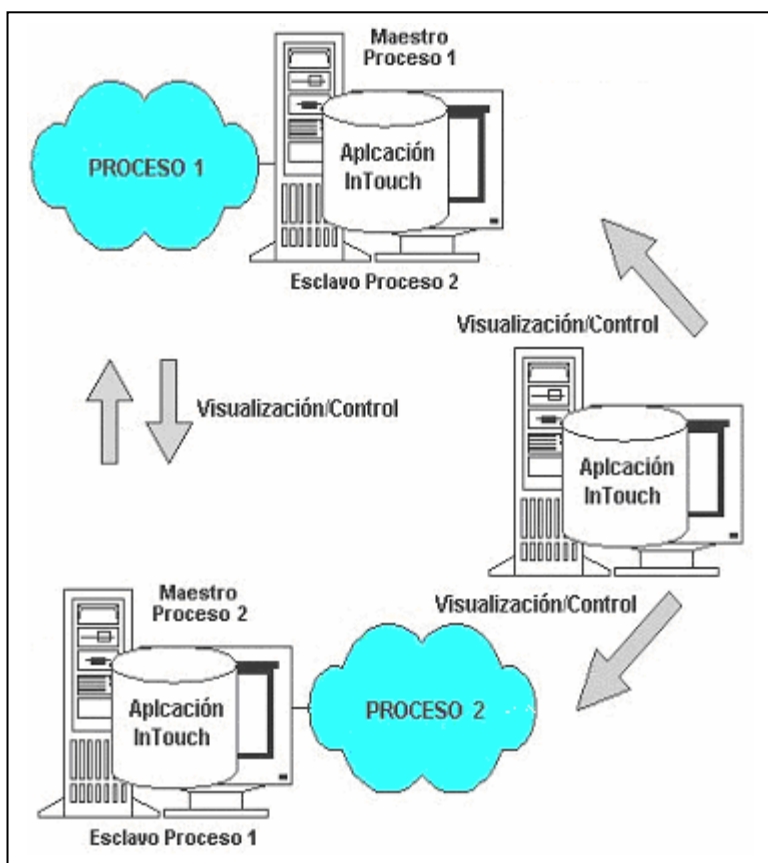


Diseño e implementación Sistema de control distribuido utilizando *InTouch* de *Wonderware*



RESUMEN

Este artículo presenta la descripción del diseño y la implementación de un Sistema De Control Distribuido que permite visualizar y controlar los módulos de procesos del Laboratorio de Automática de la Universidad Del Valle, usando el software especializado para control de procesos InTouch de Wonderware. Se desarrolló un driver de comunicación para la tarjeta de adquisición de datos TIM, que permite el transporte de información entre InTouch y los módulos de procesos.

PALABRAS CLAVES

Control Distribuido, ActiveX, Topologías de red, Driver, Control Automático

ABSTRACT

In this paper the design and implementation of a Distributed Control System is presented, The system displays and controls the process modules of the Universidad del Valle's Automatic Laboratory, using specialized software for processing In Touch of Wonderware control. Also, a communication driver to use the TIM acquisition card was developed, The card allows to transfer information between InTouch and the process modules.

- ❑ Ruby Cano Hernández
Ingeniera Electrónica
Universidad Del Valle.
- ❑ Paula Andrea Bermúdez
Ingeniera Electrónica
Universidad Del Valle
- ❑ Jesús Alberto González
Profesor Asociado
Escuela de Ingeniería
Eléctrica y Electrónica
Universidad Del Valle

- ❑ Edinson Franco Mejía
Profesor Titular
Escuela De Ingeniería
Eléctrica y Electrónica
Universidad Del Valle

*Grupo en Investigación
De Control Industria, GICI.*

KEYWORDS

Distributed control, ActiveX, network topologies, driver, automatic control.

1. INTRODUCCIÓN

Un objetivo en las industrias de procesos es combinar las unidades de procesamiento en un proceso o planta para transformar materiales crudos y energía de entrada en un producto terminado. Los principios básicos que guían la operación de estos procesos son:

- Deben ser operados con seguridad.
- La velocidad de producción debe ser mantenida.
- Las especificaciones de calidad del producto deben ser cumplidas.

Ya que muchos de los procesos industriales son por naturaleza dinámicos, los sistemas de control tienen la responsabilidad de supervisar e inducir cambios en las variables apropiadas relativas a la seguridad, producción y calidad del producto.

La premisa fundamental del control de procesos, es que la respuesta natural de todos los procesos dinámicos puede ser modificada por la influencia de un controlador. El objetivo es, por lo tanto, diseñar e implantar el controlador de manera que la respuesta dinámica sea modificada de acuerdo con un modelo deseado. Sin embargo, el grado o extensión en el cual la respuesta natural puede ser modificada apropiadamente está usualmente determinada por:

- La profundidad en el conocimiento de las características intrínsecas del proceso.
- La versatilidad de los elementos de hardware disponibles para la implantación del controlador.
- La naturaleza de las limitaciones inherentes al proceso.

Dado que no se puede hacer mucho sobre las limitaciones inherentes al proceso (como la presencia de tiempos muertos y retardos) y dado que la limitación impuesta por la rigidez de los controladores analógicos tradicionales ha sido superada por el empleo de las computadoras digitales, se ha llegado a la situación de que los mayores retos del control de procesos radican en que éstos son enormes, complejos y pobremente entendidos.

Lo anterior explica la tendencia actual de un creciente entendimiento de los procesos, con lo cual, debe ser posible desarrollar modelos más adecuados y precisos; estos modelos y el

conocimiento obtenido de su análisis hacen más fácil el diseño de sistemas de control más efectivos. En cada etapa del modelado, análisis, diseño y supervisión, la computadora juega un papel de gran importancia; además, la implantación del controlador resultante es llevada a cabo en una computadora digital. Entonces, de todos los factores que influyen en el estado actual del control de procesos, quizá ninguno es tan significativo como la computadora, ya sea como un elemento de hardware en donde se implanta el esquema de control avanzado, o como el dispositivo que facilita el análisis y monitoreo. Esto se evidencia en el hecho que prácticamente todas las plantas modernas de procesos están equipadas con su propia red de computadoras, la cual es conocida como "Sistema de Control Distribuido" (SCD o DCS).

El diseño e implementación de un Sistema de Control Distribuido en el Laboratorio de Automática de la Universidad del Valle fue el objetivo principal del desarrollo del presente proyecto.

2. CARACTERÍSTICAS DE LA APLICACIÓN

Actualmente, existe una gran oferta de software de adquisición, supervisión y control y la evolución de este software sigue una triple tendencia: intercomunicación entre aplicaciones, estandarización de las comunicaciones con los dispositivos de campo y adopción del entorno de comunicación Internet.

Entre los elementos que caracterizan la oferta actual de software de supervisión y control se destacan la programación en *Visual Basic* o *Visual C* para Aplicaciones, uso de tecnología ActiveX, compatibilidad con el entorno y tecnología de *Microsoft*, y uso del Sistema Operativo *Microsoft Windows 2000*.

Las ventajas de la utilización de las redes de comunicación en los entornos DCS se resumen en la obtención de la información en tiempo real de la planta para cualquier persona autorizada de la misma, esté donde esté.

Con el desarrollo de este proyecto, el Laboratorio de Automática de la Universidad del Valle cuenta con una aplicación que permite la gestión remota de un sistema DCS para realizar las tareas de supervisión y control del sistema sin limitaciones de distancia.

La gestión remota de un sistema DCS se compone, por una parte, del sitio dónde está ubicado el

computador principal o servidor (llamado comúnmente centro de control), y por otra, los computadores clientes que vía red con protocolo TCP/IP, pueden realizar gestión remota del sistema.

Las herramientas utilizadas en el desarrollo del proyecto son:

- ❑ *Wonderware FactorySuite* para aplicaciones DCS.
- ❑ Sistema Operativo *Microsoft Windows 2000 Server* y Profesional.
- ❑ *Microsoft Visual C++ 6.0* para el desarrollo de componentes ActiveX y de programas ejecutables.
- ❑ *Microsoft Excel* para la creación de bases de datos.

La aplicación desarrollada cumple con las siguientes especificaciones:

- ❑ Interfaz gráfica similar a los módulos del Laboratorio de Automática.
- ❑ Adquisición de datos a través de la Tarjeta Interfaz Multipropósito (TIM)
- ❑ Visualización del estado del sistema DCS, así como interacción con él mediante el envío de directivas de control.
- ❑ Base de datos que registra todos los valores de los parámetros de la aplicación DCS.

2.1. Tarjeta Interfaz Multipropósito TIM.

La Tarjeta Interfaz Multipropósito (TIM) encargada del sistema de adquisición de datos está conformada por dos tarjetas, una interna y otra externa, que permite intercambiar datos análogos y digitales entre el computador y el mundo exterior. La tarjeta interna es para inserción en una ranura tipo ISA del computador, la cual consta de los siguientes bloques funcionales:

- ❑ Bloque de decodificación de puertos e interfaces con el PC
- ❑ Bloque de entradas analógicas
- ❑ Bloque de salidas analógicas
- ❑ Bloque de puertos digitales de entrada y salida
- ❑ Bloque de contadores y temporizadores
- ❑ Bloque de interrupciones

La tarjeta externa soporta la interfaz física de usuario y funciona como bornera para las conexiones análogas y digitales, donde además se permite la visualización por medio de Led's de los datos digitales que entran y salen por los puertos E/S digitales de la tarjeta interna.

El software especializado *Wonderware FactorySuite* incluye una serie de drivers que permiten realizar la adquisición de los datos a través de sistemas SCADA o PLC's comerciales (*I/O Server*), sin embargo, también permite la adición de otros sistemas no tan genéricos mediante los denominados Controles ActiveX. Debido a que en el Laboratorio de Automática la adquisición de los datos de los módulos de proceso se realiza a través de la Tarjeta Interfaz Multipropósito, TIM, se hizo necesaria la inclusión de un driver para dicha tarjeta en el software de *Wonderware FactorySuite*, lo que implicó el desarrollo de un Control ActiveX de manera que se permitiera el manejo de puertos desde cualquiera de sus aplicaciones teniendo en cuenta las limitaciones de acceso que interpone el Sistema Operativo *Microsoft Windows 2000 Server* y Profesional.

Cabe notar que incorporar un control ActiveX en un programa "drag and drop" como Visual Basic, Visual C o el mismo *InTouch* de Wonderware, supone añadir un objeto con código asociado que realiza una determinada función de forma totalmente integrada dentro de la aplicación que se esté tratando.

3. DRIVER DE COMUNICACIÓN ENTRE LA TIM E INTOUCH DE WONDERWARE.

Aprovechando las características de *InTouch* de ser un Contenedor ActiveX se desarrolló un Control ActiveX que permitiera el acceso a los puertos de la TIM desde la aplicación de *InTouch*, dicho control fue implementado en Visual C++.

Un Control ActiveX se compone de propiedades, eventos y métodos.

Las propiedades son muy similares a variables que se pueden modificar, son atributos aplicados al control. Normalmente, los Controles ActiveX permiten un acceso completo a sus propiedades. Los contenedores pueden leer directamente estas propiedades y modificar cualquiera de sus valores; sin embargo, los controles no pueden intervenir sobre ellas.

Los eventos son mensajes enviados desde el control hacia la aplicación que los contiene.

Los métodos son funciones expuestas por el control y llamadas por el contenedor. Leen parámetros y les asignan un valor.

Aplicar métodos es una solución sencilla para que un contenedor se comunique con un Control ActiveX. Pero además, esta conversación también puede establecerse en sentido contrario. Al enviar un evento, el control transmitirá información "de retorno" al contenedor. Los eventos pueden informar al contenedor de cualquier contingencia producida

InTouch permite acceder a las propiedades, métodos y eventos del Control ActiveX y esas propiedades pueden ser asociadas con los tagnames. También puede tenerse acceso a ellas a través de QuickScripting

La comunicación con los puertos desde Windows 2000 se realizó por medio de un driver llamado Porttalk quien funciona en el ring 0 (nivel 0 del privilegio de E/S) y permite al programa en modo usuario tener acceso a los datos que pueden pasar hacia y desde el driver por medio de llamadas IOCTL.

El driver contiene dos funciones IOCTL para lectura y escritura a los puertos E/S. Se puede utilizar un archivo fuente de c, `pt_ioctl.c` para proveer soporte basado en las funciones `inportb/outportb` e `inp/outp` soportadas en ambientes de programación comunes. Incluyendo `pt_ioctl.c` e invocando `OpenPortTalk` cuando el programa inicia, y `ClosePortTalk` cuando el programa termina, se tienen las funcionalidades de `inportb/outportb` e `inp/outp`.

Con esta información se implementó una librería llamada `puertos.c` para incluirla en el conjunto de librerías del Control ActiveX de modo que este quedara con las funcionalidades del acceso a los puertos.

El Control ActiveX creado se llama `Adq2ctrl` y debe ser instalado en *InTouch* para después ser llamado desde el Wizard. El Control posee básicamente dos métodos los cuales por medio de parámetros permiten su configuración. El método `Escribir(dir, dato)` permite configurar la dirección del puerto a la que se requiere acceder y el dato que se desea enviar. El método `Leer( Dir )` permite configurar la dirección del puerto de quien se quiere obtener información.

1. DESARROLLO DE LA APLICACIÓN.

Una vez diseñado el driver de comunicación entre la tarjeta de adquisición de datos TIM e incluido en *Wonderware- InTouch*, se procedió a elaborar las

representaciones gráficas realizadas de los módulos de procesos del Laboratorio de Automática de la Universidad del Valle, una muestra de ello se presenta a continuación, al igual que una breve descripción de cada uno de los procesos involucrados en el Sistema de Control Distribuido.

4.1 Control Neumático de Presión (PCT14).

El módulo de Control Neumático de Presión (PCT14) es un equipo diseñado para introducir y demostrar los principios de medida y control de procesos. Este módulo incluye componentes neumáticos que son de interés a la industria de procesos, como son los transductores de corriente/presión. Está formado por una tubería en la cual se monta una válvula de control operada neumáticamente, una placa de orificio, un tanque de almacenamiento, manómetros y válvulas todo o nada.

La válvula neumática es operada desde un conversor I/P (corriente/presión), y transductores directos y diferenciales de presión facilitan la medida de presión y flujo respectivamente.

El software de visualización *InTouch* permite además enlaces de animación para que la información sea vinculada a la pantalla para ser vista y modificada dinámicamente.

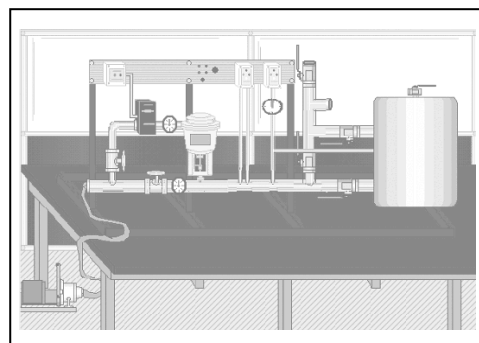


Figura 1. Control Neumático de Presión Representación gráfica elaborada con *InTouch*

A continuación se presenta una imagen del controlador implementado Fig. 2. en el ambiente *InTouch* de Wonderware, dicho controlador permite la configuración de un algoritmo PID interactivo, hacer cambios al SetPoint, estudiar las respuestas obtenidas de los procesos mediante gráficos en tiempo real, trabajar en modo manual y en modo automático y además configurar el modo de acción de la señal de control de acuerdo al tipo de actuador.

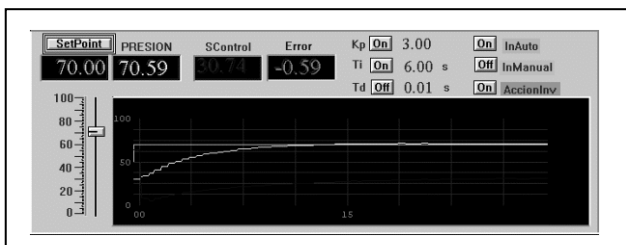


Figura 2. Ventana del Controlador

4.2 Control de Nivel (PCT-9).

En el Control de Nivel, se manipula una referencia que determina el valor de nivel que se requiere en el tanque de almacenamiento de agua (planta o proceso). La señal de realimentación la proporciona un sensor de nivel (flotador) y es llevada a acondicionador de señal, se compara con la señal de referencia y genera su respuesta de acuerdo a un algoritmo de control preestablecido que acciona una válvula motorizada (actuador). El agua es extraída del tanque de almacenamiento por una bomba centrífuga, y los cambios en el flujo de agua son medidos a través de un rotámetro. En el módulo PCT9, el nivel mínimo corresponde a 4 mA (0.2 voltios) y el nivel máximo a 20 mA (1 voltio).

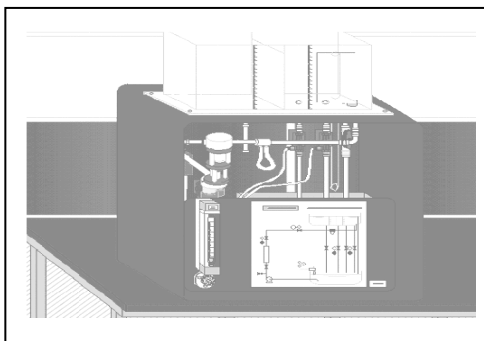


Figura 3. Control De Nivel (PCT-9). Representación gráfica elaborada con InTouch

4.3. Control de temperatura.

Este módulo contiene un intercambiador de calor a través del cual el agua caliente de un depósito eléctricamente calentado circula en contra corriente al flujo de agua fría del proceso. Termocuplas están localizadas en la entrada y salida del torrente del intercambiador y la variación en el flujo es monitoreada. El flujo del líquido del calentador se controla por medio de una válvula motorizada o por el encendido / apagado de la bomba de circulación.

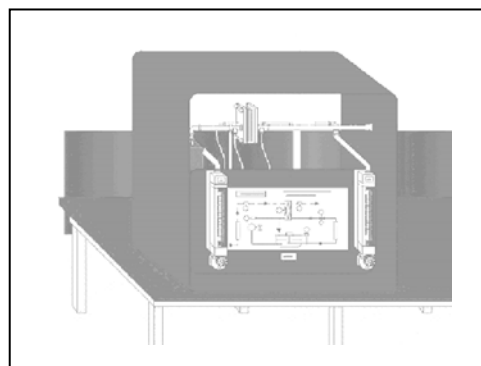


Figura 4. Control de Temperatura (PCT-13)

4.4. Control de PH (PCT16).

En este proceso se controla el nivel de acidez de una sustancia llamada efluente por medio de la adición de otra llamada reactivo. Estas dos sustancias llegan a un tanque y son mezcladas con un agitador para obtener el nivel de concentración adecuado a la salida del tanque. La adición del reactivo está controlada por una válvula dosificadora que está conectada directamente con la salida del controlador.

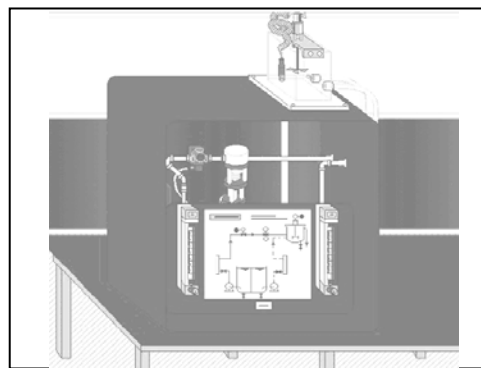


Figura 5. Control De PH. Representación gráfica elaborada con InTouch

2. ARQUITECTURA IMPLEMENTADA

Aunque InTouch presenta la arquitectura NAD como una muy buena alternativa para distribuir aplicaciones, no es posible desarrollar el DCS que se implementó a partir de esta arquitectura debido a sus limitaciones para realizar comunicaciones cuando se tienen diferentes procesos en diferentes nodos.

La arquitectura de red implementada se basa en la topología de red Maestro/Esclavo proporcionada por InTouch. En esta arquitectura, el Maestro es el nodo conectado directamente al proceso y el Esclavo es el nodo View que puede ejecutar la misma aplicación.

Se tienen entonces tres computadores, dos conectados directamente a procesos -nodos Maestro- que a su vez son esclavos el uno del otro. El tercer computador es un nodo *view* de los que ejecutan las aplicaciones de los nodos Maestro y en el cual se pueden almacenar copias de seguridad de las aplicaciones y las bases de datos arrojadas por los gráficos de históricos de cada proceso.

Tal como está concebida la implementación del DCS, se puede realizar control y visualización total de los procesos desde cualquier nodo, con permisos de seguridad dependientes del Nivel de Acceso de cada usuario.

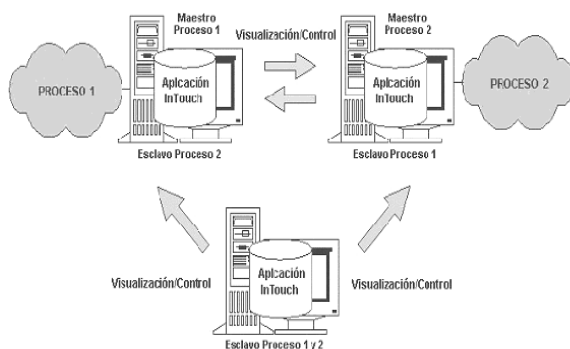


Figura. 6. Arquitectura de red implementada

Una desventaja de ésta arquitectura, es que en el caso de caerse un nodo Maestro, se cae el proceso conectado a él y además se pierde la comunicación con el nodo debido a la forma de adquisición de los datos, ya que la tarjeta de adquisición de datos TIM está conectada directamente con el puerto ISA del computador. Si la falla ocurre por una caída en la red de comunicación, el proceso continúa activo en su nodo Maestro. Una posible solución a este problema, es conectar el proceso directamente a la red, por ejemplo a través de TCP/IP, y así crear una aplicación redundante que ponga en cada nodo la información completa del proceso. Aquí se estaría creando ahora una dependencia total del DCS a la red de comunicaciones.

Para la configuración de la aplicación distribuida, una vez se ha realizado la interfaz gráfica, se asocian los objetos que intervienen en ella con sus correspondientes Tagnames y se prueba su funcionamiento en su nodo Maestro. Luego, se hace una copia exacta de la interfaz gráfica en cada nodo esclavo, pero los objetos de la aplicación son ahora asociados a los Tagnames remotos del nodo Maestro. Para configurar y poner en marcha las aplicaciones, es necesario habilitar los permisos de acceso (nombre de usuario y contraseña) que se

establecen en el sistema operativo sobre los nodos Maestros y Esclavos.

7. CONCLUSIONES

7.1. El DCS desarrollado, es una aplicación académica con fines didácticos, enfocada a utilizar los recursos existentes en el Laboratorio de Automática. Con ello se aporta al Laboratorio de Automática una diferente y novedosa forma de desarrollar prácticas de laboratorio en control clásico y avanzado a través de un software ampliamente utilizado en la industria de procesos. Además se constituye una plataforma de trabajo muy amigable que contribuye a una mejora en las prácticas docentes y que por lo tanto incentivará el gusto por la automática a las próximas generaciones de estudiantes.

7.2. InTouch de Wonderware es un software desarrollado para supervisión de procesos que permite remotamente incidir sobre un proceso. No ha sido pensado para albergar el control, sin embargo, en la aplicación se desarrolló un controlador que permite administrar las características de lazo abierto y lazo cerrado de los sistemas e integrar a su funcionamiento un algoritmo PID interactivo.

7.3. La implementación del DCS con InTouch de Wonderware evidenció la potencialidad de sistemas como éste y por lo tanto su gran difusión a nivel industrial, pues es claro que en procesos reales un control a este nivel otorga muchas ventajas tales como mejoras en la calidad y cantidad de producción, reducción de costos, optimización del mantenimiento preventivo y correctivo y simplificación de la administración de la planta en general.

8. BIBLIOGRAFÍA

- [1] BENNETT, Stuart. Real-Time Computer Control. Prentice-Hall. Estados Unidos. 1988.
- [2] CREUS SOLE, Antonio. Control Distribuido: Versatilidad, seguridad y fiabilidad del sistema. En: Mundo Electrónico, No.100, 1980, págs. 93_100.
- [3] GONZÁLEZ, J. Amable. Sistemas de Control Distribuido, parte I. En: Ingeniería Química. Enero, 1982, págs. 119 – 130.
- [4] IBARQUEN OCAMPO, Francisco Javier – PULIDO RAMIREZ, Ricardo Alfredo. Sistemas de Control Distribuido aplicados a la

Automatización. Tesis. Ingeniería Eléctrica.
Universidad del Valle. 1995.

[5] <http://penta.ufrgs.br/rc952/trab2/map.html>

[6] <http://www.automatas.org/redes/profibus.htm>

[7] <http://www.led.uc.edu.py/str/segunda.htm#situacion>

[8] www.wonderware.com

[9] <http://www.gssystem.com/new/Tech/htmltechnotes/intouch.htm>

AUTORES

Jesús Alberto González. Tecnólogo especializado en control industrial de la Universidad del Valle (19--).

Profesor asociado de la Escuelas de Ingeniería Eléctrica y Electrónica de la Universidad del Valle
jegonza@univalle.edu.co

Edison Franco Mejía. Ingeniero Electricista, Magíster en Automática, estudiante de Doctorado en Ingeniería, Universidad del Valle, Colombia.

Profesor titular de la Escuela de Ingeniería Eléctrica y Electrónica de la Universidad del Valle
efm@univalle.edu.co

Paula Andrea Bermúdez Restrepo. Ingeniera Electrónica de la Universidad del Valle. (2003)

pandreab@terra.com.co

Ruby Cano Hernández. Ingeniera Electrónica de la Universidad del Valle. (2003)

Rubyc7@yahoo.com