) = 1/s(s+1)$  en malla cerrada con control PID (Figura 4)

$Y = \text{pid}(u, ts, Kd, Ki, \text{entrada}, \text{salida})$   
 $U = \text{ones}(100, 1);$  %vector con 100 elementos  
 $Ts = 100\text{ms}; Kp = 3.0;$   
 $Kd = 1.5; Ki = 0.03;$   
 $\text{Entrada} = 1; \text{salida} = 1,$

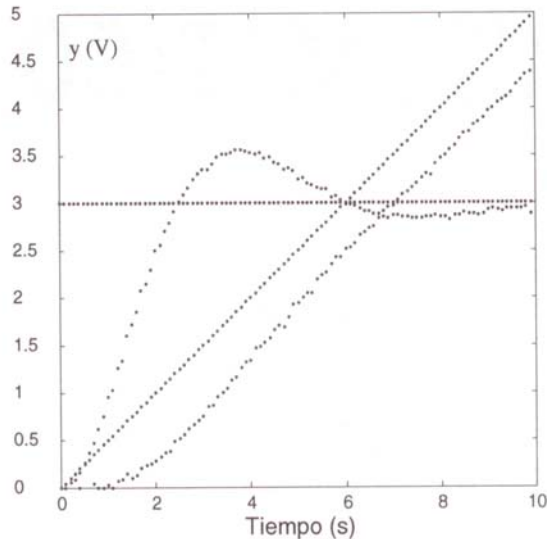


Fig. 3: Respuesta muestreada para cada entrada rampa/escalón

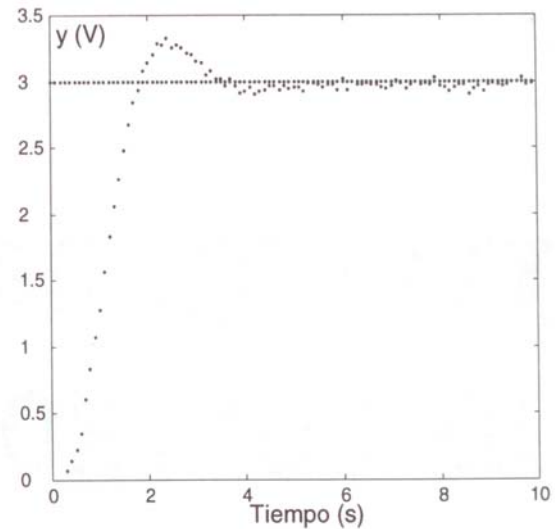


Fig. 4: Ejemplo de respuesta de malla cerrada

## CONCLUSIONES

El sistema propuesto se destaca por la simplicidad y versatilidad. Los equipos utilizados en conjunto con el uso compartido de recursos garantizan un bajo costo para la implantación del mismo. La integración de plantas físicas reales, sistema de adquisición de datos y ambiente de simulación Matlab garantiza un rango enorme de posibles actividades experimentales que involucra la teoría de control. El sistema multiusuario aquí propuesto ha sido utilizado con éxito en los cursos de control automático del Departamento de Ingeniería Eléctrica de la UFES (Brasil), que incluyen experimentos de control analógico y digital.

Se está eliminando la limitación del número máximo de usuarios para este sistema, por el uso de puertas seriales para la comunicación. Para ello, se está desarrollando un nuevo sistema que utiliza los recursos de red del software Windows para Grupos. Este nuevo sistema utiliza recursos tales como rutinas de 32 bits para acceso a la tarjeta A/D y D/A y protocolo TCP/IP para la comunicación entre servidor y clientes

## REFERENCIAS

Ogata, K.; Solving Control Engineering Problems with Matlab, Prentice-Hall (1994).

Mendes, R.S., P.A.V. Ferreira, C.A. Reis Filho y J.R. Oliveira; *Microprocesor-Based Laboratory for Control Systems Analysis and Design*, Workshop on Control Education and Technology Transfer Issues, Curitiba, Brasil, 11-12 de Septiembre (1995).

Phillips, C.L. y H.T. Nagle; *Digital Control System-Analysis and Design*, Prentice Hall (1990).

Soares Jr., L.R. y A.H.V. Casanova; *Laboratory Workstation and Experiments for Education on Discretization, Identification and Computer Control of Dynamical Processes*, Workshop on Control Education and Technology Transfer Issues, Curitiba, Brasil, 11-12 de Septiembre (1995).