

DISEÑO DE UN PROTOTIPO PARA EL CONTROL DE SISTEMAS DINÁMICOS CON FPGA Y DSPIC

Álvaro Enrique Ávila¹, Mauricio Pérez¹, Juan Camilo Gómez¹

¹Ingenieros en Instrumentación y Control, Politécnico Colombiano Jaime Isaza Cadavid, Carrera 48 No 7-115, El Poblado Medellín- Colombia. Email: alvaroavila4@gmail.com, mparcila@gmail.com, juancamilogc@ingenieros.com

RESUMEN

Este artículo describe el uso de la tarjeta FPGA y el dsPIC, como elementos para controlar sistemas dinámicos. La propuesta presentada aquí, es mostrar las ventajas y desventajas usando este tipo de dispositivos electrónicos, con relación a desarrollar nuevas herramientas de control industrial, las cuales pueden reemplazar los elementos convencionales de control comúnmente usados. Por lo tanto, fue desarrollado un controlador difuso proporcional en el FPGA, compuesto de siete particiones, para controlar la velocidad de un motor DC, y se utilizó el dsPIC[®] para el filtrado y envío de datos a un computador, usando comunicación RS-232 para visualizar el comportamiento del sistema en una interfaz grafica desarrollada en LabVIEW[™] v. 8.6. No obstante, el modulo de control fue diseñado con el objetivo de controlar y recibir varios tipos de variables, como lo son: velocidad, presión, nivel, flujo, o temperatura entre otras, ofreciendo la posibilidad de diferentes diseños de acuerdo a las necesidades del usuario.

Palabras clave: FPGA, dsPIC[®], lógica difusa, controlador proporcional, sistema dinámico.

Recibido: 4 de Octubre de 2009. Aceptado: 6 de Diciembre de 2009
Received: October 4, 2009 Accepted: December 6, 2009

FPGA AND DSPIC IMPLEMENTATION IN THE CONTROL OF DYNAMIC SYSTEMS

ABSTRACT

This paper describes the use of FPGA and dsPIC, as control elements in the way of controlling dynamic systems. The approach presented here, is to show the advantages and disadvantages using this kind of electronic devices, regarding to develop new industrial control tools, which can replace the conventional control mechanism commonly used. Therefore, it was developed a Proportional fuzzy logic controller in the FPGA, made of seven partitions, in order to control the speed of a DC motor, and a dsPIC[®] was used to filter and send the information to a personal computer, this was made using RS-232 communication to visualize the system performance in a graphic interface developed in LabVIEW[™] v. 8.6. Nevertheless, the control module was built with the objective to control and receive several kinds of variables, like: velocity, pressure, level, flow, or temperature, offering the possibility of different designs according to the user needs.

Keywords: FPGA, dsPIC[®], fuzzy logic, proportional controller, dynamic system.

1. INTRODUCCIÓN

La implementación de nuevas tecnologías a nivel industrial se hace cada día más necesaria, gracias a los estándares de calidad que tienen que cumplir los procesos y al alto nivel de competencia que existe en el medio. Debido a esto, se planteó el uso del FPGA y el dsPIC®, como elementos que permiten la implementación de técnicas de control convencionales, avanzadas, robustas; difíciles y costosas de implementar, partiendo del mundo análogo.

Por lo tanto, empleando dispositivos que manejan señales digitales como las FPGA (dispositivo donde se realiza el control del sistema), y análogas-digitales como los dsPIC® (dispositivo donde se realiza la conversión A/D, filtrado, y transmisión / recepción de las señales que se manejan en el sistema); al fusionar el trabajo de ambos dispositivos, se puede obtener un control digital robusto, donde el procesamiento de la información demanda tiempo, costos, y velocidad de respuesta, lo que nos lleva a justificar el uso de dichos elementos.

Para el trabajo con señales análogas, es necesario, dentro de lo posible tener una buena lectura con el sensor del proceso, por lo cual se implementan filtros a dichas señales, pues a pesar de tener buenos sensores en las plantas, las perturbaciones y errores en las lecturas de la variable, siempre van a estar presentes en los valores entregados por el sensor. En consecuencia, el control del proceso depende en gran parte a la veracidad de los datos, y al comportamiento o dinámica del sistema analizado.

En este proyecto, se implementó un Control Proporcional Difuso para controlar la velocidad de un motor de DC, aprovechando las ventajas del manejo digital del FPGA, y la versatilidad del dsPIC 30F3014 en el manejo de señales A/D.

2. DISEÑO DEL FILTRO FIR

La implementación de filtros para la correcta obtención de una señal o variable análoga, es necesaria al utilizarla como una señal que permita ejercer una acción en el elemento final de control (EFC), el cual, en última instancia es el que ayuda a obtener un equilibrio del sistema analizado. En

este proyecto se diseñó un filtro FIR Pasa Bajas, con el objetivo de filtrar la variable controlada del sistema.

2.1 Implementación del filtro FIR

Los filtros de respuesta finita al impulso FIR (*Finite Impulse Response*), tienen la ventaja de ser utilizados con facilidad y poseer fase lineal para una respuesta al impulso par o impar, propiedad conocida como simetría. [1]

Estos filtros tienen todos los polos en el origen, por lo que son estables, los ceros se presentan en pares de recíprocos como se ve en la figura 1; si el filtro se diseña para tener fase lineal, como lo es en este caso. También, se caracterizan básicamente por requerir de elevados órdenes a la hora de su creación, lo que implica un mayor procesamiento y circuitos más complejos para su realización; pero a la vez estos son estables.

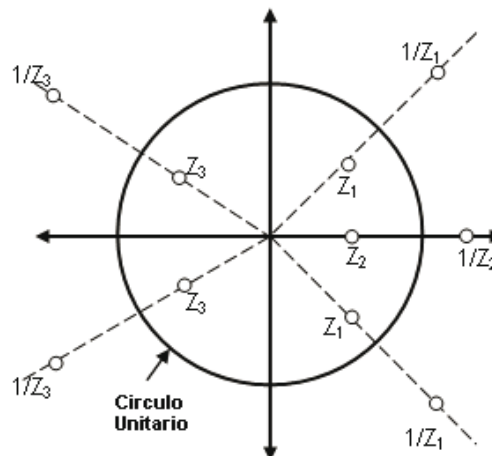


Fig. 1. Localización simétrica de los ceros para un filtro FIR de fase lineal. [2]

2.2 Selección del tipo de filtro y ventana

El tipo de filtro implementado, fue un filtro FIR pasa baja (ver figura 2), de orden 20, frecuencia de muestreo de 200kHz y frecuencia de corte de 14kHz; en este caso elegido por su efectividad en el filtrado de frecuencias bajas, además de su buena linealidad, velocidad de respuesta, estabilidad y causalidad, características que se debieron tener muy presentes en este proyecto.

Además, es preciso anotar, que la señal filtrada, debe ofrecer una buena linealidad y exactitud, puesto que esta se encuentra en el lazo de control del sistema.

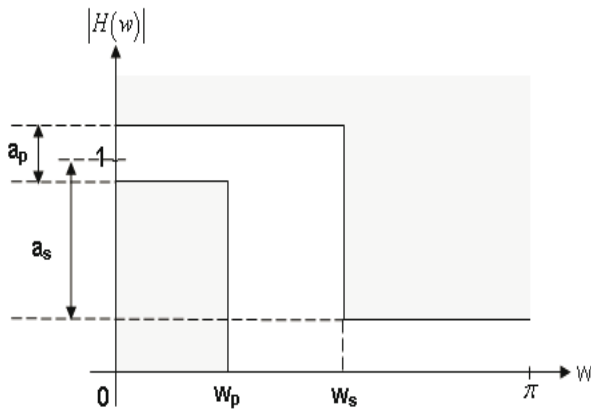


Fig. 2. Filtro pasa baja. [2]

En la figura 2:

W_p , frecuencia de corte normalizada en la banda de paso.

W_s , frecuencia de corte normalizada en la banda de parada.

a_p , rizado máximo en la banda de paso [dB].

a_s , atenuación mínima en la banda de parada [dB].

En este trabajo se hizo referencia a la técnica de ventana, debido a su efectividad, la posibilidad de observar el comportamiento del filtro, y su fácil implementación a nivel de software. [2]

Dicha técnica, se basa en la respuesta de un filtro al aplicarle la ventana deseada por medio de una multiplicación de sus ecuaciones; la ventana hace que en el filtro real diseñado se tengan menos variaciones de transición o supresión y con esto se logre un filtrado más efectivo de la señal voltaje, variable con la cual se controla la velocidad del sistema. La ventana utilizada fue la ventana de Blackman. La ecuación 1, hace referencia a la respuesta en frecuencia del filtro.

$$h(n) = w(n) * h_d(n); 0 \leq n \leq 20 \quad (1)$$

Donde $w(n)$ son los coeficientes de la ventana de Blackman, y $h_d(n)$ la respuesta en frecuencia del filtro pasa baja. La ecuación (1), se uso para determinar los coeficientes del filtro, los cuales son:

$$h(n) = \{0.0000, -0.0002, -0.0006, 0.0003, 0.0051, 0.0175, 0.0398, 0.0708, 0.1041, 0.1301, 0.1400, 0.1301, 0.1041, 0.0708, 0.0398, 0.0175, 0.0051, 0.0003, -0.0006, -0.0002, 0.0000\}$$

Al simular la respuesta del filtro con los coeficientes calculados se obtuvo la figura 3, que muestra la ubicación de los polos y ceros de este, al igual que su respuesta ante la señal escalón unitario, (ver figura 4). Dicha señal se uso en el análisis del sistema propuesto. Así, luego del diseño del filtro y de verificar su estabilidad y correcto comportamiento, se procedió a verificar y validar su comportamiento en el sistema real, tal y como se muestra en la figura 5.

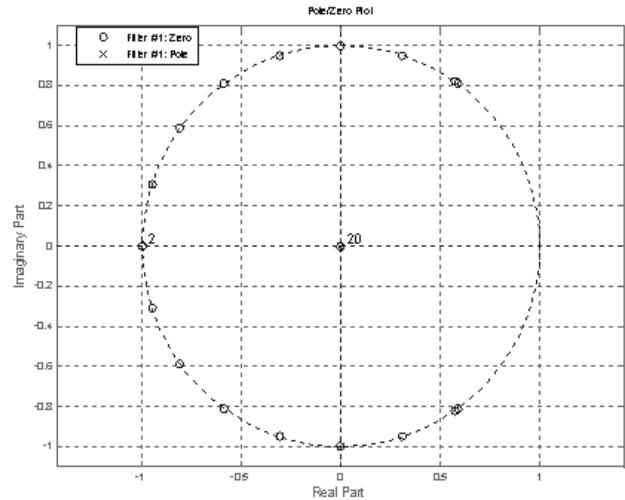


Fig. 3. Localización de polos y ceros del filtro. [3]

Como se puede observar en la respuesta del filtro ante el escalón, a nivel de simulación, (ver figura 4) y a nivel real, (ver figura 5), ambas muestran el mismo comportamiento, validando el diseño del filtro implementado.

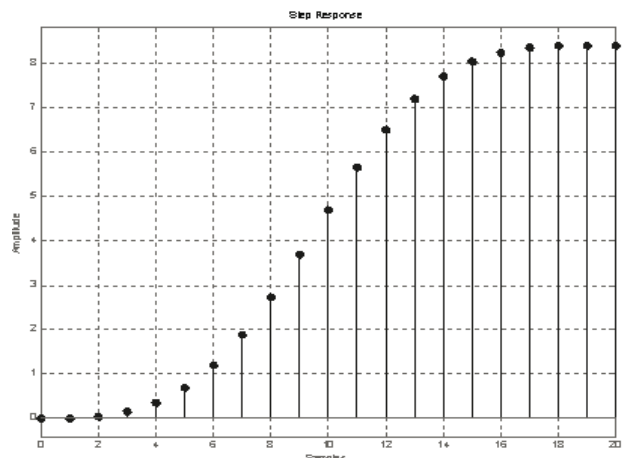


Fig. 4. Respuesta del filtro ante el escalón. [3]

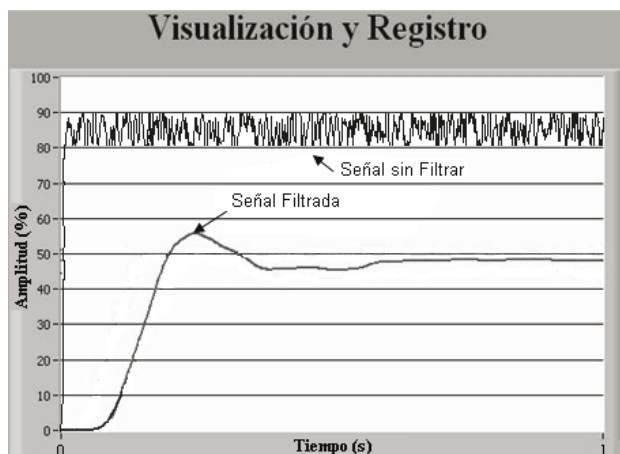


Fig. 5. Respuesta de la señal ante el filtro FIR pasa baja diseñado. [4]

3. APLICACIÓN DE LA TÉCNICA DE CONTROL DIFUSA EN FPGA

Cuando se habla del diseño de una técnica de control, siempre surge la necesidad de implementar hardware. Es ahí, cuando las variadas posibilidades se reducen al analizar la relación de costo, flexibilidad, prestaciones y complejidad de los dispositivos que se puedan usar con este fin.

Entre los dispositivos lógicos programables más recientes y avanzados están los FPGA's (*Field Programmable Gate Array*), circuitos con componentes micro electrónicos de última generación que permiten la capacidad de 5000 a 10 millones de compuertas en un solo chip. No obstante, esta característica abre la posibilidad de implementar múltiples diseños de sistemas físicos mediante FPGA's. [5]

La arquitectura de un FPGA, consiste en arreglos bidimensionales de varias celdas lógicas que se comunican mediante canales de interconexión, verticales y horizontales. Además, un FPGA contiene gran cantidad de celdas lógicas lo que hace posible la implementación de grandes funciones con la utilización de varias celdas lógicas interconectadas en forma de cascada. Esto hace factible la realización de sistemas amplios y complejos dentro del dispositivo.

La estructura básica del FPGA junto con todos sus componentes mencionados anteriormente se muestran en la figura 6. [6]

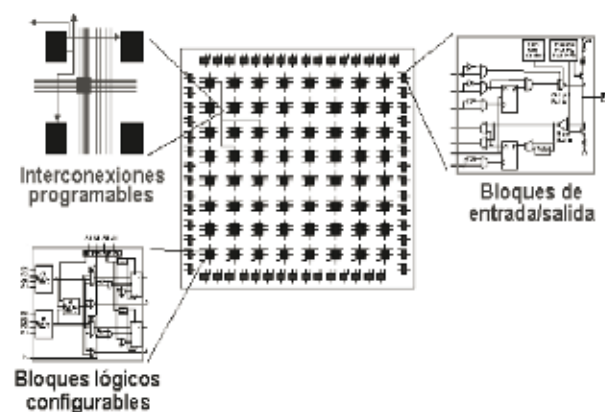


Fig. 6. Arquitectura básica del FPGA. [6]

3.1 Descripción general del funcionamiento del sistema

Ahora bien, la descripción paso a paso de este sistema es el siguiente: por medio de la interfaz de entradas y salidas de la FPGA, se envían los valores de la señal manipulada, set-point (para control), y la señal que se desea visualizar (variable). Estos parámetros son recibidos por el software programado en VHDL, específicamente para el control, el cual se encarga de generar la señal de control.

El divisor de frecuencia que está conectado a la señal del reloj de 50 MHz del FPGA, genera una señal de reloj a la frecuencia con la que funcionará el contador PWM. Por otro lado, el contador PWM, al recibir la señal del reloj del divisor de frecuencia, incrementa su cuenta de uno en uno empezando de 0 hasta 1023, similar a la generación de una señal rampa con un periodo de valor 1023, la cual será comparada constantemente con la señal de control, dando como salida una señal modulada en PWM correspondiente a la señal de control, siendo esto un convertidor digital-análogo básico para dicha señal. [7]

En la figura 7, se muestra mediante un diagrama de flujo, el proceso secuencial que se realiza de forma infinita dentro del FPGA, para el control del proceso.

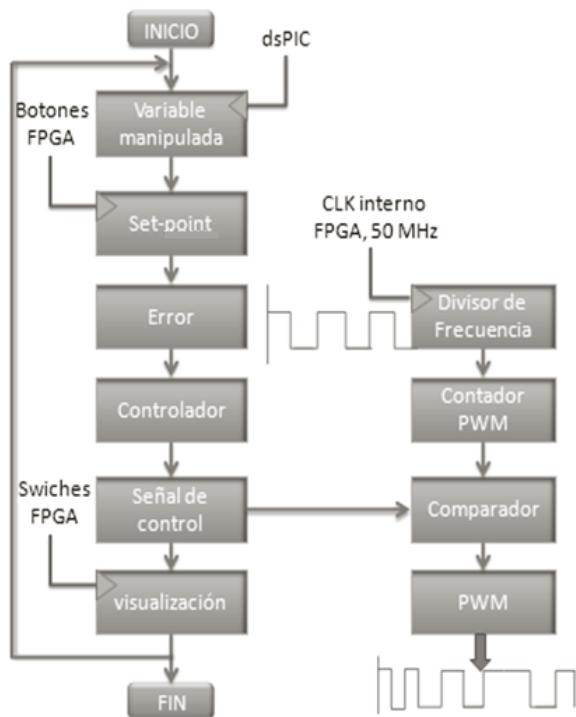


Fig. 7. Diagrama de flujo del programa principal. [7]

3.2 Simulación y validación del controlador difuso

En este proyecto se aprovecharon las ventajas tanto de velocidad de ejecución como de implementación estructurada de hardware, ofrecidas por el FPGA, lo que permitió realizar un Control Proporcional Difuso de 7 particiones tipo Mamdani, con el propósito de controlar la velocidad del sistema motor de DC, mediante el uso de la tarjeta FPGA Spartan3 XC3S200.

Para el diseño del controlador mencionado, fue necesario modelar el sistema dinámico (motor de DC), realizando la respectiva adquisición de datos del sistema, y empleando la interfaz *Control Digital v. 1.0.0*, desarrollada en MATLAB® R2008a v. 7.6.0, para analizar su comportamiento y características intrínsecas.

La figura 8, muestra gráficamente la validación del modelo identificado del sistema, mediante la interfaz *Control Digital v. 1.0.0* desarrollada en el software MATLAB® R2008a v. 7.6.0 para modelación y simulación de sistemas.

Aunque gráficamente ambos comportamientos solo se asemejan en la parte de levantamiento del sistema, y en la estabilización del mismo después

de cierto tiempo; esta identificación sin embargo, nos permitió obtener un buen control del mismo.

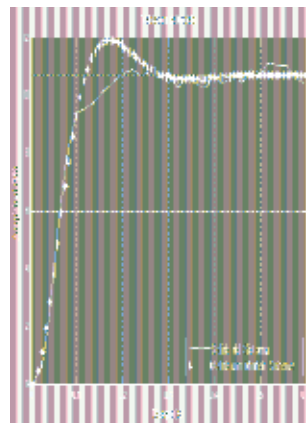


Fig. 8. Validación del modelo. [3]

Así, a pesar de la diferencia entre el pequeño sobreimpulso que se muestra en la modelación y la salida del sistema, no afectó desde el modelamiento el buen desempeño del controlador diseñado para aplicarlo al sistema real.

La ecuación 2, representa la función de transferencia que describe la dinámica del sistema, aproximando este mediante la interfaz mencionada, a un sistema de segundo orden.

$$G(s) = \frac{549}{s^2 + 25.8s + 514} \quad (2)$$

El diagrama de bloques que se presenta en la figura 9, muestra la configuración del lazo de control implementado en la herramienta de simulación, Simulink de MATLAB® R2008a v. 7.6.0, con el fin de obtener del sistema analizado la respuesta ante la señal escalón simulada, que permitió confirmar y comparar los resultados simulados con los obtenidos en el control de la planta real.

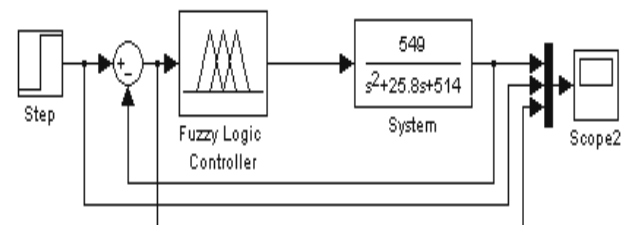


Fig. 9 Diagrama de bloques del sistema. [3]

Por otra parte, se implementaron en el *tool-box* de lógica difusa de MATLAB® R2008a v. 7.6.0, las reglas y particiones necesarias para controlar el sistema, las cuales en este caso fueron 7 como se ve en la figura 10, y por lo tanto, fueron cargadas en el bloque (*Fuzzy Logic Controller*), que muestra la figura 9, al igual que la función de transferencia obtenida, ecuación (2); obteniendo como resultado la gráfica simulada de la respuesta del sistema ante un escalón unitario, la cual se describe en la figura 11.



Fig. 10. Editor de función de membresía. Variables, (Error, Salida del sistema). [3]

Luego de observar la simulación de la respuesta del sistema ante el escalón aplicado (ver figura 11.), se puede comparar está, con la respuesta obtenida al aplicar el control a la planta real, donde se pueden analizar ambos comportamientos del control obtenido. [8]

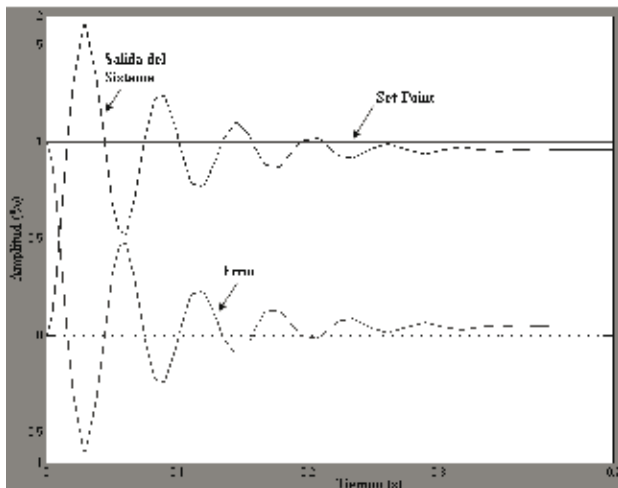


Fig. 11. Respuesta simulada del sistema ante el escalón unitario. [3]

En la figura 12, se puede ver el comportamiento real del sistema ante el control aplicado. Aunque, las oscilaciones que se observan en la figura 11, son mayores que las que se presentan en el sistema simulado, la dinámica del sistema es controlable o al menos estable en ambos casos.

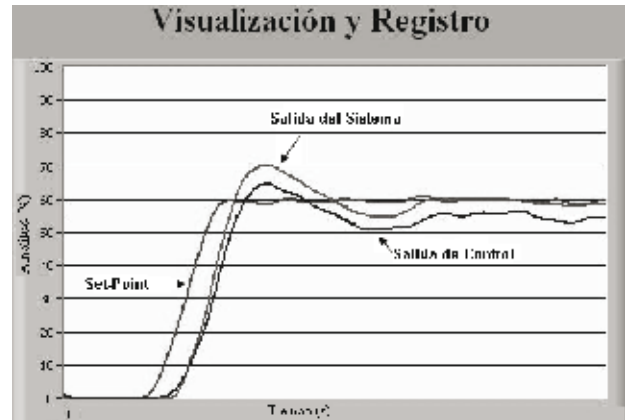


Fig. 12. Respuesta del sistema ante el escalón. [4]

4. DESARROLLO DE LA INTERFAZ GRÁFICA EN LABVIEW

La interfaz que se desarrolló en el software LabVIEW™ v. 8.6, se estructuró de tal forma que pueda ser modificada en futuras aplicaciones por otras personas, brindando la oportunidad de retomar o reutilizar parte de la interfaz diseñada, e implementar nuevos diagramas para procesos con variables de diferentes naturalezas. Todo esto, se hizo con el fin de generar desde el punto de vista del control y la instrumentación, una interfaz amigable, fácil de interpretar, y sobre todo que permitiera realizar mejoras y adaptaciones simples, a futuras necesidades acordes a los nuevos sistemas de control que se puedan generar con el transcurrir del tiempo.

4.1 Configuración del bloque VISA (Comunicación Serial)

Por medio de este bloque se ingresan los parámetros necesarios para realizar la recepción y transmisión de datos entre la interfaz gráfica creada en LabVIEW™ v. 8.6, y el dsPIC®, como se muestra en la figura 13. Por lo tanto, en este punto de la comunicación entre ambos elementos, es necesario establecer que la configuración realizada en la interfaz sea la misma en el dsPIC® 30F3014 mediante código, con el fin de no generar errores

en la escritura y lectura de los datos de entrada y salida. [4]



Fig. 13. Configuración del puerto serial, bloque VISA. [4]

Los demás bloques, sub diagramas, y funciones implementadas en la interfaz, se programaron y configuraron de forma que se puedan manipular un número mayor de variables, ofreciendo la posibilidad de realizar modificaciones en el mismo de acuerdo a las necesidades requeridas por el usuario. En la figura 14, se puede observar el ciclo realizado en la estructura desarrollada, con el fin de obtener la visualización y el registro de la(s) variable(s) que se desean monitorear.

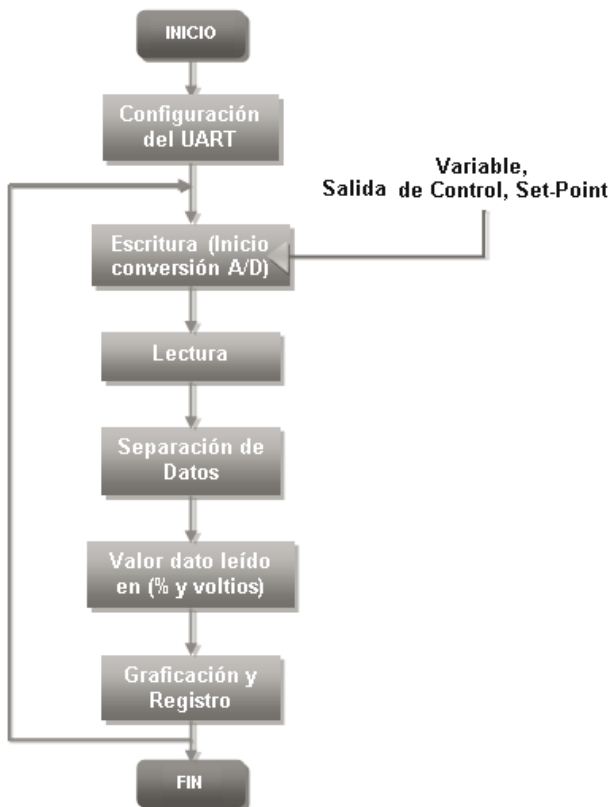


Fig. 14. Diagrama de flujo, Interfaz desarrollada en LabVIEW™ v. 8.6. [7]

Así, se permite crear entre el operario de la planta y el proceso como tal, una interface grafica, amigable, y de fácil visualización, del comportamiento y monitoreo del sistema.

El aspecto final de la interfaz desarrollada, se muestra en la figura 15, y de igual moda se aclara que dicha interfaz se puede modificar según sean las necesidades de futuras aplicación con el software acá presentado.

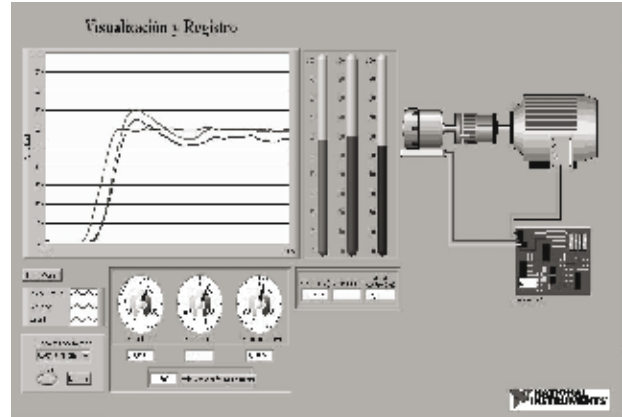


Fig. 15. Interfaz de Visualización y Registro. [4]

5. CONSTRUCCIÓN DEL MÓDULO FÍSICO DE INTERFACE ENTRE EL PROCESO REAL Y EL SOFTWARE DE REGISTRO Y VISUALIZACIÓN

En este aspecto se encuentra la descripción del hardware construido para generar la comunicación y el control del sistema analizado, brindando la posibilidad al usuario de aplicar nuevas técnicas de control a futuro, basándose en la estructura de hardware desarrollada para este proyecto.

El algoritmo o técnica de control se implementó en la tarjeta, FPGA Spartan3 XC3S200, al igual que la visualización tanto del Set-Point, como del valor de la variable leída en cada instante de tiempo. Por otro lado, en el dsPIC 30F3014 se realiza la lectura de la variable análoga, la conversión A/D, el filtrado, en formato binario a la FPGA y la transmisión por el UART al computador, empleando comunicación por RS-232. Se puede notar, que ambos dispositivos van de la mano, y que forman parte del módulo de control desarrollado, el cual, queda como plataforma experimental y de nuevas aplicaciones, haciendo de este, un modulo estándar en su diseño, con el fin de facilitar los medios y recursos

físicos a futuros usuarios, que deseen implementar nuevas técnicas de control y por que no, mejorar la técnica que se propuso en este trabajo.

5.1 Módulo de control

Este módulo está conformado por la tarjeta FPGA y el dsPIC, siendo este último el encargado de realizar la conversión A/D de la señal, enviándola luego de realizar su filtrado, en formato binario a la FPGA, dispositivo en el que se encuentra el control diseñado para el sistema. Así mismo, en la FPGA, se generan dos PWM, uno utilizado para generar el Set-Point, y el otro para generar la Salida de Control, ambas señales digitales se envían al amplificador LM358, donde se realiza mediante un circuito RC, la conversión del valor de la variable en frecuencia a un valor análogo de la misma entre 0 y 4 voltios de DC. Este esquema se puede apreciar en la figura 16.

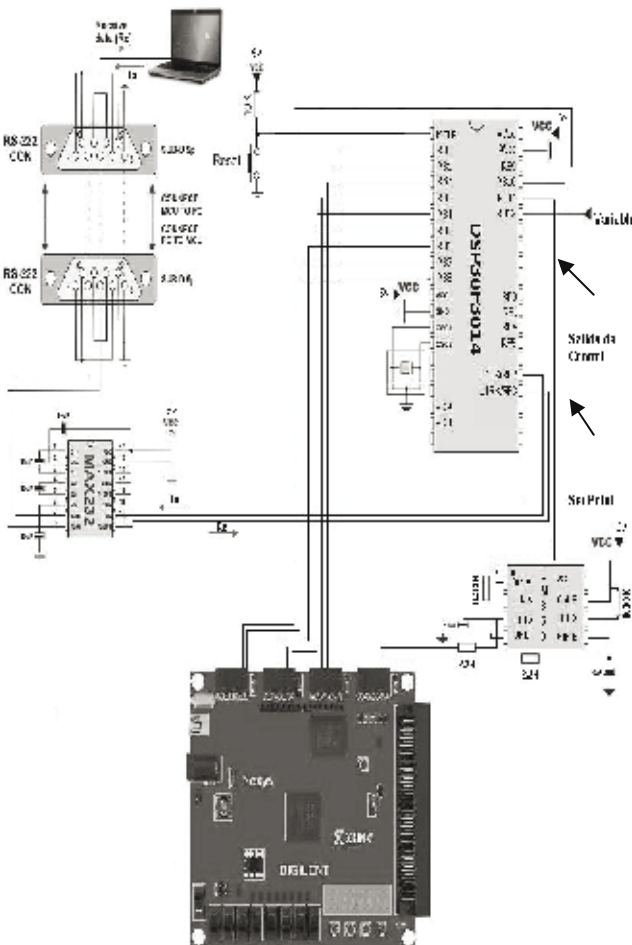


Fig. 16. Diagrama esquemático del circuito de control. [7]

Se puede observar que durante el desarrollo del proyecto, se brinda la posibilidad de diseñar nuevas técnicas de control, filtros, y modificaciones tanto a nivel de software como de hardware, lo que hace de éste, un elemento útil con relación a la implementación y análisis del comportamiento de cualquier sistema dinámico, empleando la configuración mostrada en la figura 16.

6. CONCLUSIONES

El filtro FIR diseñado cumplió con todas las expectativas propuestas, ya que su desempeño en el correcto filtrado de la variable manipulada, fue óptimo, debido a su estabilidad y fase lineal, garantizando una lectura más confiable de la variable del proceso. Este tipo de filtros, a pesar de sus grandes bondades, presentan incremento en la manipulación de los datos, lo cual es un factor a tener presente a la hora de implementarlos en otro tipo de sistemas, donde la velocidad de respuesta sea un factor fundamental en el control del proceso. De igual modo se pudo observar en la figura 12, que la velocidad de respuesta del sistema, se ve algo retardada en relación a la parte gráfica o visualización de la(s) variable(s) graficadas, lo cual no ocurre en el momento de realizar el control, pues este si responde de inmediato.

El FPGA, ofrece un alto desempeño en relación con la velocidad de procesamiento que este maneja, además, de la posibilidad de manejar grandes y complejos circuitos lógicos, gracias a los diferentes diseños y versiones ofrecidas por los fabricantes. La velocidad de procesamiento de estos dispositivos, permite realizar Controles Difusos de una forma más fácil y sencilla que en otros dispositivos de procesamiento, dando al usuario la posibilidad de crear controles robustos, los cuales a pesar de su complejidad mediante la programación lingüística que ofrece el control difuso, se pueden volver más simples a la hora de ser programados.

Las tarjetas FPGA no son genéricas, debido a que son producidas por diferentes fabricantes, por este motivo existen muchas clases, unas más avanzadas que otras; y aunque el lenguaje de programación este estandarizado por la IEEE (*The Institute of Electrical and Electronics Engineers*), las tarjetas menos avanzadas no soportan las librerías

más complejas e importantes, como lo son las de números flotantes, complejos, y reales. Por esta razón y debido a que no se poseía una mejor referencia o modelo de tarjeta, se optó por diseñar un control difuso utilizando la tarjeta XC3S200 de Xilinx, la cual es básica.

El software LabVIEW™ v. 8.6, es una de las buenas herramientas de instrumentación virtual para diseñar interfaces de monitoreo, desarrollar cálculos matemáticos, plataformas para diferentes aplicaciones tecnológicas, entre muchas más características, que facilitan, no solo el aprendizaje, sino también, el desarrollo de proyectos con grados de complejidad altos, permitiendo al usuario desarrollarlos de una forma menos tediosa y en un ambiente gráfico estructurado más ameno al programador, ayudando a este a optimizar los procesos y reduciendo tiempo de programación y desarrollo. Esta herramienta, se convierte cada día en un software más versátil, debido a su amplia gama de paquetes o librerías especializados en diferentes áreas del conocimiento, e incluso brindando la posibilidad de ejecutar códigos realizados en otros lenguajes de programación, y hasta cargar librerías que emulan dispositivos electrónicos, como por ejemplo las FPGA. Así, ayuda de un modo más didáctico y rápido a la creación de controles, interfaces de monitoreo entre otros, para que los estudiantes trabajen no un dispositivo y un software por separado, sino todo un conjunto de herramientas que se pueden intercomunicar y relacionar de una forma fácil y segura. Tanto el dsPIC® como la FPGA, brindan la posibilidad al usuario de interactuar con una gran variedad de lenguajes y librerías para su programación, minimizando el tiempo de diseño y optimizando la estructura y codificación de los diferentes programas. Además, el módulo de control diseñado con ambos dispositivos para este proyecto, brinda la posibilidad de ser modificado fácilmente, acorde a las necesidades existentes, con el fin de optimizar y automatizar procesos industriales, agroindustriales e implementar sistemas de control expertos.

La dinámica del sistema analizado, no se comporta de forma semejante tanto en la simulación, como en la implementación del control en el sistema real; sin embargo, el control obtenido es acorde a los resultados esperados, pues el sistema mostró ser estable y respondió correctamente a los cambios o

puntos de control deseados o requeridos por el usuario. Esto permite validar el correcto funcionamiento de la FPGA y el dsPIC como dispositivos de control, abriendo las puertas a nuevas aplicaciones de control con ambos elementos. Por lo tanto, dichos dispositivos si pueden ser utilizados con el fin de procesar las señales que intervengan en un proceso, ya sean estas señales de control o de las variables censadas del proceso como tal; mostrando que la combinación de estos dispositivos electrónicos, ofrecen una buena velocidad de procesamiento, al igual que un buen control y estabilidad del sistema. Una desventaja notoria en todo el proyecto, se evidencia en la parte del monitoreo de la(s) señal(es) transmitidas mediante comunicación RS-232 a un computador; pues el envío de varias señales a la vez, y además del filtrado de una o varias de estas señales, y si se le agrega que la existencia de filtros por software de ordenes elevados, hace ver la señal y la respuesta del sistema algo retardadas. Aunque, esta parte de visualización no influye directamente en el control del sistema, esto sería un aspecto a mejorar y tener en cuenta para futuras aplicaciones donde se desee tener un modo visual del comportamiento del sistema.

7. REFERENCIAS BIBLIOGRÁFICAS

- [1] Oppenheim, Willsky, Alan S., Alan V, Young, Ian T. *Signal and Systems*. Prentice Hall, 1983. Cap 6. 397-413 p.
- [2] Milivojević, Zoran. *Digital Filter Design*. Cap. 1, 2 y 3.
- [3] MATLAB® R2008a, v. 7.6. Funciones: (*System Identification, Fuzzy Logic, and Filter Design toolbox*).
- [4] LabVIEW™, v. 8.6. Manual de usuario, National Instruments.
- [5] Pong P., Chun. *FPGA Prototyping by VHDL Examples: Xilinx Spartan3*. 3 ed. Karlson, 2002. 104 p.
- [6] Manual tarjeta Nexys de Xilinx, ref. XC3S200.
- [7] Ávila López Álvaro E., Pérez Arcila Mauricio. *Implementación de FPGA y dsPIC en el control de Sistemas Dinámicos*. 2009. Politécnico Colombiano Jaime Isaza Cadavid.
- [8] Passino, Kevin M., Yurkovich, Stephen. *Fuzzy Control*. 1 ed. Addison Wesley. 1998. Cap. 4 187-191 p.