

CONEXIÓN NORMALIZADA DEL EQUIPAMIENTO EN LOS LABORATORIOS DE ENSEÑANZA DE CONTROL

Roberto Saco¹, Anibal Zanini² y Carlos A. Godfrid³

Abstract — En este trabajo se describe una forma de interconexión normalizada de diferentes procesos y plantas de laboratorios para la enseñanza de control automático. El sistema de adquisición y control es común para todos los prototipos y se implementa con una PC estándar provista con software para control en tiempo real. La conexión de las diferentes plantas con las PC's, usadas como controladores, se realiza mediante una red CAN (Controller Area Network), y de esta forma se logra hacer intercambiables a las diferentes plantas existentes mediante una conexión normalizada. Además se garantiza que las futuras plantas incorporadas al laboratorio podrán ser controladas con cualquiera de las PC's que posean la interfaz CAN. Para el diseño de los algoritmos de control se utiliza los paquetes Simulink, Real Time Workshop y Real Time Windows Target de MATLAB. El manejo del protocolo CAN, se realiza mediante una S-Function de Simulink la cual es transparente para el diseño del algoritmo de control. En base a los resultados satisfactorios obtenidos se propone una normalización para interconexión de procesos y plantas de laboratorio basada en la red CAN.

Index Terms — CAN networks, control education, equipment interconnection, MATLAB, real-time control, Simulink.

INTRODUCCIÓN

En los últimos años se encuentra una tendencia generalizada a incorporar más trabajo experimental en la formación en control automático. Tradicionalmente esta formación se centraba en cursos con fuerte contenido teórico, pero cada vez más se reconoce la importancia de enfrentar problemas de la práctica profesional en el marco de los estudios de ingeniería. En este sentido, se citan frecuentemente las conclusiones del Workshop de la National Science Foundation y la Control Systems Society de la IEEE (NSF/CSS) con nuevas directivas para la educación en control automático, que impulsan a modificar los actuales métodos de enseñanza [1]. Un primer acercamiento a los problemas de la práctica se consigue con el uso de herramientas de simulación. En este sentido se han desarrollado varios "Laboratorios Virtuales" aprovechando las posibilidades y recurso de Internet [2],[3].

En cambio, no hubo un desarrollo similar en lo relativo a experimentos reales. Y son las aplicaciones y el control en

tiempo real los ingredientes esenciales en el aprendizaje de la ingeniería en automatización. Creemos que un factor limitante en el desarrollo de experiencias en este campo se debe a que cada aplicación implica, además del experimento en sí, un trabajo considerable de interfaces, conexionado y software dedicado. Además, estas tareas requieren frecuentemente, del aporte de especialistas de diferente orientación. Por ello entendemos que disponer de una interfaz normalizada, común a las distintas experiencias de un laboratorio, permitirá desentenderse de los aspectos laterales, para concentrarse en el experimento de interés.

Por otra parte, la internacionalización de la educación universitaria se expande día a día, tanto en las carreras de grado como en las de postgrado. La necesidad de compatibilizar las herramientas usadas en las distintas disciplinas es un punto importante para facilitar la integración de los estudiantes. La estandarización de los métodos de trabajo e interconexión en el terreno experimental facilitará la posibilidad de cooperar y compartir experiencias entre instituciones tanto nacional como internacionalmente.

En este marco la idea propuesta en este trabajo es la estandarización, a nivel educativo, de sistemas de control en tiempo real y aplicaciones experimentales. La utilización del software MATLAB/SIMULINK juntamente con el paquete para control en tiempo real RTW ya está siendo utilizada como un estándar por varias instituciones [4]. Este paquete permite generar algoritmos, juntamente con su código para la ejecución en tiempo real, en forma gráfica. De esta forma se logra que los estudiantes y profesores se pueden concentrar en el diseño del sistema de control, su implementación y evaluación sin necesidad de entrar en la tediosa tarea de programación con lenguajes de bajo nivel e interfaces a hardware [5].

LA RED CAN

La red CAN (Controller Area Network) es una red de comunicación serie que fue originalmente diseñada para su utilización en la industria automotriz, aunque actualmente, también se ha hecho una red popular en la industria de la automatización y otros campos. La red CAN se utiliza principalmente en sistemas embebidos y ofrece una excelente solución para manejar la comunicación entre múltiples microprocesadores. Es una red de dos hilos, half duplex (un sentido a la vez) apropiada para aplicaciones de

¹ Roberto Saco, Universidad Nacional de Quilmes, Av. Calchaquí Nro. 5800, Fcio. Varela (1888), Buenos Aires, Argentina, rsaco@unq.edu.ar

² Anibal Zanini, Universidad Nacional de Quilmes, Av. Calchaquí Nro. 5800, Fcio. Varela (1888), Buenos Aires, Argentina, azanini@unq.edu.ar

³ Carlos A. Godfrid, Universidad de Buenos Aires, Av. Paseo Colón 850, Capital Federal (1063), Buenos Aires, Argentina, cgodfri@fi.uba.ar

alta velocidad utilizando mensajes cortos. El protocolo CAN utiliza identificadores de mensajes (ID) que pueden ser asignados en diferentes categorías de información de control [6]. Estos ID son a su vez indicadores de la prioridad de cada mensaje. En caso de disputa de la red, los mensajes de menor ID prevalecen respecto a los mensajes de mayor ID. Esto hace que la red CAN sea adecuada para su utilización en control en tiempo real. La velocidad de comunicación puede llegar hasta 1 Mbits/seg en distancias de aproximadamente 40 metros. El protocolo es ampliamente utilizado en aplicaciones sobre vehículos aunque su uso se está ampliando rápidamente a aplicaciones de todo tipo debido a su bajo costo. Los puntos a destacar de las redes CAN son los siguientes:

- Capacidad de control en tiempo real, gracias a la posibilidad de asignar prioridades a sus mensajes y a la no destrucción de mensajes en el caso de una disputa del bus.
- Capacidad de comunicación multi-maestro. Cualquier nodo puede iniciar una comunicación en forma independiente en cualquier momento.
- Bajo costo de implementación debido a la disponibilidad en una gran cantidad de microcontroladores de diferentes fabricantes como Intel, Motorola, Philips, Dallas, etc.
- Posee diferentes mecanismos de detección de errores que la hace muy robusta e ideal para trabajar en ambientes ruidosos.
- Es fácil de configurar y modificar.

Estos son los puntos que se tomaron en cuenta a la hora de elegir una red para interconectar los procesos de laboratorio con las PC's. Los beneficios de interconectar plantas y PCs con las red CAN en los laboratorios son enormes. Entre los puntos que podemos resaltar de estos beneficios son:

- Un protocolo de comunicación estándar simplifica y reduce los costos al conectar diferentes plantas y subsistemas de varios vendedores en una red común.
- El algoritmo de control es realizado por la CPU o DSP en la PC dejando tareas como adquisición, protección, y generación de señales al nodo. De esta manera, la CPU o DSP encargado del algoritmo de control tiene más tiempo para tareas específicas de control.
- La red CAN, multiplexa todos los mensajes mediante un cable de sólo 2 conductores, esto elimina la tediosa tarea del conexionado punto a punto de todas las señales entre la PC y la planta, ahorrando tiempo, dinero y facilitando las tareas a la hora de la interconexión.

HARDWARE

Existen varias maneras de implementar el hardware necesario, con capacidad de comunicación CAN. Si se consulta la documentación on-line de Mathwork se pueden encontrar soluciones para varios sistemas embebidos

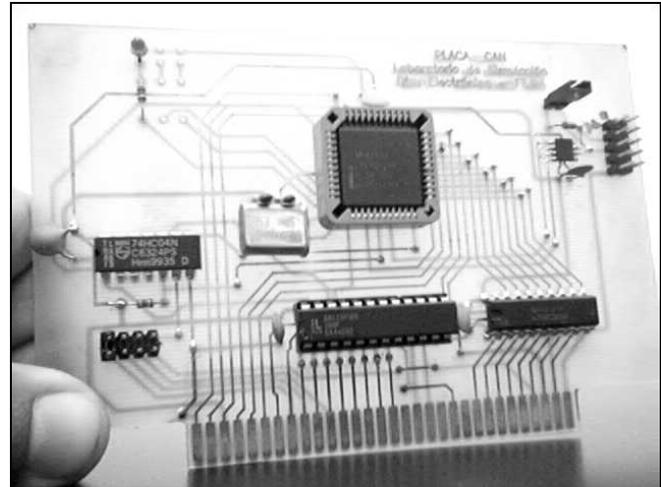


FIGURA. 1
PLACA DE RED CAN PARA PC

posibles. En nuestro caso la comunicación CAN con las PC's se logra mediante una placa ISA en un slot de la PC (Figura 1). Esta placa contiene un controlador CAN AN82527 de Intel, el cual maneja completamente todos los detalles del protocolo CAN. Para la capa física de la red se utilizó un integrado UC5350 que cumple el estándar ISO11898 y maneja velocidades de hasta 1 Mbits/segundo.

Las estaciones remotas (nodos) fueron realizados con el mismo controlador CAN AN82527. Además los nodos poseen un microcontrolador MC68HC11 que se encarga de las tareas específicas de adquisición como conversiones A/D, D/A, generación de PWM (Modulación de Ancho de Pulso), sistemas de protección de la planta y supervisión de bajo nivel. El microcontrolador se comunica con el controlador CAN para su configuración, recepción y transmisión de mensajes. Cada nodo además maneja la capa física de la comunicación con el estándar ISO11898.

SOFTWARE

Todo el software de simulación utilizado pertenece al entorno MATLAB, herramienta de uso generalizado en los cursos y proyectos de control. Se describen a continuación dos componentes importantes de MATLAB [7] utilizados para la simulación y control en tiempo real de los sistemas y el hardware que se empleó para el control de los ejemplos.

Simulink

En términos generales, Simulink es un paquete de software que permite modelar, simular, y analizar sistemas dinámicos. Soporta sistemas lineales como no lineales, modelos en tiempo continuo, discretos e híbridos. Para diseñar los modelos, Simulink provee una interfaz gráfica (GUI) que permite construir los modelos como diagramas de bloques, de la forma habitual en que se plantea el diseño de sistemas en el ámbito del control automático. Simulink incluye un conjunto de bloques de librería. Algunos de estos bloques están incluidos en el paquete principal, otros son "toolbox"

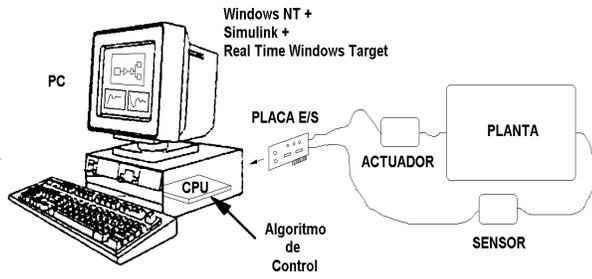


FIGURA. 2

CONTROL EN TIEMPO REAL CON EL REAL TIME WINDOWS TARGET.

específicos de determinados temas, que enriquecen la capacidad de Simulink [8]. Un vez definido un modelo, se puede simular eligiendo diferentes métodos de integración y observar los resultados de la simulación mediante bloques especiales, que muestran las variables del modelo en función del tiempo, de igual forma que un osciloscopio.

RTW

El RTW (Real Time Workshop) genera código "C" directamente del modelo realizado en Simulink. También puede generar en forma automática programas que pueden ser ejecutados en plataformas de tiempo real sin la necesidad de que estén instalados el MATLAB o el Simulink. Esto permite la ejecución de modelos continuos, discretos e híbridos en una variedad de computadoras, e interactuar con el hardware para la adquisición y generación de señales. De esta forma, se pueden construir fácilmente, no sólo controladores, sino también se pueden simular plantas que están operando en tiempo real, de modo que las señales de entrada y salida de dicha simulación, son una aproximación de las señales de la planta real.

Una opción del RTW es el Real Time Windows Target [9]. Este paquete permite utilizar una PC con sistema operativo Windows 95/98/NT como plataforma para las simulaciones en tiempo real (Figura 2). Una serie de bloques de librería para placas de adquisición son incluidas en el mismo. Si se tiene alguna de estas placas, la interfaz al hardware es directa. En caso contrario, se puede escribir en código "C" la interfaz específica para la placa hardware por medio de una S-Function [10]. Las S-Functions son bloques dinámicos del Simulink, que cumplen distintas funciones en una simulación. Entre ellas, tareas de entrada y salida, si se las programa en lenguaje "C".

DRIVER DE LA PLACA CAN

Para permitir la interfaz entre el algoritmo realizado en Simulink y la placa CAN se utilizó una S-Function. Esta función realiza la configuración del CI AN82527 así como la configuración de los ID de mensajes, su recepción y transmisión. Los detalles de la configuración y manejo de mensajes son transparentes para el usuario, el cual sólo

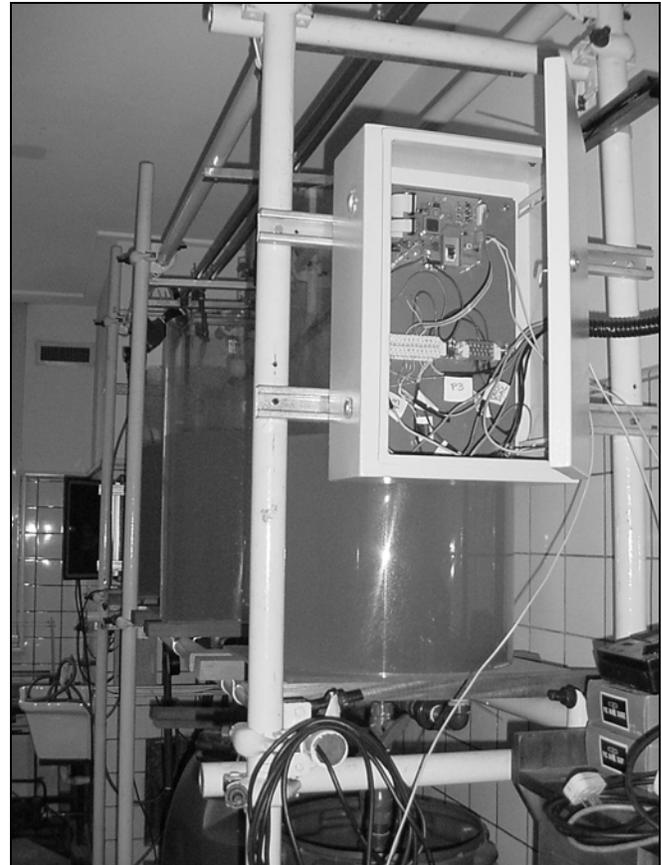


FIGURA. 3

PLANTA PARA CONTROL DE NIVEL DE LÍQUIDOS Y NODO CAN

utiliza un bloque S-Function configurado como entrada o salida de la planta. Desde Simulink, los detalles de la configuración y manejo de mensajes es transparente para el usuario, el cual sólo utiliza el bloque S-Function como entradas o salidas de las señales de la planta.

LAS APLICACIONES EXPERIMENTALES

Hasta el momento se han integrado dos aplicaciones con la red CAN. Una primera aplicación es una planta experimental para control de nivel de líquidos (Figura 3). Consiste en tres tanques que descargan su contenido (agua) a un depósito. Los tanques están interconectados y provisto de válvulas manuales para regular su descarga o intercambio de líquido entre ellos. Cada tanque posee un transmisor de nivel de líquido realizado con un sensor de presión diferencial. Una bomba centrífuga aspira el agua del depósito y la envía a uno de los tanques a través de una válvula neumática. El experimento consiste en controlar el nivel de líquido del primer, segundo o tercer tanque mediante el control de la válvula neumática. El nodo CAN, se encarga de adquirir las señales de los sensores de nivel y generar la señal para la

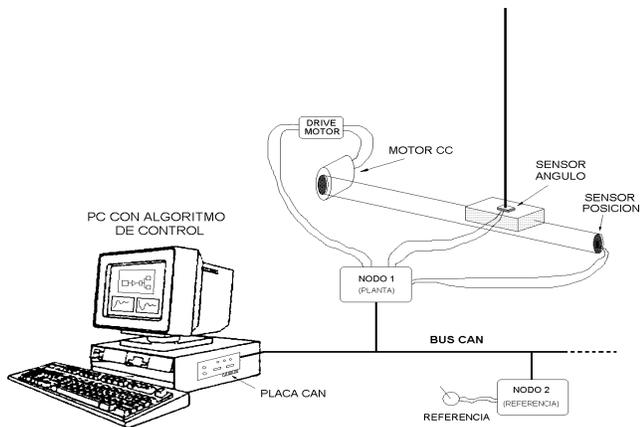


FIGURA. 4

CONTROL DE UN PÉNDULO INVERTIDO A TRAVÉS DE LA RED CAN.

válvula neumática. Por otro lado, el nodo se encarga de comunicar estas señales con la PC a través de la red CAN.

La segunda planta es un péndulo invertido el cual debe ser mantenido en posición vertical mediante un adecuado algoritmo de control (Figura 4). El nodo se encarga de adquirir la señales del ángulo del péndulo y la posición del carro así como la generación de la señal de modulación de ancho de pulso (PWM) utilizada para controlar el motor que mueve el carro. Este nodo envía y recibe estas señales a la PC a través de la red CAN.

Las dos plantas del laboratorio están conectadas a dos PC mediante una red CAN común a ambas. Cada PC puede controlar indistintamente a cualquier planta. Otra configuración posible es que una de las PC controle una planta y la otra PC utilizarla como monitor del proceso y adquisición de datos o como ajuste de los parámetros del algoritmo de control.

CONCLUSIONES

Los trabajos realizados demuestran la conveniencia de disponer de una interfaz normalizada para vincular desarrollos experimentales.

En el caso presentado, un proceso de laboratorio para control de nivel de líquidos y un sistema mecánico para estabilizar un péndulo invertido, fueron controlados usando la interfaz CAN propuesta.

Los resultados demuestran que la red se comporta en forma apropiada para el control de ambos experimentos.

En el caso de la operación de ambas plantas en forma simultánea el tráfico en la red resultó transparente para el control de las mismas. La facilidad a la hora de la conexión de la PC's con los experimentos justifica ampliamente el uso de la red. La posibilidad de que en un futuro todas las plantas del laboratorio de control, así como de otros laboratorios, pueden ser conectadas bajo el mismo protocolo común CAN nos alienta a seguir profundizando aún más en el tema.

REFERENCIAS

- [1] P. Antsaklis, T. Basar, R. DeCarlo, N. H. McClamroch, M. Spong, and S. Yurkovich, "Report on the NSF/CSS Workshop on New Directions in Control Engineering Education", *IEEE Control System Mag.*, Vol, No 19., Oct. 1999, pp. 53-58
- [2] W. A. Moreno, J. Leffew, O. Cárdenas, N. Ramos, N. Pernalet, *et al.*, "Integrating Instructional Technology Methodologies in a State of The Art Industrial Control Laboratory", *ASEE/IEEE Frontier in Education Conference*, F1D-8, October 2000
- [3] V. Kapila, M. S. de Queiroz, and A. Tzes, "Interactive Control Education with Virtual Presence on the Web", in *Proc. American Control Conference*, June 2000, pp. 3985-3990
- [4] W. E. Dixon, D. M. Dawson, B. T. Costic, and M. S. de Queiroz, "A MATLAB-Based Control Systems Laboratory Experience for Undergraduate Students: Toward Standardization and Shared Resources", *IEEE Trans. On Education*, Vol, No 3., August 2002, pp. 218-226
- [5] R. Saco, E. Pires, and C. Godfrid, "Real Time Controlled Laboratory Plan for Control Education", *ASEE/IEEE Frontier in Education Conference*. Session T2D, Boston, November 2002, pp. 12-16
- [6] P. Pedreiras, L. Almeida, "Asynchronous Communication on FTT-CAN: Experiments results", *4th IFAC International Conference on Fieldbus System and their Applications*. pp. 113-120
- [7] D. C. Hanselman and B. Littlefield, "Mastering MATLAB 6: A Comprehensive Tutorial and Reference", Prentice-Hall, 2001
- [8] Mathworks, "Real Time Workshop for use with Simulink", *Mathwork User's Guide Ver 2.1*, Massachusetts, 1999.
- [9] MathWorks, "Run Simulink and Stateflow models on a PC in Real Time", <http://www.mathworks.com/products/rtwt>
- [10] MathWorks, "Writing S-Functions", <http://www.mathworks.com/access/helpdesk/help/toolbox/simulink/simulink.shtml>