

Control de procesos industriales utilizando hardware reconfigurable

Control of industrial processes using reconfigurable hardware

Federico Fernández¹, Juan Carlos Fabero^{2,3} & Hortensia Mecha López^{2,4}

¹Departamento de Investigación y Postgrado, Facultad Politécnica U. N. San Lorenzo, Paraguay. Email: fefernandez@pol.una.py

²Facultad de Informática, Universidad Complutense de Madrid, España. Email: ³jcfabero@ucm.es; ⁴horten@ucm.es

Resumen: La realización de proyectos de automatización de procesos industriales es, desde el punto de vista del hardware y software, cada vez más complejo y sofisticado, dado que su implementación exige capacidad de cálculo; presentación gráfica; protocolos de comunicación, tanto con el exterior, con el punto donde se encuentra el proceso controlado, entre los distintos módulos componentes del sistema de control distribuido encargado de monitorizar las señales procedentes de los sensores, y modificar las variables del proceso controlado mediante actuadores. En este trabajo presentamos un sistema de control automático de procesos en tiempo real, aplicando hardware reconfigurable. Esta tecnología se presenta como una alternativa a los controladores lógicos programables (PLC), utilizados tradicionalmente en la industria, debido a su costo cada vez menor, capacidad de procesamiento paralelo, que permite mayor velocidad de proceso, capacidad de comunicación y presentación gráfica. Pero por sobre todo, el aspecto que hace diferente a las FPGAs con respecto a otras tecnologías es la capacidad de reconfiguración dinámica, ventaja que le permite que en caso de avería del controlador se pueda reconfigurar para impedir la parada del proceso controlado, factor muy importante para evitar daños y pérdidas económicas.

Palabras clave: FPGA, PLC, reconfiguración parcial, tiempo real.

Abstract: Performing projects for automation of industrial process is, in terms of software and hardware, increasingly complex and sophisticated, as their implementation requires calculation capacity; graphic presentation; communication protocols, both with the outside, with the point where the controlled process is located, between the different component modules of the distributed control system in charge of monitoring the signals coming from the sensors, and modifying the variables of the controlled process by means of actuators. In this paper we present an automatic process control system in real time, applying reconfigurable hardware. This technology is presented as an alternative to programmable logic controllers (PLCs), traditionally used in the industry, due to its increasingly low cost, parallel processing capacity, which allows greater process speed, communication capacity and graphic presentation. But above all, the aspect that makes FPGAs different from other technologies is the dynamic reconfiguration capability, an advantage that allows it to be reconfigured in case of a failure of the controller, in order to prevent a controlled process to be stopped, a very important factor to avoid damages and economic losses.

Palabras clave: FPGA, PLC, partial reconfiguration, real time.

INTRODUCCIÓN

Los controladores lógicos programables han sido por años la base de los sistemas automáticos de control de procesos industriales (Berrojo *et al.*, 2002). Entre sus ventajas están factores como su bajo costo, confiabilidad, facilidad en el manejo y la fiabilidad que concede el haber sido probado con años de éxito en su operación. Se adapta tanto a entradas y salidas digitales como analógicas. Si se elige la configuración modular puede adaptarse a situaciones en la que la arquitectura del proceso de control debe sufrir modificaciones debido al

aumento en el número de entradas y salidas que el sistema de lazo cerrado debe procesar. Otro aspecto a tener en cuenta es el hecho que en las redes de PLC (Mecafenix, 2019) tienen sus propios protocolos de comunicación, (Fielbus, Profibus), que les da la posibilidad de formar redes dedicadas para la supervisión y control de procesos industriales.

En cuanto a las formas de programación tres son los más utilizados El diagrama LADDER o escalera (Milik, 2014) fue uno de los primeros en ser utilizados, ya que se asemeja mucho a los diagramas con relevadores. Los diagramas de bloques

Recibido: 28/02/2019 Aceptado: 13/06/2019



son diagramas en su mayoría son preferidos por personas acostumbradas a trabajar con circuitos de compuertas lógicas. Finalmente el lenguaje de comandos que se compone de una serie de instrucciones que se pueden ejecutar, como sucede con los lenguajes superiores, de forma condicionada. El problema con estos dos últimos lenguajes de programación es que la sintaxis y los nemónicos son propios de cada fabricante de las PLCs por lo que se presentan problemas de migración de un fabricante a otro. De la misma manera el Diagrama Ladder es un lenguaje más intuitivo pero también presenta problemas de migración entre un fabricante y otro.

Un aspecto a tener en cuenta en todo lo referente a supervisión de procesos industriales son los Sistemas SCADA, (Supervisory Control And Data Acquisition), que fusiona lo mejor de la tecnología de control de procesos industriales, la capacidad de proceso, comunicación, almacenamiento de datos y manejo de gráficos que tienen los computadores personales con la facilidad de realizar operaciones de monitoreo y control de manera virtual e independiente de las distancias (Subbaraman *et al.*, 2010; Radulovic *et al.*, 2012).

El crecimiento experimentado en la oferta de sistemas automáticos de control hace que sean numerosos los fabricantes que se disputan un hueco en este mercado lo que hace que estos productos se encuentren continuamente evolucionando dando como resultado productos de mejores rendimientos y menor precio.

Sin embargo debido a la complejidad cada vez mayor de los procesos industriales han empezado a aparecer factores a tener en cuenta como la velocidad de operación, consumo de potencia, flexibilidad y necesidad de contar con un hardware que pueda tener robustez a la hora de la presencia de fallas en su funcionamiento.

La aparición de la tecnología basada en el hardware reconfigurable ha significado un gran salto en el diseño de dispositivos digitales, en la capacidad de reducir los tiempos de desarrollo de diseños nuevos, otorgando menor tiempo al periodo de chequeo de errores y por ende la posibilidad de ir optimi-

zando la configuración del diseño, ventajas estas que permiten que en la actualidad sean numerosas las líneas de investigación relacionadas al diseño digital utilizando dispositivos reconfigurables.

Otro aspecto a considerar es que un diseño basado en FPGA puede adaptarse a un sistema supervisor de control de procesos (Patel *et al.*, 2015), ya que indistintamente puede cumplir la función de maestro o esclavo por la versatilidad que tiene a la hora de ser diseñado para una función específica.

A diferencia de un PLC las FPGAs (Economakos & Economakos, 2019; Yoo *et al.*, 2013) tienen las siguientes ventajas tales como menor ciclo de desarrollo debido a las herramientas de simulación que tienen los entornos de desarrollo como así también poder realizar pruebas con los diseños parciales con diferentes configuraciones haciendo que el tiempo de desarrollo del producto final disminuya.

También se debe considerar la flexibilidad ya que los diseños pueden ser adaptados a los cambios en las necesidades del proceso como ser mayor número de sensores y actuadores, aumento de terminales de esclavos, agregar funciones o instrucciones en los diseños existentes, simplemente cambiando el lenguaje HDL y configurando la FPGA para el nuevo diseño. Esto es lo mismo que desarrolla una arquitectura modular en un sistema basado en PLCs (Gawali & Sharma, 2009; Chmiel *et al.*, 2016). Uno de los aspectos en la que se diferencia las FPGAs de otras tecnologías es la posibilidad de ejecutar en paralelo bloques de diseño hardware lo que proporciona un mejor rendimiento y precisión en comparación con los PLC convencionales (Economakos *et al.*, 2014). Esta característica permite que un diseño complejo se pueda dividir en diferentes tareas que se ejecutan en paralelo lo que hace que el diseño final se ejecute a mayor velocidad que un PLC convencional, ya que este último realiza todas las operaciones secuencialmente. Además de estas características lo que hace verdaderamente ventajoso a las FPGA es la capacidad de reconfiguración dinámica (Fernández & Fabero, 2014) lo que le permite cambiar su configuración sin tener que detener su funcionamiento siendo este un factor importante

en cualquier proceso de control automatizado.

Cuando se produce un error en su funcionamiento que puede afectar no sólo a la parte funcional del sistema sino también, y de manera muy importante, a la propia memoria de configuración que define su funcionalidad se hace necesaria aplicar técnicas de tolerancia a fallos que permiten que el proceso controlado no se vea afectado en su funcionamiento normal impidiendo la parada del proceso controlado con el consiguiente efecto en que esto implica. Así, cuando se detecte discrepancia entre los resultados obtenidos por alguna de las réplicas funcionales del módulo puede reconfigurarse dinámicamente la réplica afectada y continuar con la ejecución de la tarea.

A continuación se describe la estructura del trabajo. La sección II describe la arquitectura del sistema, las variables que se van a medir, el control de las señales, medio de comunicación, etc. En la sección III se describe el diseño del controlador tanto para el PLC como para la FPGA. En la sección IV se describen los resultados obtenidos para finalmente cerrar al trabajo con las conclusiones,

Arquitectura del Sistema

La arquitectura del sistema es el control del proceso de dos tanques, Figura 1, en la que uno de ellos es alimentado por dos fuentes de dos composiciones diferentes cuyo nivel es controlado por dos sensores todo nada. Los alimentadores (Q0.0 y Q0.1) empiezan a cargar el primer tanque cuando se activa el interruptor I0.0. Cuando se activa el sensor de nivel inferior I0.1 empieza a funcionar un mezclador cuya función es homogeneizar la combinación de ambas composiciones.

Una vez que el nivel del primer tanque alcanza

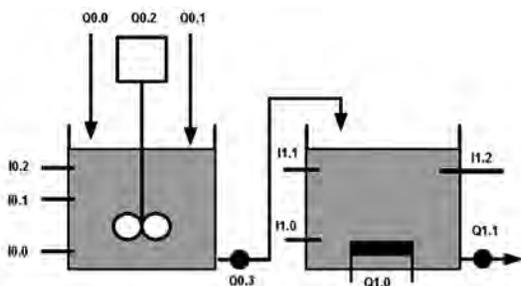


Figura 1. Proceso a controlar.

el sensor de nivel superior I0.2 se interrumpen los alimentadores Q0.0 y Q0.1 a la vez que se activa la bomba Q0.3 que conduce el producto inicial del primer tanque al segundo para la segunda etapa del proceso. En funcionamiento continuo cada vez que el nivel de primer tanque alcanza I0.2 se cerraran Q0.0 y Q0.1 abriéndose nuevamente al bajar el nivel por debajo se I0.2. repitiéndose continuamente este ciclo.

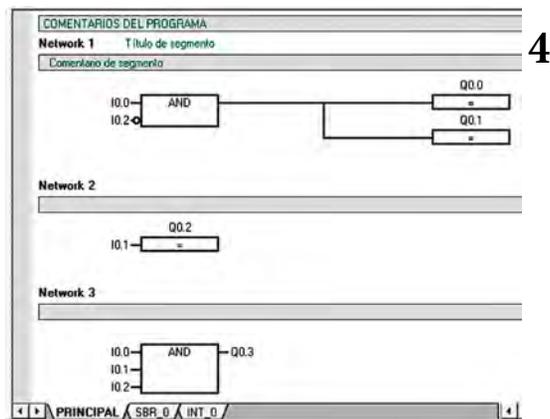
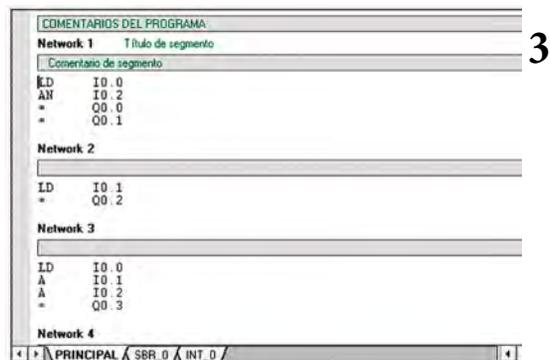
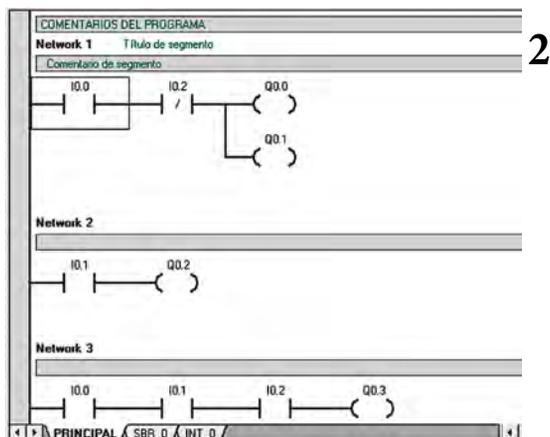
En el segundo tanque el producto procedente del primer tanque se lo debe elevar a una temperatura determinada antes de enviarlo a una etapa posterior del proceso para lo cual se utiliza un control todo nada cuyo sensor es un par bimetálico y el actuador es una resistencia industrial que suministra el calor necesario para la temperatura de referencia. Una vez estabilizada la temperatura se lo envía a una etapa posterior dentro del proceso.

Al elevarse el nivel en el segundo tanque y activar el sensor I1.0 se pone en funcionamiento el calentador Q1.0. Una vez que el nivel alcanza el interruptor I1.1 se activa la bomba Q1.1 que lleva el producto a una temperatura dada a una etapa posterior del proceso. La temperatura del segundo tanque es monitorizada por el sensor I1.2. Cuando este alcanza la temperatura de referencia se desactiva Q1.0 activándose nuevamente una vez que la temperatura del tanque cae por debajo de la temperatura de referencia repitiéndose el ciclo continuamente...

Como el objetivo de este trabajo es demostrar que se puede utilizar FPGAs para realizar operaciones de monitoreo y control industrial similares a los PLCs diseñaremos un sistema de control automático basado en el PLC S7- 200 con el entorno de desarrollo V3.2 STEP7 MicroWIN SP4 de Siemens, y posteriormente el mismo sistema de control utilizando el Nexys 4 (Diligent, 2016) con el entorno de desarrollo Vivado de Xilinx (Xilinx, 2015).

DISEÑO DEL CONTROLADOR

Primeramente desarrollamos el control utilizando el PLC S7 -200. En la Figura 2 se ve el diseño en escalera. Se muestra el esquema correspondiente



Figuras 2-4. Esquemas del controlador. 2) Esquema en escalera del controlador. 3) Esquema en lenguaje de comandos del controlador. 4) Esquema en lenguaje de compuertas del controlador.

al control de primer tanque.

En la figura 3 vemos la misma funcionalidad pero en lenguaje de comandos. En la Figura 4 vemos el controlador pero en esquema de lenguaje de compuertas.

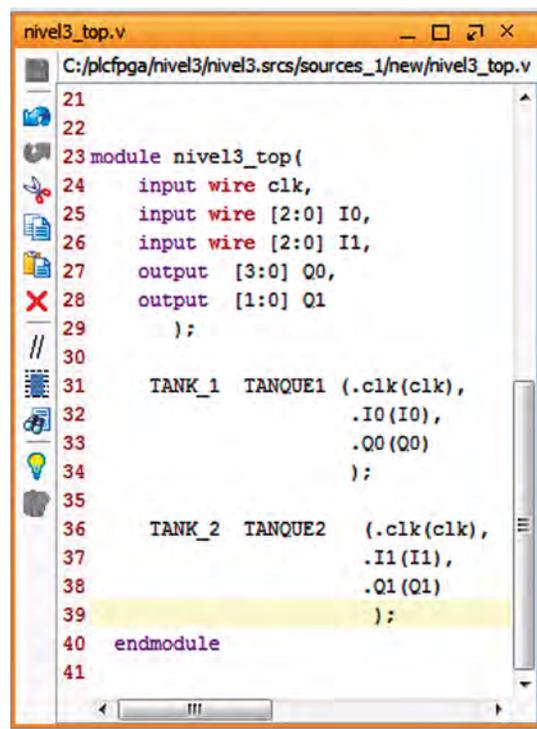


Figura 5. Programa estructurado en Verilog.

En todos los casos la respuesta del PLC ha sido satisfactoria. Siguiendo con el objetivo de nuestro trabajo hemos hecho la misma funcionalidad pero utilizando una placa basada en Artix 7 de Xilinx.

Para la programación de nuestro controlador de procesos utilizamos el lenguaje Verilog con el entorno de desarrollo Vivado de Xilinx. En la Figura 5 vemos la sintaxis del programa de control utilizado un esquema de programación estructurada en donde cada tanque es controlado por un modulo independiente y ambos módulos son coordinados por un gestor principal.

En la Figura 6 se puede observar el código para el control del primer tanque.

En la Figura 7 vemos el código para el segundo tanque. En ambos casos se han mantenido los nombres de las variables para una mejor comparación entre ambos controladores.

Finalmente se debe destacar que el entorno Vivado también tiene una funcionalidad que permite generar un esquema de diagramas esquemáticos que se puede utilizar como fuente de programación en

```

TANK_1.v
C:/plcfpga/nivel3/nivel3.srcs/sources_1/imports/new/TANK_1
22
23 module TANK_1(
24     input wire clk,
25     input wire [2:0] I0,
26     output reg [3:0] Q0
27 );
28
29 always @(posedge clk)
30     case (I0)
31         3'b000 : Q0 = 4'b0000;
32         3'b001 : Q0 = 4'b0011;
33         3'b011 : Q0 = 4'b0111;
34         3'b111 : Q0 = 4'b1100;
35
36     endcase
37 endmodule
38
    
```

6

```

TANK_2.v*
C:/plcfpga/nivel3/nivel3.srcs/sources_1/imports/tankes/TANK
22
23 module TANK_2(
24     input wire clk,
25     input wire [2:0] I1,
26     output reg [1:0] Q1
27 );
28
29 always @(posedge clk)
30
31     case (I1)
32         3'b000 : Q1 = 2'b00;
33         3'b001 : Q1 = 2'b01;
34         3'b011 : Q1 = 2'b11;
35         3'b111 : Q1 = 2'b00;
36         3'b101 : Q1 = 2'b00;
37     endcase
38 endmodule
39
    
```

7

Figura 6. Códigos para controlar los tanques. 6) Para el primer tanque. 7) Para el segundo tanque.

los sistemas automáticos de control de procesos, Figura 8.

A diferencia de otras tecnologías utilizadas en el control de procesos industriales como los PLCs o más recientemente sistemas basados en Arduino o Raspberry Pi, las FPGA pueden ser reconfigurada sin necesidad de interrumpir su funcionamiento.

Por ello su aplicación puede recuperar si un modulo de

sufre una falla puedes ser reconfigurada para reemplazar el modulo que no funciona. De allí a el

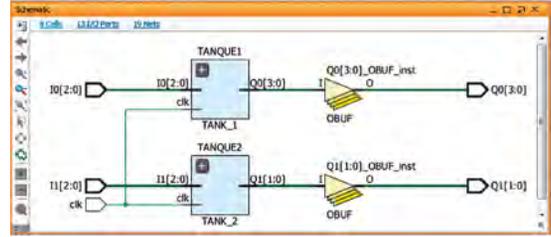


Figura 8. Esquema del controlador.

mismo puede ser el reemplazo de un controlador maestro en un sistema distribuido o un nodo de medida y control que funciona como esclavo De allí se observa nuestro esquema de control supervisado que puede funcionar de manera ininterrumpida.

RESULTADOS

Hemos probado primeramente los códigos con el PLC S7- 200 de Siemens. Las entradas y salidas hemos simulado utilizando interruptores y actuadores siendo los resultados los esperados. Seguidamente hemos realizado las pruebas con los interruptores y elementos de salida de la placa Nexys 4 siendo los resultados similares. Aunque el proceso controlado es bastante simple, el mismo puede ser ampliado a procesos más complejos que incluyan mayor número de entradas y salidas, como así también señales digitales y analógicas.

CONCLUSIONES

Nuestro proyecto esta en fase inicial de ejecución. El modelo puede ser ampliado para mas sensores y actuadores por modulo, además de aumentar el número de los mismos por módulo. En etapas posteriores nos proponemos ampliar su funcionalidad permitiendo la reconfiguración parcial dinámica aumentando la robustez del sistema ante la aparición de fallas en el controlador. Posteriormente se dotará al sistema de comunicación inalámbrica ya sea WIFI y/o Bluetooth para que los distintos módulos de medida y control puedan ser mas versátiles y el sistema de control tenga un mayor alcance y mayor conectividad con otras redes. Igualmente se puede ampliar la funcionalidad del modulo de control maestro añadiendo un modulo que funcionaría como Mirror para su funcionamiento en caso

de falla del mismo y siendo reemplazada evitando el colapso de todo el sistema controlado.

Otro aspecto importante a considerar es la posibilidad de que se pueda acceder mediante móviles y tabletas basadas en sistemas androide y usar estas terminales como puntos auxiliares para la modificación de las variables del proceso.

Finalmente exploraremos las posibilidades que ofrece la FPGA como herramienta de gestión de entornos SCADA por las facilidades que ofrece para la gestión de gráficos.

AGRADECIMIENTOS

Este trabajo fue realizado dentro del Proyecto de Investigación PINV-15-99, con financiamiento tanto del Programa PROCENCIA del Consejo Nacional de Ciencia y Tecnología (CONACyT), así como de la Facultad Politécnica de la Universidad Nacional de Asunción.

LITERATURA CITADA

- Berrojo, L., Corno, F., Entrena, L., Gonzalez, I., López, C., Sonza Reorda, M. & Squillero, G. (2002). An industrial environment for high-level fault-tolerant structures insertion and validation. *Proceedings of the 20th IEEE VLSI Test Symposium*, 1: 229-236.
- Chmiel, M., Kulisz, J., Czerwinski, R., Krzyzyk, A., Rosol, M. & Smolarek, P. (2016). An IEC 61131-3-based PLC implemented by means of an FPGA. *Microprocessors and Microsystems*, 44: 28-37.
- Digilent. (2016). *Nexys 4™ FPGA Board Reference Manual*. https://reference.digilentinc.com/_media/nexys:nexys4:nexys4_rm.pdf.
- Economakos, C. & Economakos, G. 2009. An architectural exploration framework for efficient FPGA implementation of PLC programs. *2009 17th Mediterranean Conference on Control and Automation*, 1: 1172-1177.
- Economakos, C.E., Skarpetis, M.G. & Economakos, G. (2014). Program-based and model-based PLC design environment for multicore FPGA architectures. *2014 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, 1: 726-733.
- Fernández, F. & Fabero, J.C. (2014) Allocation algorithms for dynamic partial reconfiguration on reconfigurable hardware. *3rd Conference of Computational Interdisciplinary Sciences*, 1: 402 - 404.
- Gawali, D. & Sharma, V.K. 2009. FPGA Based Micro-PLC Design Approach. *2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies*, 1: 660-663.
- Mecafenix, F. (2019). Lenguajes para programación de plc. *Ingeniería Mecafenix*. <http://www.ingmecafenix.com/automatizacion/lenguajes-programacion-plc/>.
- Milik, A. (2014). On Ladder Diagrams Compilation and Synthesis to FPGA Implemented Reconfigurable Logic Controller. *Advances in Electrical and Electronic Engineering*, 12(5): 443-451.
- Patel, D., Bhatt, J. & Trivedi, S. (2015). Programmable logic controller performance enhancement by field programmable gate array based design. *ISA Transactions*, 54: 156-168.
- Radulovic, M.I. Celebic, V.V. Kabovic, M.M. Kabovic, A.V. (2012). Realization of the teleprotection equipment interface for working with PLC equipment using FPGA logic. *2012 20th Telecommunications Forum (TELFOR) Telecommunications Forum (TELFOR)*, 1: 525-528.
- Subbaraman, S., Patil, M. & Nilkund, P. (2010). Novel integrated development environment for implementing PLC on FPGA by converting ladder diagram to synthesizable VHDL code. *2010 11th International Conference on Control Automation Robotics & Vision*, 1: 1791-1795.
- Xilinx (2015). Xilinx Artix-7 FPGAs. A new performance standard for power-limited,

cost-sensitive markets. https://www.xilinx.com/publications/prod_mktg/Artix-7-Product-Brief080113.pdf.

Yoo, J., Lee, J.H. & Lee, J.S. (2013). A research

on seamless platform change of reactor protection system from PLC to FPGA. *Nuclear Engineering and Technology*, 45(4): 477-488.