



SISTEMA BASADO EN FPGA APLICADO AL CONTROL AUTOMÁTICO DE PROCESOS INDUSTRIALES¹

Federico Fernández²

Juan Carlos Fabero³

Hortensia Mecha López⁴

Resumen

Las tecnologías utilizadas en los sistemas automáticos de control son cada vez más complejos y sofisticados por lo que su implementación permite procesos cada vez más complejos en cuanto a capacidad de cálculo, virtualización de procesos reales en tiempo real, utilización de diferentes protocolos de comunicación y el manejo de los distintos módulos componentes que forman el sistema de control distribuido encargado de monitorizar las señales procedentes de los sensores y modificar las variables del proceso controlado mediante los actuadores. El objetivo de este proyecto es la implementación de un sistema de control de procesos industriales en tiempo real aplicando hardware reconfigurable. Esta tecnología se presenta como una alternativa a los tradicionales controladores lógicos programables (PLC) utilizados en la industria debido a su costo cada vez menor, capacidad de procesamiento paralelo que permite mayor velocidad de ejecución, capacidad de comunicación y presentación gráfica, además de la capacidad de reconfiguración dinámica. Estos aspectos unidos a los protocolos actuales de comunicación inalámbrica incluidos en los estándares de redes de comunicación industriales hacen una solución para los proyectos de automatización de procesos industriales.

¹ Este trabajo fue realizado con financiamiento de los fondos del Programa PROCENCIA del CONACYT, PINV-15-99 y de la F.P. – U.N.A.

² Doctorando en Ingeniería Informática (Universidad Complutense de Madrid). Profesor Facultad Politécnica Universidad Nacional de Asunción

³ Profesor Doctor Investigador de la Facultad de Informática (Universidad Complutense de Madrid).
Correo: fabero@dacya.ucm.es

⁴ Doctora en Ciencias Físicas. Profesora Facultad de Informática (Universidad Complutense de Madrid).
Correo: hortem@dacya.ucm.es



Palabras-claves: FPGA, tiempo real, PLC, reconfiguración parcial

Introducción

Los sistemas automáticos de control han sido implementados por los controladores lógicos programables (PLC). 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 y analógicas. Si se elige la configuración modular puede adaptarse a situaciones en 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.

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

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.



I CONGRESO INTERNACIONAL DE CIENCIA, DISEÑO Y TECNOLOGÍA

11 y 12 de octubre | Encarnación, Paraguay



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 optimizando 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.

A diferencia de un PLC las FPGAs tienen las siguientes ventajas:

- menor ciclo de desarrollo debido a las herramientas de simulación que tienen los entornos de desarrollo
- se pueden realizar pruebas con los diseños parciales con diferentes configuraciones haciendo que el tiempo de desarrollo del producto final disminuya.
- 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. 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.

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, que 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 solo 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 la aplicación de 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 que esto implica. Así, cuando se detecten discrepancias entre los resultados obtenidos por alguna de las réplicas funcionales del módulo, este puede reconfigurarse dinámicamente y continuar con la ejecución de la tarea.

Finalmente, hay que mencionar que originalmente las redes de comunicación industriales, a diferencia de los sistemas ofimáticos, posibilita el intercambio de los diferentes componentes de un sistema de control distribuido como sensores, actuadores, interfaces HMI, PLCs, PCs industriales, etc., permitiendo la flexibilidad en el intercambio de datos de los equipos que participan en la red.

II. PROCESO CONTROLADO

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 con la finalidad de homogeneizar la combinación de ambas composiciones.

Una vez que el nivel del primer tanque alcanza 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 cierran Q0.0 y Q0.1, abriéndose nuevamente al bajar el nivel por debajo de I0.2.

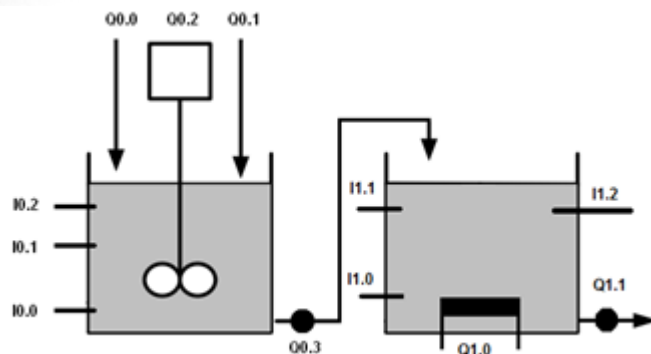


Fig. 1. Proceso a controlar

En el segundo tanque el producto procedente del primer tanque se 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., que cuando alcanza la temperatura de referencia desactiva Q1.0 y 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 las 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 con el entorno de desarrollo Vivado 15.4 de Xilinx.

III. DISEÑO DEL CONTROLADOR



Primeramente, desarrollamos el control utilizando el PLC S7 -200. En la Figura 2 se ve el diseño en escalera y se muestra el esquema correspondiente al control de primer tanque.

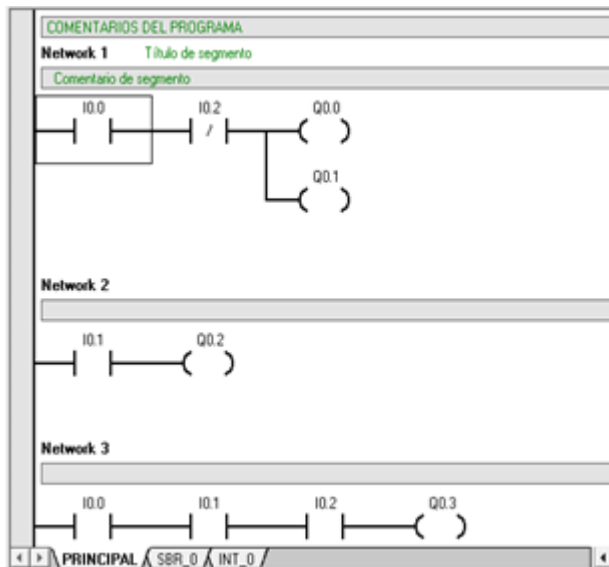


Fig. 2. Esquema en escalera del controlador

En la figura 3 vemos la misma funcionalidad, pero en lenguaje de comandos.

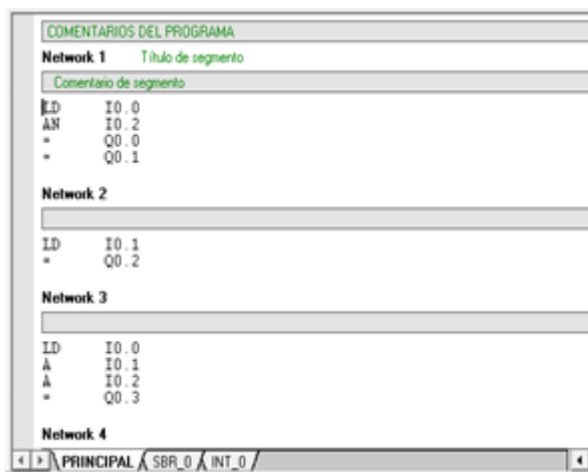


Fig. 3. Esquema en lenguaje de comandos del controlador

En la Figura 4 vemos el controlador, pero en esquema de lenguaje de compuertas.



En la Figura 4 vemos el controlador, pero en esquema de lenguaje de compuertas.

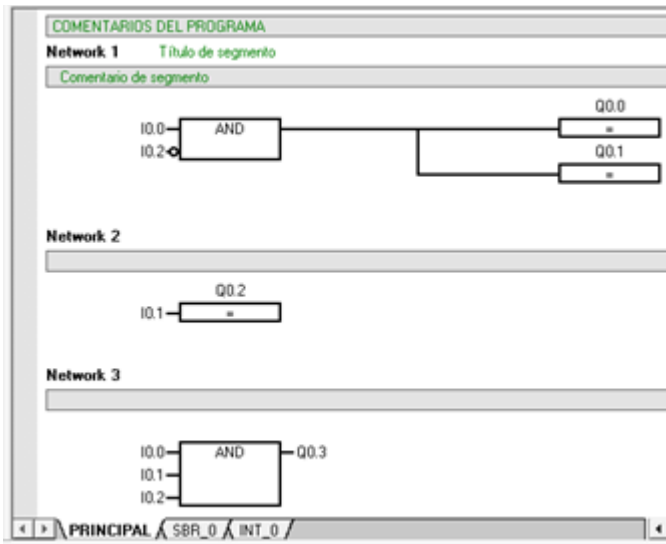


Figura 4 Esquema en lenguaje de compuertas del controlador

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.

```
nivel3_top.v
C:/plcpga/nivel3/nivel3.srcs/sources_1/new/nivel3_top.v
21
22
23 module nivel3_top(
24     input wire clk,
25     input wire [2:0] I0,
26     input wire [2:0] I1,
27     output [3:0] Q0,
28     output [1:0] Q1
29 );
30
31     TANK_1 TANQUE1 (.clk(clk),
32                   .I0(I0),
33                   .Q0(Q0)
34                   );
35
36     TANK_2 TANQUE2 (.clk(clk),
37                   .I1(I1),
38                   .Q1(Q1)
39                   );
40 endmodule
41
```

Fig. 5. Programa estructurado en Verilog





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 módulo 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.

```
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
```

Fig. 6. Código para controlar el 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 los sistemas automáticos de control de procesos (Figuras 8a, b y c).





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 reconfiguradas sin necesidad de interrumpir su funcionamiento.

Por ello su aplicación permite reconfigurarse si un módulo sufre una falla para reemplazar el modulo que no funciona. Puede ser el reemplazo de un controlador maestro en un sistema distribuido o un nodo de medida y control que funciona como esclavo, observándose nuestro esquema de control supervisado que puede funcionar de manera ininterrumpida.

```
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
```

Fig. 7 Código de control para el segundo tanque

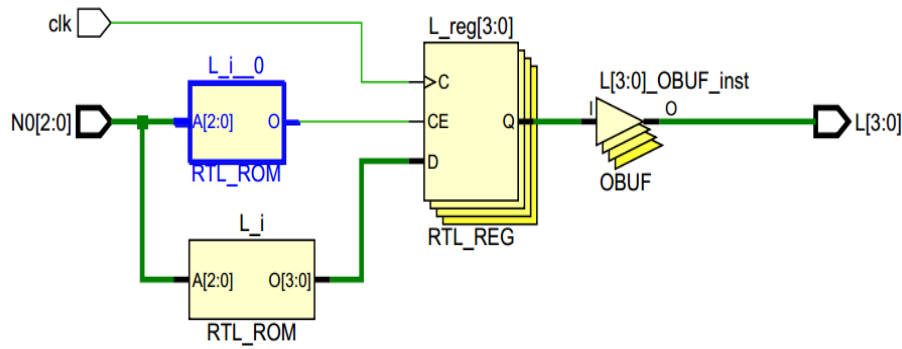


Fig. 8 a

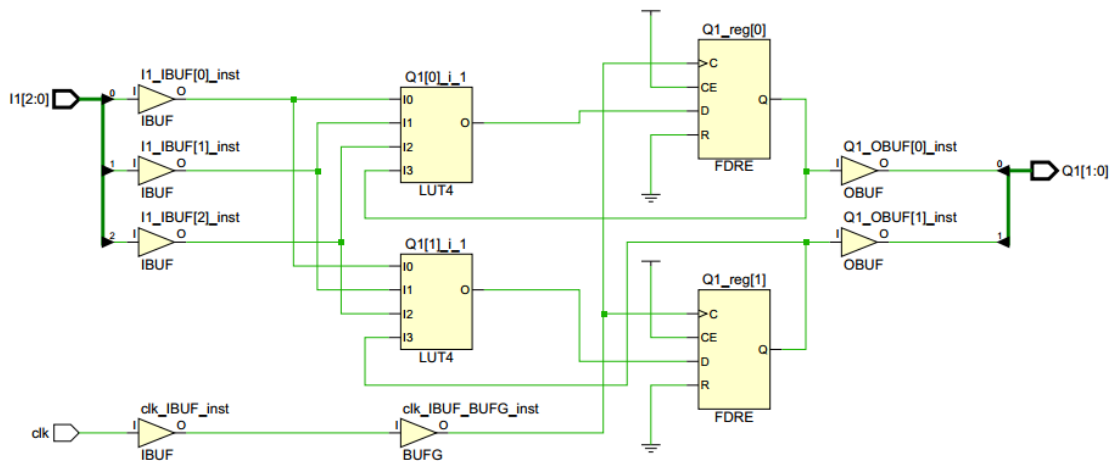


Fig. 8 b

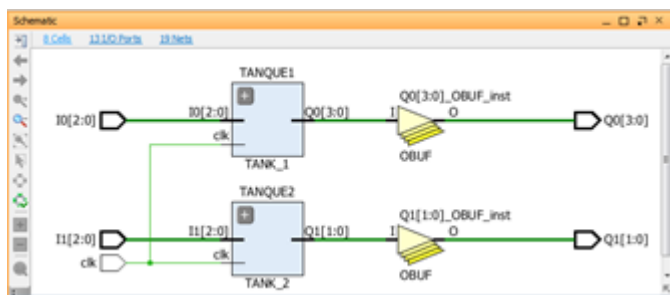


Fig. 8 c Esquema del controlador.

IV. RESULTADOS





Para comprobar la respuesta de ambos sistemas hemos probado nuestro modelo de proceso utilizando primeramente el diagrama Ladder con el PLC S7- 200 de Siemens. Las entradas y salidas las 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.

V. CONCLUSIONES

El proyecto está en fase inicial de ejecución. El controlador puede ser ampliado para un mayor número de sensores. 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 más versátiles y el sistema de control tenga un mayor alcance y mayor conectividad con otras redes.

Referencias

JUNBEOM YOO; JONG-HOON LEE; JANG-SOO LEE. A RESEARCH ON SEAMLESS PLATFORM CHANGE OF REACTOR PROTECTION SYSTEM FROM PLC TO FPGA In: Nuclear Engineering and Technology, Vol. 45, Iss 4, Pp 477-488 (2013); Elsevier, 2013

L. Berrojo, F. Corno, L. Entrena, I. Gonzalez, C. López, M. Sonza Reorda, and G. Squillero. An industrial environment for high-level faulttolerant structures insertion and validation. In Proc. 20th IEEE VLSI Test Symp., pages 229 - 236, 2002.

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

Chmiel, M.; Kulisz, J.; Czerwinski, R.; Krzyzyk, A.; Rosol, M.; Smolarek, P. An IEC 61131-3-based PLC implemented by means of an FPGA. Modern Techniques of Design and Implementation of Highly Flexible Controllers, Microprocessors and Microsystems. July 2016 44:28-37 Language: English. DOI: 10.1016/j.micpro.2015

Economakos, C. Skarpetis, M. Economakos, G. Program-based and model-based PLC design environment for multicore FPGA architectures 2014 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO) Informatics in Control, Automation and Robotics (ICINCO), 2014 11th International Conference on. 01:726-733 Sep, 2014.

Subbaraman, S. Patil, M. Nilkund, P 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 Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on. :1791-1795 Dec, 2010.

F. Fernández and J.C. Fabero. Allocation algorithms for dynamic partial reconfiguration on reconfigurable hardware. In 3rd Conference of Computational Interdisciplinary Sciences, pages 402 - 404, September 2014.



I CONGRESO INTERNACIONAL DE CIENCIA, DISEÑO Y TECNOLOGÍA

11 y 12 de octubre | Encarnación, Paraguay



Radulovic, M.I. Celebic, V.V. Kabovic, M.M. Kabovic, A.V. Realization of the teleprotection equipment interface for working with PLC equipment using FPGA logic 2012 20th Telecommunications Forum (TELFOR) Telecommunications Forum (TELFOR), 2012 20th. :525-528 Nov, 2012.

Adam Milik On Ladder Diagrams Compilation and Synthesis to FPGA Implemented Reconfigurable Logic Controller Advances in Electrical and Electronic Engineering, Vol 12, Iss 5, Pp 443-451 (2014).

Gawali, D. Sharma, V.K FPGA Based Micro-PLC Design Approach 2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies Advances in Computing, Control, & Telecommunication Technologies, 2009. ACT '09. International Conference on. :660-663 Dec, 2009.

Digilent 2015. Nexys 4™ FPGA Board Reference Manual. Disp. en:
https://reference.digilentinc.com/_media/nexys:nexys4:nexys4_rm.pdf

Economakos, C.Economakos, G. An architectural exploration framework for efficient FPGA implementation of PLC programs 2009 17th Mediterranean Conference on Control and Automation Control and Automation, 2009. MED '09. 17th Mediterranean Conference on. :1172-1177 Jun, 2009.

Patel, D. Bhatt, J. Trivedi, S. Programmable logic controller performance enhancement by field programmable gate array based design Elsevier ISA Transactions January 2015 54:156-168

<http://www.ingmecafenix.com/automatizacion/lenguajes-programacion-plc/>