

FPGA BASED REPROGRAMMABLE MAIN CIRCUIT BOARD AND AUXILIARY CIRCUIT BOARD DESIGN

M. F. ILASLAN*, T. C. AKINCI

Department of Electrical Engineering, Istanbul Technical University,
Ayazaga Campus, Maslak, 34469, Istanbul, Turkey
*Corresponding Author: ilaslan18@itu.edu.tr

Abstract

Nowadays, designers have to use either design boards or ready-made development boards in electronics applications. Although development cards are very practical for amateur users, they can be insufficient for professional applications. Designers is a word that covers a wide area. Because professional and amateur users are the equivalent of this word. The objectives of both types of users are different. While amateur users want to improve themselves in certain areas, professional users want financial gain through various projects. Therefore, it is aimed to make a circuit board that can combine two types of users in this study. Students, given as examples to amateur users, will be able to improve themselves in terms of design and software in the field of embedded systems. Professional users can sign projects that can contribute to the industry with Field Programmable Gate Arrays (FPGA). Therefore, the design and application of the boards gain prominence for this purpose. Unlike development cards, a stable, simple, and professional card design-application which is more economical in terms of professional applications have been made. Main and auxiliary circuit boards design based on FPGA have been designed and printed in this study. That study has been planned as a new and practical FPGA-based development board. In this way, all users and designers will be able to provide the main circuit board and auxiliary circuit board specified in this study, economically and realize their projects faster than other development cards. In addition, amateur users are provided with a basic opportunity to improve themselves on FPGAs and various gate arrays used in many areas. Both developed boards were found to be more efficient in terms of processor performance than the existing development boards by the virtue of FPGA.

Keywords: Development boards, FPG, Microcontroller, PCB design.

1. Introduction

With the rapid development of science and technology, the reduction of integrated circuit elements to Nano dimensions has reduced the physical dimensions of electronic devices and increased the accessibility to those devices. [1]. Thus, the applicability of electronic systems has increased and enabled designers to produce projects in large areas. In this period, applications in electronic systems have been developed by using Peripheral Interface Controller (PIC) integrated processors for many years. In cases where PICs are insufficient, ready-made development boards have been preferred by designers. The lack of these ready-made development cards in professional applications has enabled designers to use more stable, reprogrammable and powerful processors [2, 3].

In electronic applications, it is necessary to ensure the synchronous relationship between the signals. Therefore, phase measurement is a requirement. Conventional electronic systems for time measurement are designed by using a classical mixed-signal approach [4]. Due to the components that are reprogrammed, stable, and robust such as FPGAs, the relevance of digital architecture has increased [5].

Field Programmable Gate Array (FPGA) is one of the integrated processors with this type of architecture and features. These integrals, commonly known as processors, are called microprocessors and microcontrollers because of their size and different processing capabilities [6]. While a small part of the microprocessors is used in computers, a large part of them is embedded in electronic circuit boards. Processors, which have widely used areas, are economic and they can be performed certain commands but the most important disadvantage of these processors are their unorganized architectures. For these reasons, processors with general-purpose architecture and reconfigurable architecture are preferred by designers in different applications. In this study, an FPGA based development board was designed and its performance was evaluated with the existing boards in terms of application areas.

Besides successful performances of FPGAs, there are also some difficulties. The difficulty of the design of FPGAs compared to other electronic circuit elements is that there is no connection of the block in the structure within the FPGA at the beginning of the project [7]. Thus, the strong logic block knowledge is needed by the users.

In addition, another disadvantage is an issue that concerns all electronic devices in general. Thanks to the advancing technological developments, the new types of gate array series have a better performance than the previous generation in every few years. However, when these integrated devices are considered in terms of price comparison, previous generations keep their advantages for development cards. Besides these features, there are differences in terms of many technical features such as logic cells and input/output pin numbers, Digital Signal Processing (DSP) performance, etc. In the table below, the 7 series families' comparison is shown in the Table 1.

Table 1 shows the differences between 7 series family models of Xilinx Company. There are also some differences that will be shown in Fig. 4, between FPGAs and microcontrollers.

Processor Architecture is one of the substantial parts of embedded systems. Microprocessors are the central processing unit (CPU) of the computers we use today. This is where all programs and data are processed. In this part, processor

architectures will be examined in two basic groups that are general purpose architectures and rearrange architectures.

Table 1. 7 Series families comparison [8].

Max. Capability	Spartan-7	Artix-7	Kintex-7	Virtex-7
Logic cells	102 K	215 K	478 K	1.955 K
Block RAM	4.2 Mb	13 Mb	34 Mb	68 Mb
DSP slices	160	740	1.920	3.600
Transceivers	-	16	32	96
Transceiver speed	-	6.6 Gb/s	12.5 Gb/s	28.05 Gb/s
Serial bandwidth	-	211 Gb/s	800 Gb/s	2.784 Gb/s
PCIe interface	-	x4 Gen2	x8 Gen2	X8 Gen3
I/O pins	400	500	500	1.200
Package options	Low cost, Wire Band	Low cost, Wire Band	Bare-Die Flip-Chip	High Perf Flip-Chip

General Purpose Architecture is mentioned a computer that could do any calculations without changing hardware was designed by John Von Neumann in 1945. The approach of Von Neumann is [9];

- The memory that stores the program and the memory that stores data have collaborated, unlike Harvard architecture.
- The control unit consists of a program counter that stores the address of the next command to be processed.
- Arithmetic-Logic Unit (ALU) is the unit where commands are executed.

Harvard architecture is a CPU architecture created by separating the channels of data and commands to the Central Processing Unit. The schematic diagram of Harvard and John Von Neumann Architecture (VN) is shown in Fig. 1.

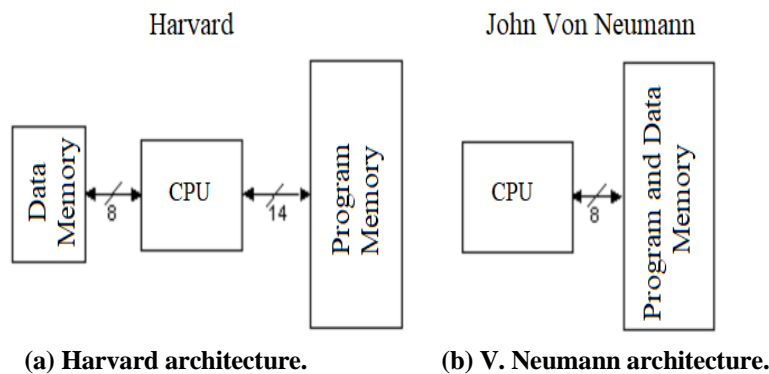


Fig. 1. Architecture types of MCU [10].

Rearrange Architecture If the boundary lines of the application to be made is certain and general-purpose, the usage of processors with non-reorganized architecture makes sense both financially and easily in terms of the solution of the application. However, the best choice for a specific application to run on an embedded system is designing an optimized processor. Because Application-Specific Integrated Processors (ASIP) are built on the instruction set chip for the

application and they increase the performance rate with the approach of adapting the hardware to the application [9]. The VN architecture, which is not considered good in performance due to its inability to perform parallel calculations, is called a general-purpose processor architecture. Although the VN architecture is better than ASIP in terms of flexibility, it falls behind in performance. One of the best examples of reconfigurable architectures is Field Programmable Gate Arrays (FPGA). FPGAs have been produced after microprocessors and have grouped differently from other processors thanks to their different internal structures. This type of equipment, which is produced in a constant manner during the production stage, is able to change its structure according to the application and its functionality is realized at a very low level of granular structure. For example, the hardware can be modified by adding or removing a NAND port with one or two inputs to the hardware. Fine-grained rearrangeable hardware is basically represented as Programmable Logic Devices (PLD). That includes Programmable Logic Arrays (PLA), Programmable Array Logic (PAL), Complex Programmable Logic Devices (CPLD) and FPGA [9]. The main difference between PAL and PLA is that PLA is comprised of arrays of programmable AND and OR gates, while PAL is comprised of programmable AND and fixed OR gates.

In this study, both schematic and printed designs of the main circuit and auxiliary circuit board were made by using Altium Designer © package program.

2. Field Programmable Gate Arrays (FPGA)

Although many users define FPGAs are ordinary processors, FPGAs are an advanced version of many processors. Firstly, FPGAs have been produced by Xilinx in 1985. They can be quickly re-adapted and programmed with the help of electrical signals [11]:

- Logic Cell: A logic cell consists of a Look-up Table (LUT), one D Flip-Flop and one 2-1 Multiplexer (MUX).
- Input/output blocks.
- Interconnections.

The simple FPGA structure is shown in Fig. 2. Also, the connections, blocks and cells in the structure of the FPGA are shown in Fig. 3.

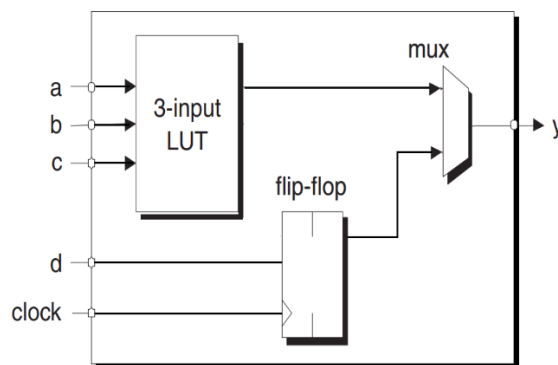


Fig. 2. Simple structure of the logic block [12].

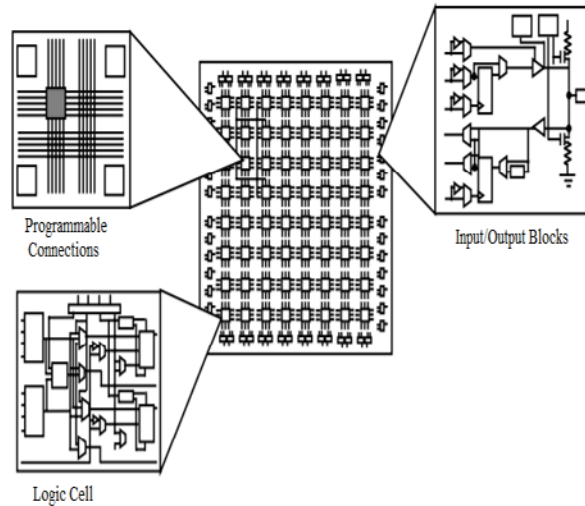


Fig. 3. Internal structure of FPGA [13].

Choosing the right devices for any designed system is difficult in many ways, such as economic, efficiency, and compatibility. However, the system must always have a processing engine, memory and logic [14]. Microprocessors already have the necessary requirements and fixed instructions for the operating system that users want to program. These instructions are linked to their respective block within the processor. Its internal structure consists of wired systems, which is completely different from FPGAs. Because the cable structure prevents the reprogrammable structure of the FPGAs. The internal structure of the FPGAs can be likened to a rail system with the whole system connected to each other by switches. The way to determine the point to be reached will be thanks to the system created by the programmer. In short, the main difference between microprocessors and FPGAs is that microprocessors can do most of the actual processing, but more specific tasks are done by FPGA blocks. For example; the microprocessor working to process the image transmitted from a camera that shoots at 24 frames per second, processing each frame in turn; FPGAs process multiple frames at the same time thanks to their parallel processing capability. This provides advantages in terms of time, speed and efficiency. In addition, FPGAs with the necessary hardware and graphical processing units to increase efficiency in artificial neural network applications can be performed quickly thanks to their parallel processing and energy efficiency maximization features [15].

Although FPGAs have a different structure than MCUs, they can be used together. A harmonious study can be achieved by using one of the appropriate communication protocols such as UART that will ensure communication between each other. With this kind of work that is mentioned in detail at reference [16], it can alleviate intensive cycles on MCUs.

The microcontrollers contain the necessary components, such as memory and timers. It is programmed to perform simple and standard operations by assisting other hardware elements. FPGAs are more flexible than micro-controllers. This flexibility can be advantageous or disadvantageous in some applications. One of

the reasons why it is called a flexible structure is that it is reprogrammable, and that causes FPGAs to spend more energy. The relationship between the microcontroller and FPGA is shown with the Venn diagram in Fig. 4.

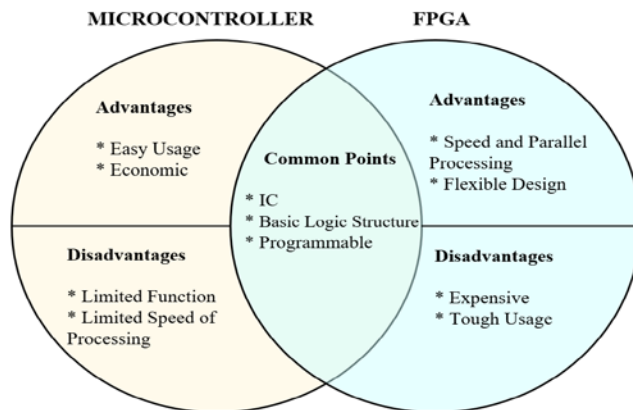


Fig. 4. The Venn diagram between microcontroller and FPGA.

FPGAs are compared with ASIPs as well as microcontrollers. Application-Specific Integrated Circuits (ASIC), which means the print circuit of ASIPs, cannot be reprogrammed. They are special circuits that are designed only for the specific application. The major advantages of FPGA-based circuits over ASICs are being reprogrammable and not having reproduction costs. Furthermore, the programmability of FPGA-based circuits makes it possible to develop designs in areas that do not allow for hardware changes. Figure 5 shows the comparison of FPGA-based and ASIC circuits. Also, the differences between ASICs and FPGAs are shown in Table 2.

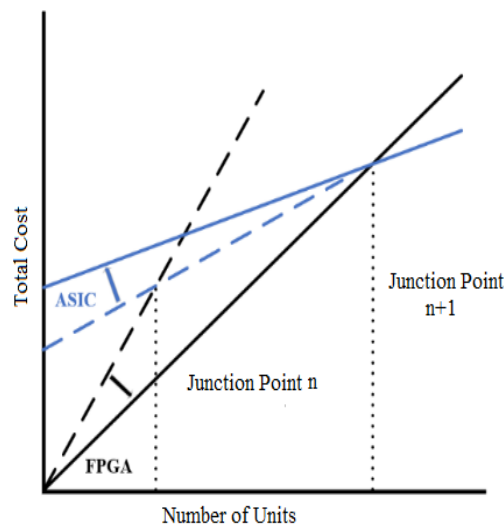


Fig. 5. Comparison of FPGA and ASIC intersections [17].

Table 2. The differences between ASICs and FPGAs [18].

No.	ASICs	FPGAs
1	Have permanent circuitry.	Have reconfigurable circuitry.
2	Have higher entry barrier.	Have lower entry barrier.
3	Have shorter design cycles.	Have higher design cycles.
4	Have lower power consumption.	Have higher power consumption.

3. Design and Application of Main and Auxiliary Circuit Boards

The Spartan 3E model of Xilinx has increased the amount of logic per input/output in comparison to its previous series and it reduces the cost per logic cell significantly. It has a 90nm process technology. One of the main reasons why FPGA is preferred as a motherboard processor is that it can be used in many areas such as home networks, digital equipment and broadband applications with its low cost and strong functionality. In general, the pins in the architectural structure are as follows:

- Configurable Logic Blocks (CLBs): CLBs which are comprise of several Logic cells into a block and add special-purpose circuitry like adder or subtractor carry chain [19]. In addition to data storage, they perform a wide range of logical functions. They also include flexible LUT structures.
- Input/output Blocks: They control the data flow between the input/output pins and the internal logic of the device.
- Block Random Access Memory (RAM): It enables the storage of data in the form of 18-bit double-port blocks.
- Multiplier Blocks: It accepts two 18-bit binary numbers as input and calculates the result.
- Digital Clock Manager (DCM): It provides self-adjusting, fully digital solutions for distributing, delaying, multiplying, dividing and phase-shifting clock signals.

The microprocessor used in the main circuit board of this study is the SPARTAN 3E (XC3S250E) model of Xilinx company. This processor has a total of 100 pins with 25 pins on each side. Twenty-eight of these pins are reserved for input/output pins to use in the auxiliary circuit board application. In addition, five Light Emitting Diodes (LED) are connected to the input/output pins for operation on the main circuit board. In order to avoid noise problems, capacitors are placed at the points leading to the ground. The frequency stability is also ensured by the 40MHz crystal oscillator.

Xilinx PROM (Programmable Read-Only Memory) circuit board is also used on the main circuit board. The compatibility of the elements on the circuit and the programming of the FPGA with the Joint Test Action Group (JTAG) operator have an important role in the operation. The programmable configuration in the system is the platform flash series of PROMs. Platform Flash PROM is a reprogrammable NOR flash device. These PROMs that available in a density of 1 to 32 Mb, provide an easy-to-use, cost-effective and reprogrammable method for storing large Xilinx FPGA configuration bitstreams. When external PROMs were used in the past, they left their place to Erasable Programmable Read-Only Memory (EPROM) and Electronically Erasable Programmable Read-Only Memory (EEPROM).

In the auxiliary circuit application, outputs were taken from the input/output pins of the processor of the main circuit board. Pi filter circuit is added to each output. This is due to the use of a passive and efficient filter on a non-powerful

auxiliary circuit board. The capacitor connected in parallel to the output reduces the voltage fluctuations between the load terminals also the coil connected in series reduces current fluctuations. Figure 6 shows the printed motherboard.

One of the important choices of electronic circuit designs is oscillator choices. If a circuit design with a very high frequency rate is not made, according to the designer's experience, as in this study, standard-value commonly used oscillators can be preferred. In oscillator selections, there are many factors including capacitors used on circuit, capacitors to be connected to oscillator, gain margin. In this study, the paper of STMicroelectronics is shown in [20] company was followed. The schematics and PCB drawing of the circuits is shown in *Appendix A*.

Analog-to-digital converters used in the auxiliary circuit board must be compatible with the FPGA processor used in the main circuit board. Physical differences in the print design for pin fit are provided by female and male connections. Analog-to-digital and digital-to-analogue converters are selected as 12 bit to work with the processor.

Datasheets are examined and 8-channel ADC128S022 circuit element for the analogue-to-digital converter and AD5328A circuit element for digital-to-analogue converter are selected. The mainboard is shown in Fig. 7 and the auxiliary board is shown in Fig. 8. The assembled version of the boards is shown in Fig. 9.

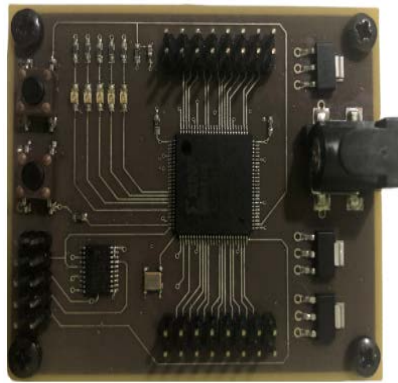


Fig. 7. Main board.

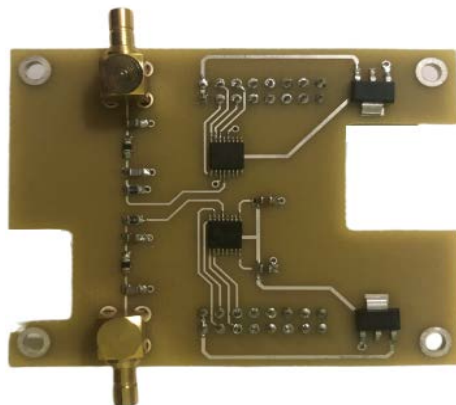


Fig. 8. Auxiliary board.

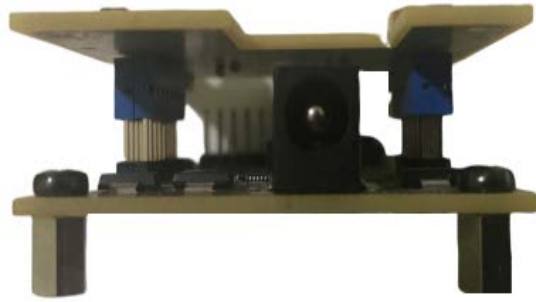


Fig. 9. Front view of the assembled version of the boards.

In the literature, there are also projects in which FPGAs are used as based processors in health and safety systems besides their kits [21]. When the study given in the reference is compared with the prototypes of this study; the prototypes of this paper refers to a more general study. Furthermore, projects prepared with auxiliary circuit boards such as image data processing and distance calculation etc. have a wide range of applications from the defence industry to transportation systems.

An application was made with VHDL, which is the programming language of the FPGA component used, and the results are shown below. That application is making full adder by using two half adders. The programming codes are shown in Appendix B. The simulation part of the Xilinx ISE Project Navigator program was used to simulate the full adder. The RTL schematic of Full Adder and the simulation result is shown in Figs. 10 and 11 respectively.

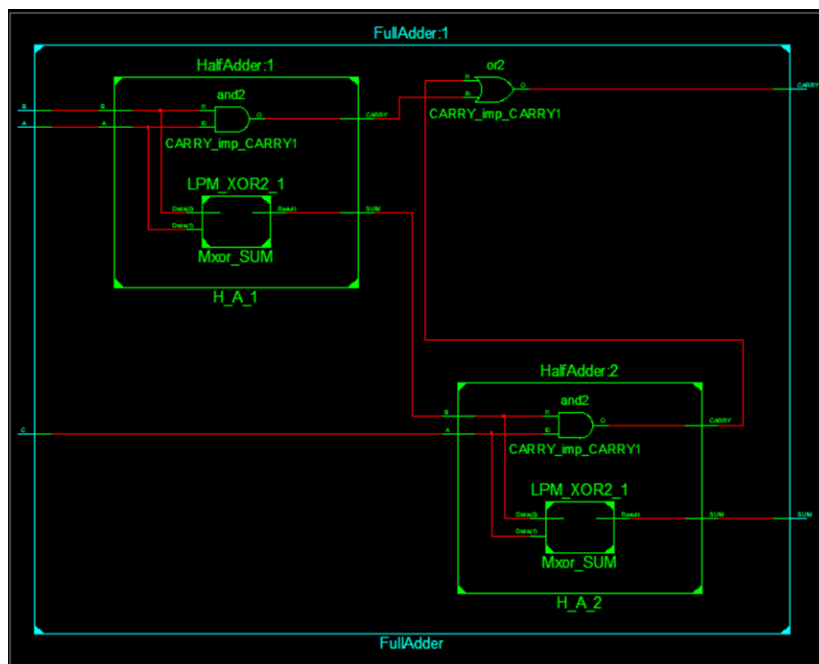


Fig. 10. The RTL schematic of full adder.

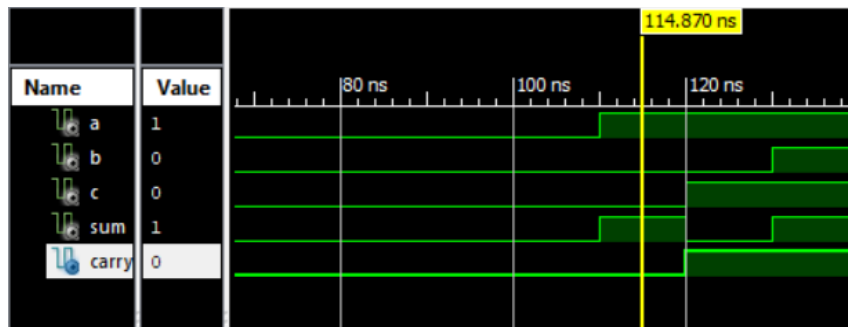


Fig. 11. The simulation result when a is 1, b and c are 0.

Figure 11 shows the first simulation result of the application with the circuit that are mentioned in this study. This is the first figure when the a is 1, b and c are 0.

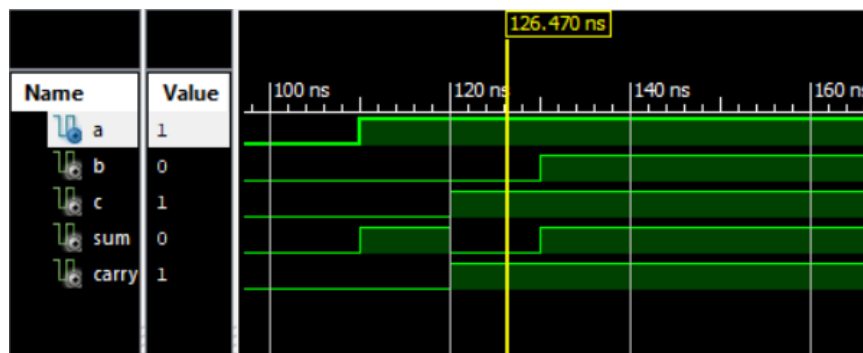


Fig. 12. The simulation result when a and c are 1, b is 0.

The other simulation results could be understood. For example, values of a and c are 1, b is 0 in the previous period. The values of a , b , c , sum and carry are 1 in the next period.

4. Conclusions

FPGAs are one of the most efficient microprocessors in the industry. Also, FPGA is more advantageous and successful in many areas against its opponents such as CPLDs and ASICs. Because of FPGAs' features that are hardware quality, flexibility, special architecture, lightweight, parallel processing capability, and economical, it makes the use of FPGAs more practical in all areas that require real-time processing. In this study, the main circuit board is designed based on the SPARTAN3E processor. Considering the disadvantages of collecting circuit elements in other education kit boards, auxiliary board design and suggestions were prepared. Analog-Digital and Digital-Analog converter circuit is designed as an auxiliary board model.

When the main circuit board is analysed, it becomes an alternative to the initial development cards available in the market with its size, the width of its usage areas. In this study, it does not mean that the circuit board has less features than other development boards due to the number of components. It is a great feature that the

ADC-DAC converter, which is the auxiliary circuit board, and the advantage of each user to design their own auxiliary circuit board. All this work will guide the users' project designs and provide the design of new auxiliary circuit boards.

As a result of this application, it is thought that the monolithic kits designed for education take up more space and size. Therefore, it supports small, simple designs to be quick to learn and to minimize the damages caused by distracting and vulnerable circuit elements on the main circuit board for each project to be developed. In addition, thanks to its designed structure, it is half the size of Xilinx development cards available in the market. It is budget-friendly thanks to its multi-story and fragmented structure. Compared to the Arduino development boards, it enables it to perform more operations faster and successfully due to the difference in the processor structure. In conclusion, simply designed, easy to understand alternative circuits have been produced against complex design education kits. The compatibility of the auxiliary circuit boards with the mainboard differs from other examples in that it can develop a specific design for each application to be designed.

Abbreviations

ALU	Arithmetic-Logic Unit
ASIC	Application-Specific Integrated Circuits
ASIP	Application-Specific Integrated Processors
CLB	Configurable Logic Block
CPLD	Complex Programmable Logic Devices
CPU	Central Processing Unit
DCM	Digital Clock Manager
DSP	Digital Signal Processing
EEPROM	Electrically Erasable Programmable Read-Only Memory
EPROM	Erasable Programmable Read-Only Memory
FPGA	Field Programmable Gate Arrays
JTAG	Joint Test Action Group
LED	Light Emitting Diode
LUT	Look-up Table
MUX	Multiplexer
PAL	Programmable Array Logic
PCB	Printed Circuit Board
PIC	Peripheral Interface Controller
PLA	Programmable Logic Arrays
PLD	Programmable Logic Devices
PROM	Programmable Read-Only Memory
RAM	Random Access Memory
VN	John Von Neumann Architecture

References

1. Haselman, M.; and Hauck, S. (2010). The future of integrated circuits: A survey of nanoelectronics. *Proceedings of the IEEE*, 98(1), 11-38.
2. Al Mashhadany, Y.I. (2012). Design and implementation of electronic control trainer with PIC microcontroller. *Intelligent Control and Automation, Scientific Research Publishing*, 3(3), 222-228.

3. Barsoum, N. (2010). Speed control of the induction drive by temperature and light sensors via PIC. *Global Journal of Technology and Optimization*, 1, 35-59.
4. Mitra, J.; and Nayak T.K. (2018). An FPGA-based phase measurement system. *IEEE Transactions on Very Large-Scale Integration (VLSI) Systems*, 26(1), 133-142.
5. Heshmatpanah, J.; Boroumand, S.; and Masouleh, M.T. (2014). FPGA design and implementation for real time vision applications on NTACO mobile robot. *Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*. Tehran, Iran, 172-177.
6. Wolf, W.H. (2004). *FPGA-based system design* (1st ed.). New Jersey: Prentice Hall PTR.
7. Unsalan, C.; and Tar, B. (2017). *Digital system design with FPGA: Implementation using Verilog and VHDL* (1st ed.). New York: McGraw-Hill Education.
8. Xilinx. (2018). 7 Series FPGAs data sheet. Retrieved April 30, 2020, from https://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf.
9. Ozdamar, S.G. (2007). Bilgisayar Mimarisinde Yeni Yaklaşımlar: Tekrar Düzenlenebilir Mimariler. Retrieved June 20, 2019, from https://web.itu.edu.tr/orencik/BilgMimYenYakl2007/Selcuk_G_Ozdamar/Tekrar_Duzenlenebilir_Mimariler_Rapor_504061529.pdf.
10. Matic, N. (2000). *The PIC microcontroller book 1* (1st ed.). Belgrad: Mikroelektronika.
11. Krishna, G.; and Roy, S. (2017). Fundamentals of FPGA architecture. *Technical and Scientific Publisher*, 2, 12-30.
12. Maxfield C. (2004). *The design warriors's guide to FPGA's* (1st ed.). Oxford, UK: Elsevier.
13. Ponce-Cruz, P.; Molina, A.; and MacCleery, B. (2016). *Fuzzy logic type 1 and type 2 based on labVIEW™ FPGA*. Switzerland: Springer International Publishing.
14. Parnell, K.; and Bryner, R. (2004). Comparing and contrasting FPGA and microprocessor system design and development, *Xilinx WP213*, 1(1).
15. Shawahna, A.; Sait, S.M.; and El-Maleh, A. (2018). FPGA-based accelerators of deep learning networks for learning and classification: A Review. *IEEE Access*, 7, 7823-7859.
16. Shrestha, P.; Subedi, B.; Girard, M.; Parikh, C.; and Kandalaft, N. (2020). Wireless communication between FPGA and microcontroller. *10th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, 0402-0405.
17. Trimberg, S.M. (2015). Three ages of FPGAs: A retrospective on the first thirty years of FPGA technology. *Proceedings of the IEEE*, 103(3), 318-331.
18. Moore, A.; and Wilson, R. (2017). *FPGAs for Dummies*®, (2nd Intel® Special Ed.). Hoboken, New Jersey: John Wiley & Sons, Inc.
19. Sass, R.; and Schmidt, G.A. (2010). *Embedded systems design with platform FPGAs principles and practices*. Burlington; Morgan Kaufmann, Elsevier.
20. STMicroelectronics. (2020). AN2867 Application note, (Rev 12). Oscillator design guide for STM8AF/AL/S, STM32 MCUs and MPUs. Retrieved May 1, 2020, from https://www.st.com/resource/en/application_note/cd00221665-oscillator-design-guide-for-stm8af-al-s-stm32-mcus-and-mpus-stmicroelectronics.pdf.

21. Jafari, A.; Ghovanloo, M.; and Mohsenin, T. (2017). A real-time embedded FPGA processor for a stand-alone dual-mode assistive device. *IEEE 25th Annual International Symposium on field-programmable custom computing machines (FCCM)*. Napa, CA, USA, 199.

Appendix A

Schematic Designs of the Circuits

In the appendix part, both schematic designs and 3D representation of main circuit board are added, to provide users to take full advantage of this study. All designs were drawn by using Altium Designer-PCB Design Software ©.

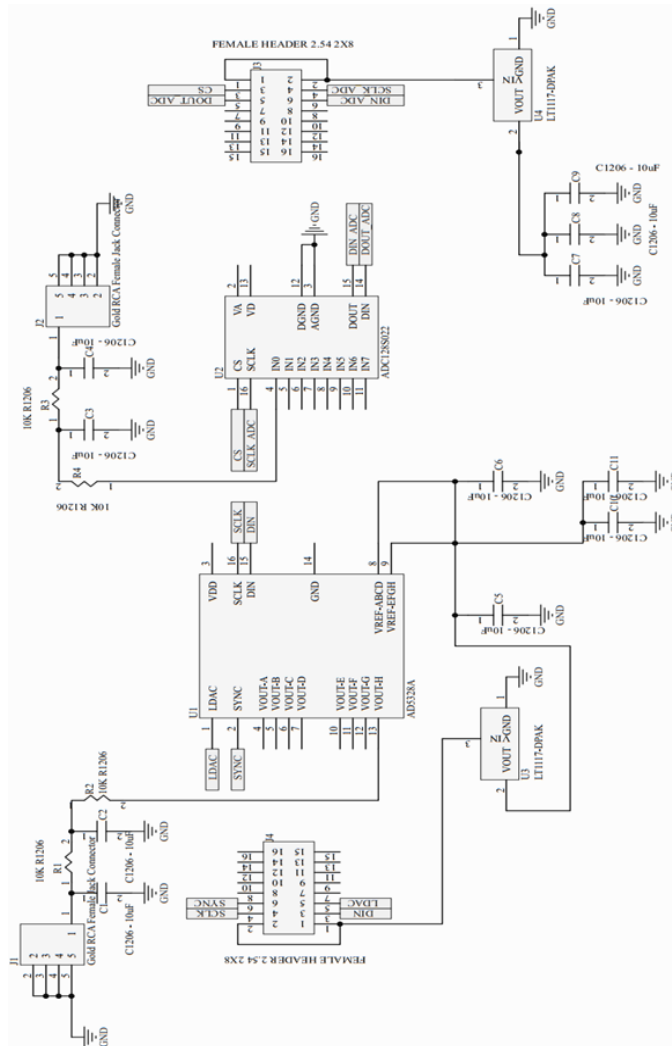


Fig. A-1. Schematic design of the auxiliary circuit board.

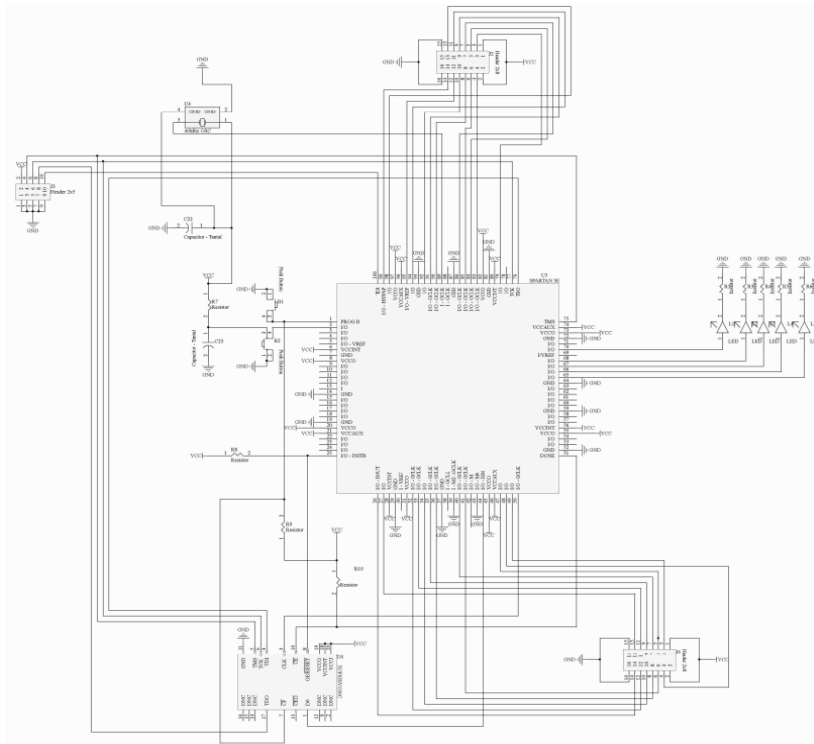


Fig. A-2. Schematic design of the main circuit board.

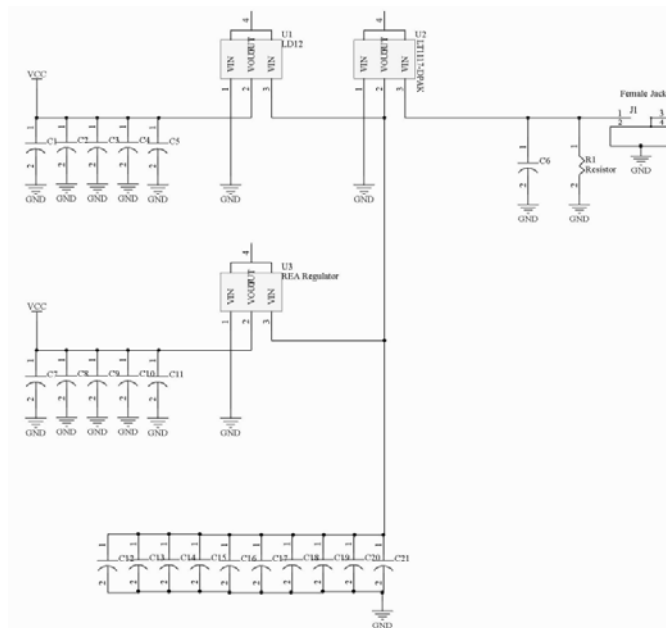


Fig. A-3. Schematic design of main board's regulators.

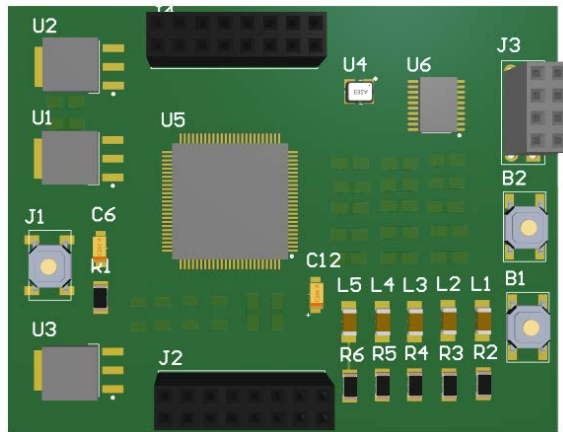


Fig. A-3. 3D representation of components layout of the main circuit board

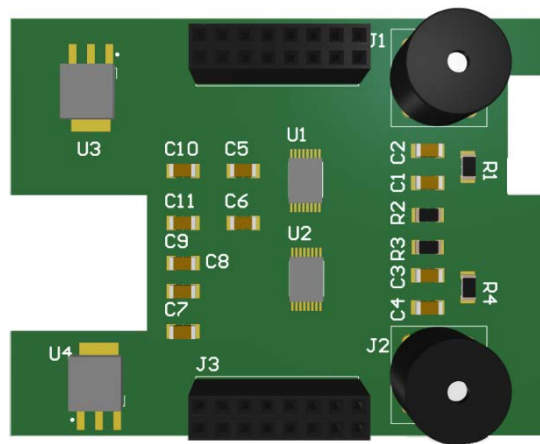


Fig. A-4. 3D representation of the auxiliary board

Appendix B

Application Codes

The point that distinguishes electronic engineering from all other engineering is the support of the physical product that is revealed with software. For this reason, a simple application has been made for the circuit whose design and printing mentioned in this study. Thus, users are informed about the VHDL programming language. Simple information about the projects that can be done with the product mentioned in the study is given. Simulation codes are written separately.

```

32     A:    in std_logic;  -- input 1
33     B:    in std_logic;  -- input 2
34     SUM:  out std_logic; -- output
35     CARRY: out std_logic -- output
36 );
37 -- component declaration have to be done btw architecture and begin
38 end component;
39
40 begin
41 -- by calling components, utilize the hardware has already done
42 H_A_1: HalfAdder -- calling first component or (HalfAdder) by generating label
43 port map(
44     A=>A,      -- from HalfAdder part to FullAdder part
45     B=>B,      -- the values are assigned from smaller to bigger
46     SUM=>sum1,
47     CARRY=>carry1
48 );
49
50 H_A_2: HalfAdder -- The label name must be same as smaller part
51 port map(
52     A=>C,      -- the values are assigned according to TT
53     B=>sum1,
54     SUM=>sum2,
55     CARRY=>carry2
56 );
57
58 SUM<=sum2;      --final SUM out will be sum2
59 CARRY<=carry1 or carry2; --final CARRY out will be carry1 and carry 2
60
61 end Behavioral;
10 library IEEE;
11 use IEEE.STD_LOGIC_1164.ALL;
12
13 entity FullAdder is
14 Port
15 (
16     A:    in std_logic;  -- input 1
17     B:    in std_logic;  -- input 2
18     C:    in std_logic;  -- input 3
19     SUM:  out std_logic; -- output
20     CARRY: out std_logic -- output
21 );
22 end FullAdder;
23
24 architecture Behavioral of FullAdder is
25
26     signal sum1, sum2, carry1,carry2: std_logic; -- connect sum and carry with two half adders
27
28     component HalfAdder is -- Looks same as entity Full Adder
29     -- Half adder is defined with exact same manner as previous program
30     Port
31     (

```

Fig. B. Programming codes of the application.