ISSN impreso 1996-6660 / ISSN online 2519-5719
DOI: https://doi.org/10.31381/perfilesingenieria.v20i21.6320

First steps using FPGA technology

Primeros pasos usando la tecnología FPGA

Pedro Selencio, Landaeta Herrera

RECIBIDO: 15 de febrero de 2024.
ACEPTADO: 20 de mayo de 2024.

Abstract
This research has as main objective to show in a general way the FPGA technology, Field Programmable Gate Array – Arrangement of programmable Gates on the Chip. We will talk here about its architecture, manufacturing companies, description of a simple circuit using VHDL, Very High Speed Hardware Description Language, Quartus II Software and the INTEL Cyclone II card. A simple combinational logic circuit was used to carry out the simulation with the Quartus II software. Leaving everything clearly indicated to carry out larger projects. The results of the study show that the graphs obtained as a response of the circuit studied were as expected, and also that the properties of the software under study could be verified. The general characteristics of the FPGA technology were studied. The basic functions of the Quartus II software were known to execute a project in all its phases. This is a basic project. In the future there are projects that work with sequential logic and with other tools such as the ISP Lever software from the LATTICE Semiconductor company, which allows, using only software, to develop FPGA projects. In other words, this research serves as the basis for larger-scale projects.
Keywords: FPGA, VHDL, software, logic, technology.

Resumen
Esta investigación tiene como objetivo principal mostrar de manera general la tecnología FPGA, Field Programable Gate Array – Arreglo de Compuertas programables en el Chip. Se hablará aquí de su arquitectura, empresas fabricantes, descripción de un circuito sencillo usando VHDL, Very High Speed Hardware Description Language, Software Quartus II y la tarjeta Cyclone II de INTEL. Se uso un circuito sencillo de lógica combinacional para efectuar la simulación con el software Quartus II. Dejando todo claramente indicado para realizar proyectos de mayor envergadura. Los resultados del estudio muestran que las gráficas obtenidas como respuesta del circuito estudiado fueron las esperadas, y además que se pudieron comprobar propiedades del software en estudio. Se estudiaron las características generales de la tecnología FPGA. Se conocieron las funciones básicas del software Quartus II para ejecutar un proyecto en todas sus fases. Este es un proyecto básico, a futuro vienen proyectos donde se trabaje con lógica secuencial y con otras herramientas como el software ISP Lever de la empresa de Semiconductores LATTICE, que permite, usando solamente software, desarrollar proyectos de FPGA. Es decir, esta investigación sirve como base para proyectos de mayor envergadura.
Palabras clave: FPGA, VHDL, software, lógica, tecnología.

Cómo citar
P. S. Landaeta, «First steps using FPGA technology », Perfiles_Ingenieria, vol. 20, n.º 21, pp. 136-149, jun. 2024.

© Los autores. Este artículo Open Access está publicado bajo la Licencia Creative Commons Atribución 4.0 Internacional (CC-BY 4.0).

1. Introduction

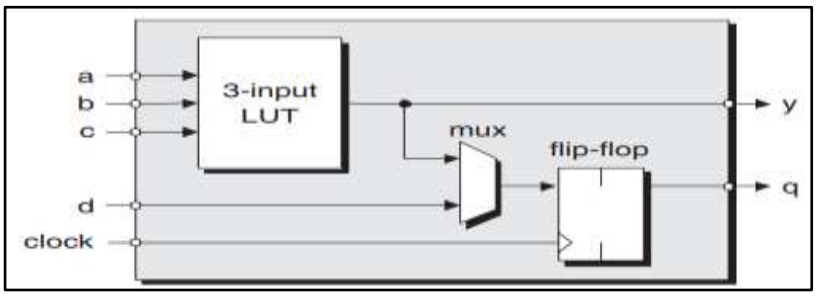
FPGA technology has existed since the mid 80's and came to solve a gap that existed between PLDs [1] (PLD: Programmable Logic Device) and the ASICs [2] (ASIC: Application Specific Integrated Circuit). The main objective of this research is to talk about all these basic issues and describe a very simple project using VHDL language, and thus know the initial steps that must be taken to undertake projects using this technology. There are several aspects that must be taken into account to understand this interesting world of FPGAs, one of them is the issue of device manufacturers. In this group are XILINX, INTEL FPGA (Former ALTERA) and LATTICE. There are others, but only these three are mentioned because they are the relevant ones in this investigation. XILINX for being the first FPGA manufacturer worldwide, INTEL FPGA because it is the one used in research and is also the second most important worldwide for its sales, and LATTICE because it provides the FPGAs used to work with Free Software and Hardware. Another factor to take into account is related to the language used to describe FPGAs, such as HDL (Hardware Description Language). There is VERILOG and VHDL, in this investigation VHDL will be used [3] [4].

Another point of interest is the choice of the FPGA Chip to use in the project. For this investigation, the FPGA CYCLONE II from the company INTEL FPGA will be used. This is a chip already discontinued by the company, but it is still available for sale over the Internet. And the last thing to deal with is related to the software provided by the manufacturing company to describe the hardware. In the case of INTEL FPGA, from now on simply INTEL, it works with the QUARTUS II web edition 13.0 sp1 (sp: Service Pack) software, which is a version offered free by the company, and is more than enough to carry out the projects initials of any beginner, it is also the only version of the software that supports the CYCLONE II chip [5].

The work methodology focuses on making known the architecture of FPGAs and also showing the steps to undertake a project with FPGA from the INTEL company, with the CYCLONE II chip, using the VHDL language with the support of the QUARTUS II web edition 13.0 software sp1 explaining it from the most elementary. The internal architecture of the CYCLONE II chip shows that it has everything necessary to undertake the basic projects that are aspired to be carried out, since the chip has 4608 LEs (Logic Elements) and the company offers the card with a plug for +5V power, port to download the description made in the QUARTUS II software, LEDs and Pushbuttons. All arranged and ready to start with the initial projects. LEs are the smallest part of the FPGA architecture [6], it is called that in INTEL and in XILINX it is called Logic Cell. The LEs they are made up of the LUTs (Look Up Tables), these can have 2, 3, 4, 5, 6 or another number of inputs depending on the FPGA model. In the LUTs is where everything that is the gate logic of the design in question is built or assembled. Multiplexers (mux) are used when it is necessary to design systems with multiple inputs and a single output. And the Flip-Flops that all FPGAs have are type D, whatever the manufacturer. Figure 1 shows the basic internal architecture of an FPGA where its D input (D for Data: 1 or 0), the synchronization clock called CLOCK and the Q output can be observed. For each high clock level, the Q output will take the value of input D. Figure 1 shows the LUT, MUX, and Flip Flop and how they come together in the internal building architecture of the FPGA.

ISSN (Digital): 2519-5719

Figure 1.
Basic components of an FPGA.



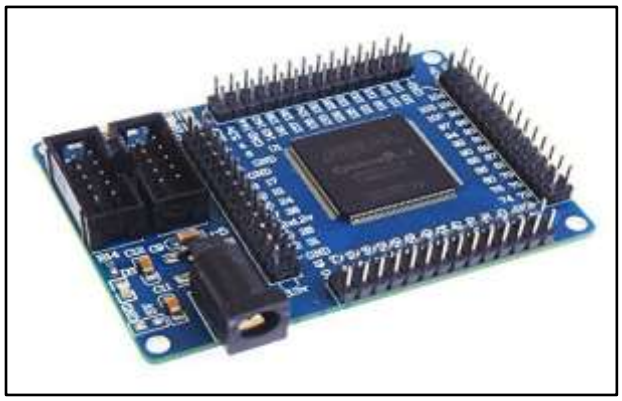
It can be read in the data sheet of the manufacturer of the IC (IC: Integrated Circuit) CYCLONE II on the Internet that the Logical Array Consists of LABs (Logic Array Block), with 16 LEs in each LAB [7].

2. Materials and methods

The work consists of talking about the architecture of the FPGA, describing using VHDL a basic combinatorial logic circuit using Quartus II and the cyclone II chip. This Chip is mounted on a card as shown in Figure 2 where you can see the integrated FPGA ep2c5T144C8 with the peripherals that the manufacturing company places to take advantage of the circuit in basic initial projects and other larger projects that you want to do.

The hardware shown in figure 2 is complemented by using the QUARTUS II simulation tool [8] allows working with the VHDL language and with other languages that QUARTUS II incorporates, as well as having the option of making the assemblies in schematic mode.

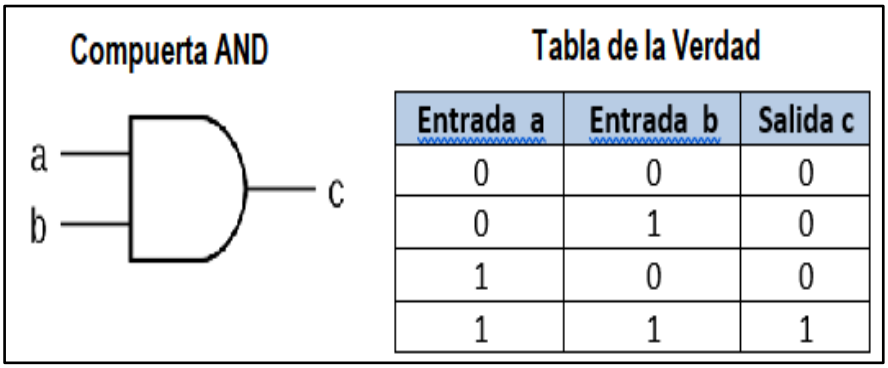
Figure 2.
FPGA mounted on the card with peripherals



To start with the work, suppose that you have the circuit of figure 3 [9] and you want to run it with the FPGA. This involves using VHDL to describe the circuit and verify that the truth table is true. Using FPGA, it is possible to implement in a practical way all combinatorial

and sequential logic assemblies. In addition to state machines, memories and digital signal processing, all with VHDL. This is necessary to comment so that the new readers of the subject can understand or assimilate that something that was done with 10 integrated circuits or more, is now done with a single chip or IC FPGA.

Figure 3.
Basic logic circuit to describe using VHDL



It is mostly to put into context what it means to use FPGAs and the advantages they bring. The first thing to have is a software to make the description. In this case, the tool is searched for on the page of the manufacturer of the card with which it is working [9]. Since you are using an INTEL company card, you must go to the page provided by the manufacturer. figure 4 shows the availability of the QUARTUS II software to be downloaded with the INTEL logo on the website.

Figure 4.
Web page where you can see the version of the software to download

After this, the indicated software is downloaded and work will begin on the descriptions of the selected hardware. The software you download is a free version from the company. It has some limitations, but the important thing is that it is more than enough for the objectives set out in this research. It is already known that the FPGA chip is the CYCLONE II from the INTEL company. And also that the software that is being indicated to download is QUARTUS II 13.0 sp1. For other FPGA series, for example CYCLONE III, CYCLONE IV or CYCLONE V, and

ISSN (Digital): 2519-5719

for other FPGA versions from the same INTEL company, the QUARTUS II software changes its version. On the INTEL company page is the QUARTUS II recommendation for your selected FPGA chip.

By having the hardware to be described and the software installed on the PC (Personal Computer), the VHDL description of the hardware indicated in Figure 3 is carried out. To do this, you must study the basics of VHDL and on the Internet there is a lot of material related to tutorials that serve as support to start with this language. Professor Carlos Fajardo from the Industrial University of Santander in Colombia has some videos that very well illustrate all this about VHDL and he teaches in each one of them the degrees of difficulty in the description. [10].

After all this study, a description can be made as shown in Figure 5. The description in figure 5 highlights the points of interest in VHDL. First, the libraries with which you are working are observed. All descriptions have the Library IEEE semicolon header, then comes the batch of libraries that will be used in the description, one that should always go is use ieee.std_logic_1164.all semicolon. The semicolon signifies the closure of that line of code. Then in the description comes the entity. The entity has to do with the declarations of inputs and outputs of the digital system that is being developed. Finally, the architecture, in this space the codification that indicates what you want to do in the project is placed.

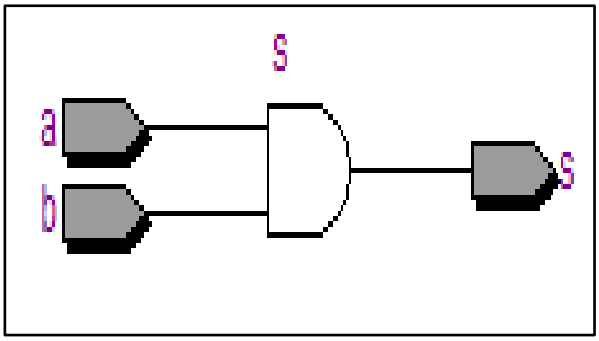
Figure 5.
VHDL description of the hardware in Figure 4

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity Nuevo is

port(
        a: in std_logic ;
        b: in std_logic ;
        s: out std_logic  );

end;

architecture Comportamiento of Nuevo is
begin
      s <= a and b;

end Comportamiento;
```

After having the code ready comes the compilation part. In the Processing menu or the toolbar, select the Start compilation option. The software will send a message saying if everything is fine or not, in the case of the example everything was satisfactory, that is, there are no errors. Already at this level you have the possibility to see the described hardware using the QUARTUS II software and you can also see the simulation. The compilation process returns the circuital synthesis. Following the path TOOLS in the →Netlist Viewers→ RTL Viewer

menu bar you get to the RTL view (Register Transfer Level) [11] of the description made in VHDL

Figura. 6
Hardware result of compilation and synthesis



The definitive proof that the circuit will be exactly the one that was originally intended to be assembled consists in making the compilation that automatically brings with it the circuital synthesis. Then, reaching the RTL viewer in the menu bar, you will see the circuit produced by the software used, be it QUARTUS II or ISE WEBPACK from XILINX, as seen in the circuit in Figure 6, and which shows the interpretation given by the VHDL description software corresponding to the circuit that was intended to be modeled (see Figure 3).

After the above, the simulation stage is carried out. It should be clarified that the MODELSIM software from the INTEL company will not be used because this is an initiation investigation on the subject of FPGAs. Therefore, the University Program VWF (Vector Wave File) tool is used. You are in File New, double click on University Program VWF. When doing the above a window will appear, double click on the blank part to the left of that figure and in this way a window called Insert Node or Bus will appear, which prepares the way to go to the input and output signals of the project you are working on, click on the Node Finder button.

After clicking on Node Finder, check that the Pins All option is in the next window and click on the List button and all the inputs and outputs placed in the description made in VHDL will appear.

ISSN (Digital): 2519-5719

Figure 7.
Window for simulation

After clicking on the List button, you must select the inputs and outputs that you want to appear in the simulation, see figure 7. After that, click on the OK button. When doing so, the window shown below in Figure 8 appears.

After the above, click on the clock window option in the toolbar. This option allows assigning the value to the time of each of the input signals. In [12] and [13] an explanation is made in more detail of what is discussed here.

The entries are then selected and a time is assigned to each one. In this case under study, 100 ms high and low was assigned to input A. And 200 ms high and low to input B, see figure 9.

After simulating the description made, it is passed to the recording stage in the IC FPGA. For this, some hardware devices are used, made up of what the company INTEL calls the USB Blaster, made up of the electronics card named USB Blaster, a communications cable to be used between the USB Blaster and the CYCLONE II FPGA board, and a USB cable to be used between the CYCLONE II board and the computer.

Figure 8.
Input and output variables ready for simulation

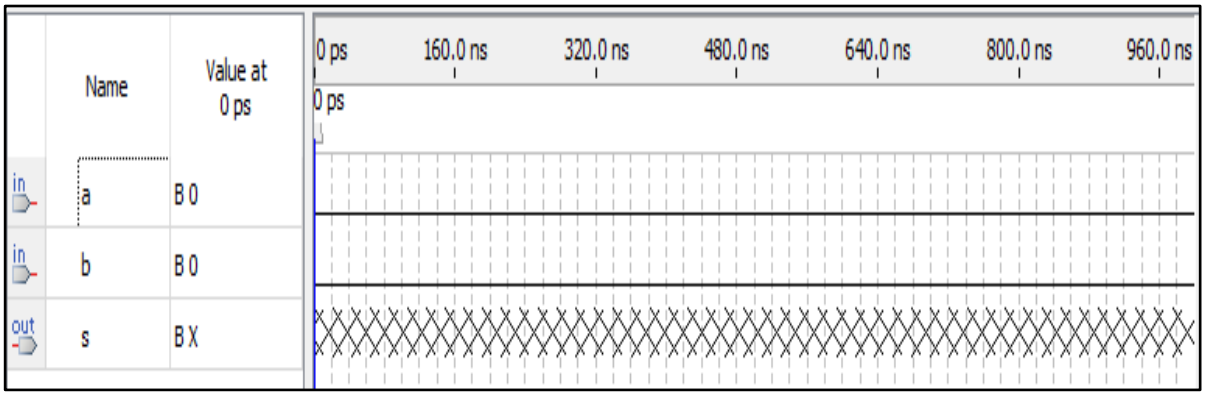


Figure 9.
Input and output signals after running the simulation

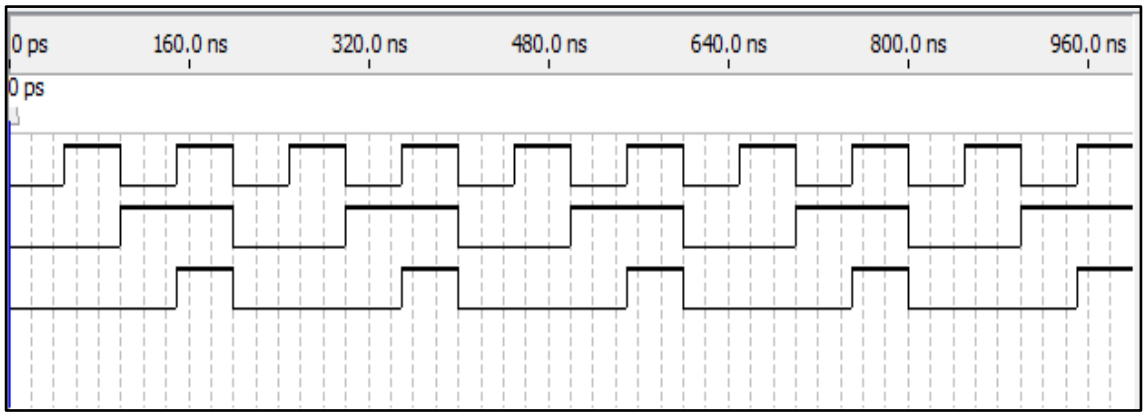
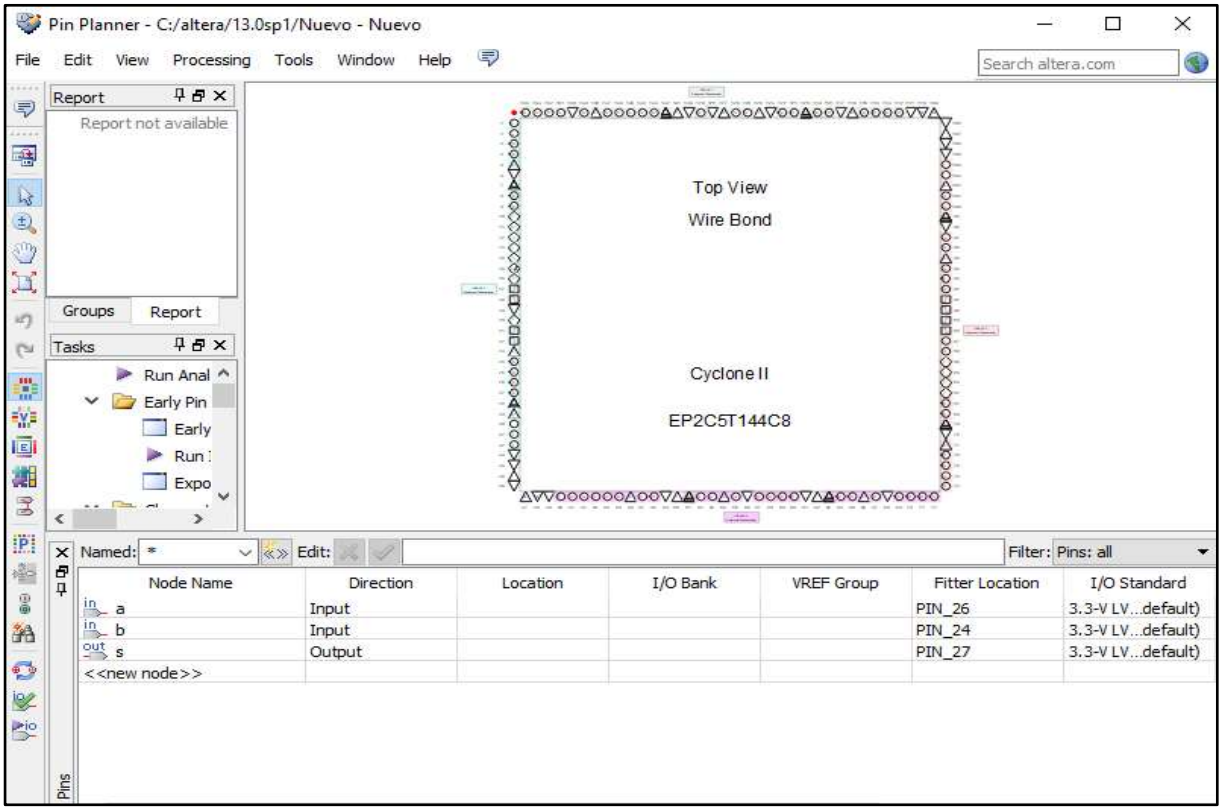


Figure 10.
Option to assign the input and output pins of the FPGA



So, to record the FPGA chip what is done first is the pin assignment. In the Quartus II software you can see the pin assignment using Pin Planner, choosing from the Assignments menu bar you get to the Pin Planner option. This window in figure 10 shows the pin assignment for the circuit proposed in Figure 3. After the assignment, the FPGA is recorded using the USB

ISSN (Digital): 2519-5719

Blaster. In figure 11 all the parts of the USB Blaster assembly are shown. These three parts are the basic ones to record what is done in the Quartus II after the description is compiled satisfactorily. Additional information in this regard is shown in [14].

Figure 11.
USB Blaster to burn the description in the FPGA

3. Results

- The main objective of the research was achieved, as it was to show the FPGA technology in a general way. Using an elementary combinatorial circuit, all the necessary steps were taken to implement a project with FPGA. Software to use, references and important comments to understand each step of the process and know all the necessary equipment for its implementation in hardware.

- As part of the main objective of the investigation that was also fulfilled, there is the description made in VHDL shown in Figure 5. It shows all the parts in which any description is composed.

- Another of the objectives was to know the architecture of the FPGAs, in Figure 1 the most basic of what these integrated circuits are composed of Multiplexers, flip flops and Lookup tables (LUT) are mentioned. There are many more components, such as the gate array, Microprocessor, DSP, ROM memories, input and output devices for communication with the real world in applications that warrant it. Since the research refers to the beginnings of this technology, these issues were not touched on because it would provide material for another article. On the other hand, this architecture differs from the one used in the CPLDs in processing capacity and price of the device.

- The practical assembly in Figure 6 is the result obtained after compiling the description made with the Quartus II software. This shows that any circuit that is described with VHDL will appear inside the FPGA chip.

- As a last result, we have the output of the combinatorial logic circuit studied. In Figure 9 you can see the signals supplied by the Quartus II software.

4. Discussion

A relevant point to bring to the discussion is the lack of basic material to understand the elementary principles of FPGAs. It is very difficult for a beginner to find recent material from indexed journals or research papers endorsed by universities, be it theses, essays or technical reports where research on basic aspects of this technology is shown. Hence the importance of this material to contribute to that void found in it. Taking the above into account, a book that does not pass and transcends time to help every beginner in this FPGAs is the book by Clive Maxfield edited by Newnes and Elsevier in 2004 [15]. You can also find publications of FPGA scientific articles for beginners in magazines that are not indexed in recognized databases such as SCOPUS, Scielo or Latindex for example, but it is what is available to be downloaded freely and without any restriction, such this is the case of the International Journal of Trend in Scientific Research and Development (IJTSRD). There you can find the article by San Naing et al, published in 2019 on Operation of FPGA-based Digital Logic Circuits for Beginners [16]. If the research has some type of funding, more recent research can be found. The rest that is found on the Internet is research that goes beyond the basics, such as those carried out by applications of the NIOS II processor [17] of the FPGA of the INTEL company, works related to signal processing for aerospace applications [18] or math applications using VERILOG [19].

Although this research works with proprietary software and hardware, free software and hardware are not intentionally left aside. The reason for working with proprietary hardware and software basically has its essence because it was what was found at the beginning of the investigation. The free versions have VERILOG and myhdl from the Python ecosystem. These free versions need a Free Software (FS) and Free Hardware (FH) structure to close the project cycle. Because it's not just programming in VERILOG. In FPGA projects, making the description, which for other software is programming, is not everything. The description can also be done from a proprietary software such as QUARTUS II and it comes from a company that of course does not work with FS and FH. In FPGA projects, you should always invest in an electronic card, and it is at that point that it is not so free because you still have to make an initial investment of a few Dollars (USD) in the electronic card that adapts to this methodology. This can go from 30 USD to 150 USD. After that initial expense is where the freedoms of use, study, distribution and modification come. But, it is not the central topic of this investigation because just knowing the structure of the new card with a different chip than the one that has been used is a separate investigation, since the majority of FH uses FPGA ICs from the LATTICE company and later knowing the software associated with it to close the project cycle would correspond to another investigation.

Another aspect that was left out of the research was FPGA simulation using PROTEUS. Actually, what is available in PROTEUS is a SPLD, Simple Programmable Logic Device, GAL 22v10. To simulate this device, the VHDL language simulation software must generate a .jed extension, which QUARTUS II does not provide. There you need other software. In this case, the ispLEVER software from LATTICE meets the desired characteristics. The company allows you to download a free version with a 1-year license to use. There is other software that can generate the .jed file but the description is no longer in VHDL, as in the case of WinCUPL that's why it's important to use ispLEVER.

5. Conclusions

Although in the investigation we only worked with an AND gate for reasons of seeking maximum simplicity for the rapid understanding of the learning process of project development with FPGA. Larger combinational circuits and also sequential circuits could be worked on. The advantage of this is that a large number of Integrated Circuits would not be used, thus reducing the number of errors that can be made in a practical assembly by a high percentage. Since an FPGA IC is enough to make a project that can contain 200 Flip Flops and 100 Gates or more.

To complement what has been said above, it should be said that the entire work process with the Quartus II software for this simple AND gate is the same as it would be used for a complex system of gates, flip flops and larger digital integrated circuits. The steps are identical, and that is what makes this investigation relevant.

The versatility of the INTEL Quartus II software to describe circuits and to achieve the simulation and tuning of the described circuit was demonstrated.

6. References

[1] P. Kumar, «Development of Programmable Logic Devices, » International Journal of Innovative Research in Computerand Communication Engineering, vol. 8, nº 4, 2020.

[2] P. Kumar, «Analysis of ASIC Design Methodology in Advanced Digital System, » International Journal of Innovative Research in Computer and Communication Engineering, vol. 8, nº 4, 2020.

[3] E. Saeed, Basics VHDL Labs. Basics Lab for Learning VHDL. Learning with Experiments, Al Nahrain. Irak: Lap Lambert, 2020.

[4] A. Ivanova, «A Concept of Visual Programming Tool for Learning VHDL, » IOP Conference Series: Materials Science and Engineering, 2020.

[5] INTEL-ALTERA, «Quartus II Handbook Version 13.1. Volume 1: Design and Synthesis, » November 2013. https://www.intel.com/content/www/us/en/software-kit/666221/intel-quartus-ii-web-edition-design-software-version-13-1-for-windows.html.

[6] S. Rai, P. Nath, A. Rupani, S. Vishvakarma y A. Kumar, «A Survey of FPGA Logic Cell Designs in the Light of Emerging Technologies, » IEEE Access, vol. 9, pp. 91564 - 91574, 2021.

[7] INTEL, «Cyclone II Device Handbook, Volume 1, » 2008. [En línea]. Available: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyc2/cyc2_cii5v1.pdf.

[8] C. Figueroa y S. Giraldo, Diseño y construcción de un instrumento prototipo para el análisis de la ley de Hooke, caida libre y pendulo simple usando dispositivos FPGAs, Pereira. Colombia, Universidad Tecnológica de Pereira. (Tesis de Maestría), 2019.

[9] INTEL, «Descargar software QUARTUS II,» 2021. [En línea]. Available: https://www.intel.com/content/www/us/en/software-kit/666221/intel-quartus-ii-web-edition-design-software-version-13-1-for-windows.html.

[10] C. Fajardo, «Introducción a VHDL. Circuitos Secuenciales: Registros y Contadores. Universidad Industrial de Santander,» 2013. [En línea]. Available: https://www.youtube.com/watch?v=vTmBuR-JgvM.

[11] B. ROBLES, Principios del FPGA y aplicaciones en el control de procesos industriales., Universidad de El Salvador. San Salvador. El Salvador: (Tesis de Pregrado), 2016.

[12] F. Urbano, «Simulación usando Altera Quartus II 13 sp1 Web Edition,» 2013. [En línea]. Available: https://www.youtube.com/watch?v=C7f5mO6Tjk0.

[13] C. Benalcazar y I. Andrade, Diseño e implementación de prácticas con FPGA para la materia de Digitales II mediante la herramienta de trabajo Cyclone II de Altera, Guayaquil. Ecuador, (Tesis de Pregrado), 2012.

[14] INTEL, «Intel FPGA USB Download Cable User Guide,» 2016. [En línea]. Available: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/ug/ug_usb_blstr.pdf.

[15] C. Maxfield, The design warrior´s guide to FPGA, Road Burlington, Massachusets, USA: ELSEVIER, 2004.

[16] S. Naing, N. San y T. Chao, «FPGA Based Digital Logic Circuits Operation for Beginners,» International Journal of Trend in Scientific Research and Development (IJTSRD), vol. 3, nº 5, pp. 495-501, 2019.

[17] H. Cao y U. Meyer, «XML-Based Automatic NIOS II Multi-Processor System Generation for Intel FPGAs,» Electronics, 2022.

[18] F. Santos, Emulación basada en FPGA de los efectos de los single event upsets ocasionados por la radiación en circuitos digitales tolerantes a fallos, Madrid. España, Universidad Complutense de Madrid. (Tesis Doctoral), 2018.

[19] M. Mustaph y N. Zulkarnain, «Full cycle trigonometric function on Intel Quartus II Verilog, » International Conference on Engineering and Technology (IntCET 2017), 2018.

[20] U. Cem y T. Bora, Digital System Design with FPGA: Implementation Using Verilog and VHDL, Estambul. Turquia: McGraw Hill, 2017.

Trayectoria académica

Pedro Selencio, Landaeta Herrera

Universidad Politécnica Territorial del Estado Aragua “Federico Brito Figueroa”, Venezuela

Profesor jubilado de la Universidad Politécnica Territorial del Estado Aragua (UPTA) “FBF”. Venezuela. Profesor perteneciente al departamento de Electrónica y Telecomunicaciones.

Jefe del departamento de Investigación en la sede de Maracay.

Especialista Ingeniero en Electrónica UPTA-FBF

Actualmente soy investigador independiente ya que estoy en condición de Jubilado con 25 años de servicio.

Asistí a congresos científicos nacionales e internacionales, fui seleccionado para optar al premio Nacional de Ciencia con mi investigación de sistemas inalámbricos.

Autor corresponsal: landaetabiz@gmail.com

Orcid: https://orcid.org/0000-0002-8843-4991

Contribución de autoría

Pedro Landaeta: Un único autor.

Conflicto de intereses

El autor sostiene que no existen conflicto de intereses en el desarrollo de la investigación.

Responsabilidad ética y legal

El desarrollo de la investigación se realizó bajo la conformidad de los principios éticos del conocimiento, respetando la originalidad de la información y su autenticidad. Se han citado las fuentes con sus respectivos datos, y no me ha atribuido algún concepto o filosofía planteada. Todos han sido asignados a sus respectivos autores.

Declaración sobre el uso de LLM (Large Language Model)

Este artículo no ha utilizado para el desarrollo de la investigación textos provenientes de LLM (ChatGPT u otros).

Financiamiento

La presente investigación ha sido financiada con los recursos propios del autor.

Correspondencia: landaetabiz@gmail.com

ISSN (Digital): 2519-5719