Towards FPGA learning through a game-based approach

Miguel Marcelo
NOVA School of Science and Technology
Monte de Caparica, Portugal
Email: m.marcelo@campus.fct.unl.pt

Filipe Moutinho
NOVA School of Science and Technology
UNINOVA, Centre of Technology and Systems
Monte de Caparica, Portugal
Email: fcm@fct.unl.pt

***Abstract*—Field-Programmable Gate Arrays (FPGAs) are becoming more relevant with time, but only a restricted group of people are capable of working with this technology due to its complicated learning curve. This paper presents the work that is being done towards learning FPGAs by taking a game-like approach. Games often entice players enough to keep them focused in the task at hand and by taking that into account, by gamifying the learning of FPGAs, it is possible to take advantage of said player's attention to help convey the knowledge in a creative, inventive and dynamic manner.**

I. Introduction

Technological advancements are continuously shifting in today's society, but sometimes these changes are not related to a new technological development but rather a new or different way of implementing a certain system. Integrated circuits, that can be configured and reconfigured, have made their way into being implemented into existing technologies to achieve better performances overall.

Field-programmable Gate Arrays (FPGAs) are integrated circuits that can achieve higher parallelism within the circuit and lower system's processing time, thus making its way into different types of applications in the last couple of years [1, 2]. During an event, hosted by *The Next Platform* on the 5^{th} of February of 2020, Archonix's vice president of product planning and business development, Manoj Roge firmly affirmed that FPGAs will have an important role as "a programmable accelerator for deployments from cloud to edge to IoT" [3].

The motivation to this work ends up focusing on finding ways to teach those who wish to learn about FPGAs by introducing a dynamic method that intends to grab the player's attention and hopefully provide a unique environment that will assist the user during the learning process. The method has been applied into introducing players to the basics of logic gates and circuitry in "Make it True", a mobile game which is available in [4].

Ideally the contribution of this work will consist on a browser game, designed to explain and help the player learn about FPGAs and their properties and functions. The user will be able to learn about each of the main blocks that take part in this device's architecture and assimilate what those block's purposes are. Given the nature of these devices, this knowledge will be built from ground up, starting with small challenges, puzzles and quizzes, and little by little adding extra layers of difficulty and introducing more complex concepts.

This paper will be divided in three main parts, excluding the present introduction. The first part, which will be the second section of this article, will bring to light the solutions that are already available to those who wish to learn about FPGAs, while also listing drawbacks relative to each method. The third section will explain in more detail the game that is being developed. The last section of this paper will convey and explore the conclusions taken from all the work done until the present date taking into account the game in development, the existing solutions and will explore the reasoning for taking this method into account.

II. Related Work

Various methods are available to teaching FPGAs in hopes of attracting more enthusiasts into this very restricted area [5]. To organize the available methods researched into smaller groups, this section is divided into three main sources of information, giving an overview of the content found. The order of these subsections is somewhat relevant to point out, since most of the interested public, while searching the available teaching methods, will follow their research along this path - this organizational aspect is based on personal experience when searching and learning about this topic, because it can somewhat represents the efforts of a new enthusiast towards learning about FPGAs.

A. Videos

The first approach taken by most would be to instinctively open a search engine and type "What is an FPGA?" and check up on a few links and read. Reading and processing that type of content is not as clarifying for many, so some users turn to videos as a possible tool to learn. Videos are a great source of information since crucial points and concepts are made through a fast and simple manner. This content somewhat encourages people to keep digging and feeding that curiosity bug within them that is anxious to learn about FPGAs. The type of content that was reviewed can be separated into two different types: the lessons and the tutorials.

1) Lessons: In a question of minutes, a viewer can get to know some of the basic concepts that characterize an FPGA to the more complex ones, and what before seemed hard to understand now feels graspable and entices one's curiosity for more. Intel, for example, has a team of professionals responsible for a YouTube channel that presents a series of videos that explore the FPGA's basic concepts [6], applications, their programming languages and advantages given their use [7].

2) Tutorials: Another type of content that is searched for are video tutorials, such as [8, 9], where the content creator shows the viewers a step by step build-up of a proposed problem or system. Many content creators focus on this aspect and even have a series, where they teach their viewers how to implement a simple circuit, such as a two integer adder, to a more complex system, as for example a whole finite-state machine. These tutorials many times wind up being beneficial, considering that they are made with the purpose to aid the viewers build and entire project from scratch and learning about the process step by step, developing tools that will help them solve future problems or implement future designs.

Drawbacks: Tutorials often cloud viewers when showing a way to proceed with means to reach a final goal. Viewers are not always critical about the information that they collect, specially if they are still in the process of learning, leading to a very narrow set of solutions to a said problem. The content creators within this field are often experienced and familiarized with both the FPGAs and the programs used to develop an FPGA design, each with their own set of "tools" and "tool boxes". In other words, they themselves already possess knowledge and experience that leads them to take certain decisions and approaches when faced with an obstacle during the design process. Where, on the other hand, new pupils lack that experience making it hard for them to understand why such an approach was the selected one or even if there is any other. Even though videos are a great source of information, FPGAs are not easy to understand, and solemnly learning through videos, whereas these are about FPGA architectures and logic block or an implementation of a switch controlled LED, they may not be enough to fathom the entirety of the concepts that are needed, therefore additional bibliography and methods are required to further mature and develop this knowledge.

B. Projects and Activities

Sometimes, by having one getting a hands-on experience with the actual FPGA without knowing about their mode of operation is a good way to build-up some of the basic concepts of both the device's structure, programming and configuration. Many universities and engineering programs do carry out this strategy in practical or laboratory classes, as exemplified in [10], providing the students with some simple bibliography, or step by step instructions that serve as tutorials for a certain task in order to help the students familiarize with the device and tools that will help develop and finalize the project at hand.

Contrasting with the ideology of the method described in this paper, with this approach students learn about FPGAs by making use of these devices and participating in lessons, where the concepts are taught. The lessons exist to explain the basic concepts and present information, later evolving and building up that knowledge, step by step, to a higher level of complexity, whereas at this point, with all the given information, the student can already implement and develop a full project to later be evaluated on the matter.

Attracting students to develop an interest for this technology while applying a game related concept to it is not new, although using a different approach from the one proposed in this paper. An article by Jiménez-Fernández *et al.* [11] showcases a work done for an Advanced Digital Design course, that consisted in having the students that were enlisted to develop a game, through the use of VHDL (VHSIC Hardware Description Language) and FPGAs. But why a game? Well, its very hard to find a college student that somehow would not be enticed by this approach, because after all, its just a funner way to develop some of the knowledge that is intended to be conveyed in the course, and having it be exposed in such a way can greatly benefit the students productivity and learning process.

Drawbacks: It can be easier for the students to get answers for their question when conflicted with a problem, by either looking to question a tutor or their classmates, but this method is highly dependable of the quality of tutoring given to the students. This method, as mentioned before, is typically implemented within university-level of studies, and that itself can be considered a drawback since not everyone has the same accessibility to these means.

C. Books

Part of the process of learning about FPGAs is to get to know how and why they work, and books are a excellent source of information due to their technical nature and dense quantities of different topics, all related to these devices. Extensive, descriptive and in-depth reading might be the solution for some of the FPGA enthusiasts, but for many people it is hard to keep it together and fully take in all the information that is being conveyed. Bibliography that stood out can somehow be divided into two categories: bibliography focused towards FPGA architectures and bibliography focused towards FPGA programming and configuration.

1) FPGA architecture literature: Books, such as [12], highly explore the main blocks that exist in an FPGA board (regarding a few different manufacturers), showcasing different takes on those block's architectures and evaluating their impact in the device's performance, power consumption and area-efficiency, whereas in [13] these parameters are studied within the modern commercial FPGA architectures to predict future trends in the field. Withal, a book from the author Andrew Moore [14] was written in a simpler and fun way, attempting a friendlier approach to explaining the complexity that FPGAs can pose.

2) Programming FPGA literature: Understanding how one of these devices is configured and what programming languages are used to do so is perhaps the hardest of the approaches when seeking to learn about FPGAs. This second category not only refers to HDL (Hardware Description Language) teaching books, for languages such as Verilog and VHDL [15, 16], but also comprises FPGA data sheets which are highly important due to each board having its own configurations and perks. Sometimes its inevitable to talk about FPGA programming without introducing its concepts and some authors end up combining both the categories mentioned into a single book [17].

Drawbacks: Information conveyed in such a dense fashion does not attract many people into learning about these devices, making this approach one of the least favorable. Although the information within these resources can prove to be valuable its very hard to follow and read the entirety of a technical book, even though some try to make it a little less tedious. Being

the book about the FPGA's general architecture or about the introduction to the programming options available to FPGA design, with this solution in place the main problem still stands.

III. Proposed game

With this paper, the objective is to present the work being done towards a game-based solution to learn about FPGAs. The question that propels this project forward is "How do to get more people to learn about FPGAs in a fun, easy and simple way?".

The game will be based in a level progression system, where the next level is only unlocked after the player reaches the goal proposed in the present level. With this in mind, and by tending to the main objective, which is to teach the basic concepts of FPGAs in order to help more people join and contribute within the reconfigurable device community, these concepts will be introduced through a bottom-up based design. The plan for the first levels is to get players to familiarize themselves with the simpler circuits such as multiplexers, decoders, registers and flip-flops, for example, and with that, to help them devise a working design that would be able to be implemented in an FPGA. Future levels will present harder challenges that will be solvable through the use and the combination of knowledge acquired from previous levels.

This game aims to teach players about the main FPGA blocks and how they can be used to implement the aforementioned circuits. These main blocks would be the Configurable Logic Blocks (CLBs), Input/Output Blocks (IOBs), RAM and Multiplier Blocks, Digital Clocks Managers (DCMs), and the Switch Matrix (SM) and interconnect network. Knowing the available blocks and respective functions of a given FPGA helps designers know what are the options and possibilities within the given device.

The proposed game will have a notebook mechanic to it, in the sense that, when a player completes a level, a "Lesson" is unlocked and saved in a "Notebook". Lessons are pages within a Notebook that will exist in the game to provide the users with information about a certain block, function or concept. The Notebook can be consulted at anytime and will be made available to the player within the level and even in the game's Main Menu, to assist them at overcoming other challenges ahead in case the player wishes to review the material from previous levels.

When a level is complete and the Lesson is saved to the Notebook, the player will also gain "Knowledge". In order to explain this concept in a simpler way, an example will be used. A player that just completed a level where it was required to design a multiplexer, the multiplexer will be unlocked as a block, where in the future, when the player needs to implement that logic into the system, he no longer will have to design the multiplexer from scratch and can just add the logic block into the design to perform the same function.

Making way to the final goal, the game is still at a prototype level of development. For these first steps a tool named *GDevelop 5* [18] is proving to be quite effective to layout what could be a first test to this game. Designing a game from scratch is challenging and the importance given to game assets such as buttons, animations and sprites are sometimes overlooked, therefore, the first step into building the prototype was to create and design sketches of the game menus, such as the main menu and the level selection menu, and their respective buttons, sprites and animations. A second challenge will be to load and link all of these aspects together into the game canvas and be able to transition between both menus, and hopefully transition into a selected level to start playing. The last phases of production are to design as many levels as possible and their respective assets, one by one, in order to put up playable levels that can be used to learn about FPGAs and their components.

After a few weeks of both learning and interacting with the tool utilized, a few drawbacks were encountered regarding the quality of the sprites while testing the game and the tool's limited publishing method for users with free accounts, although the software does offer a simple and easy graphic and modular method to build a game. With this in mind other tools were searched for developing a final product and *Phaser* [19], a HTML5 game framework, that works for both desktop and mobile game development, does resolve the aforementioned drawbacks but heavily relies on JavaScript knowledge to develop a fluid and neat working game.

As to this point the answer is still left unattended, but now that the game was somewhat presented, or at least the plan for it, this paragraph will justify the choice made regarding the method taken into account. In [20], the learning process is described as a process that is highly built upon an individual's active participation in the task at hand, and a process that is strongly affected by one's experience while executing that said task. Because practicing, experiencing and struggling directly with the challenges faced while doing an activity generates a number of questions that when answered and solved fully complete and close the learning process. Therefore, due to their nature, games intrinsically require this activeness, offering a favorable environment, in this case, to learn about FPGAs.

IV. Conclusion

This paper proposes a new approach for FPGA learners, and enthusiasts, to make use of when curious about these devices. This approach does not invalidate the use of the other mentioned methods, nor does it prove to be more functional. The suggested method is built upon the use of the learning process that benefits from the focus and attention that games require from players, given that the game in mind is pedagogical. Yet a game-based method can be a great starting point into captivating the curiosity and interest of newer public. Withal, the combined use of all the mentioned methods may be the best approach towards fully learning and improving the related skills that are needed to understand and program FPGAs.

The plan towards this paper's goal is established, but the game still remains under development and many of the aspects here explored might still be iterated on.

Acknowledgment

This work was partially financed by Portuguese Agency "Fundação para a Ciência e a Tecnologia" (FCT), in the framework of project UIDB/00066/2020.

References

[1] Kani K. Guner, Taylan O. Gulum and Burcu Erkmen. "FPGA-Based Wigner-Hough Transform System for Detection and Parameter Extraction of LPI Radar LFMCW Signals". In: *IEEE Transactions on Instrumentation and Measurement* 70.c (2021). ISSN: 15579662. DOI: 10.1109/TIM.2021.3060584.

[2] Divya Singh, Sunita Prasad and Sandeep Srivastava. "Implementation of artificial intelligence cognitive neuroscience neuron cell using adaptive velocity threshold particle swarm optimization (AVT-PSO) on FPGA". In: *2017 6th International Conference on Reliability, Infocom Technologies and Optimization: Trends and Future Directions, ICRITO 2017* 2018-January (2018), pp. 548–552. DOI: 10.1109/ICRITO.2017.8342488.

[3] Timothy Prickett Morgan The Next Platform. *The Fourth Wave Of FPGA Compute*. 2020. URL: https://www.nextplatform.com/2020/02/05/the-fourth-wave-of-fpga-compute/ (visited on 17/05/2021).

[4] *Make it True - Google Play*. 2021. URL: https://play.google.com/store/apps/details?id=com.ViacheslavRud.Circuit&hl=pt&gl=US (visited on 19/05/2021).

[5] David F. Bacon, Rodric Rabah and Sunil Shukla. "FPGA programming for the masses: The programmability of FPGAs must improve if they are to be part of mainstream computing". In: *Communications of the ACM* 56.4 (2013), pp. 56–63. ISSN: 00010782. DOI: 10.1145/2436256.2436271.

[6] Intel. *Basics of Programmable Logic: FPGA Architecture*. Youtube. 2018. URL: https://www.youtube.com/watch?v=jbOjWp4C3V4 (visited on 17/05/2021).

[7] Intel. *FPGA Design*. Youtube. 2020. URL: https://www.youtube.com/watch?v=0Ho4rDswOeE&list=PL0pU5hg9yniZ2ka-XBXROXNR0pAEAEFCB (visited on 17/05/2021).

[8] Greidi Ajalik. *Beginners Guide to get started with Xilinx FPGA Programming*. Youtube. 2018. URL: https://www.youtube.com/watch?v=WY-F3knih7c&list=PLqOe1_kmWOx33G3gOzQSajSdrTtW9shBO (visited on 17/05/2021).

[9] Invent Box Tutorials. *Learn FPGA*. Youtube. 2019. URL: https://www.youtube.com/watch?v=vjBsywUSKWk&list=PL2935W76vRNGRtB09yXBytO6F3zSZFZGr (visited on 17/05/2021).

[10] Karyono and Arya Wicaksana. "Teaching Microprocessor and Microcontroller Fundamental using FPGA". In: *2013 International Conference on New Media Studies, CoNMedia 2013* (2013). DOI: 10.1109/conmedia.2013.6708541.

[11] Carlos Jesús Jiménez-Fernéndez et al. "Learning VHDL through teamwork FPGA game design". In: *Proceedings - 2020 14th Technologies Applied to Electronics Teaching Conference, TAEE 2020* (2020), pp. 1–5. DOI: 10.1109/TAEE46915.2020.9163756.

[12] Stephen D. Brown et al. *Field-Programmable Gate Arrays*. Boston, MA: Springer US, 1992. ISBN: 978-1-4613-6587-7. DOI: 10.1007/978-1-4615-3572-0.

[13] Ian Kuon, Russell Tessier and Jonathan Rose. *FPGA architecture: Survey and challenges*. Vol. 2. 2. 2007, pp. 135–253. ISBN: 9781601981264. DOI: 10.1561/1000000005.

[14] Andrew Moore. *FPGAs For Dummies®*. 2nd Intel®. John Wiley & Sons, Inc., 2017, p. 40. ISBN: 9781119390497. URL: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/misc/fpgas-for-dummies-ebook.pdf.

[15] Bryan Mealy and Fabrizio Tappero. *Free Range VHDL*. 2018, p. 194. URL: http://www.freerangefactory.org/.

[16] M. Morris Mano and Michael D. Ciletti. *Digital Design With an Introduction to the Verilog HDL, VHDL, and SystemVerilog*. Pearson Education, Inc., 2018. ISBN: 9780134549897.

[17] Evgeni Stavinov. *100 Power Tips for FPGA Designers*. 2011. ISBN: 9781450775984.

[18] Photon Storm Ltd. *Phaser*. URL: https://phaser.io (visited on 24/06/2021).

[19] Florian Rival. *GDevelop5*. URL: https://gdevelop-app.com (visited on 24/06/2021).

[20] Donald C. Thatcher. "Promoting Learning Through Games and Simulations". In: *from the SAGE Social Science Collections* (1987), pp. 183–205. ISSN: 1059-6011.