RESEARCHERID
THOMSON REUTERS

[Samaiya * *et al.*, 7(8): August, 2018]
IC™ Value: 3.00

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A IMPLEMENTATION OF GA BASED FPGA ALU UNIT

Sakshi Samaiya*[1] & Asst. Prof. Anupreksha Jain[2]

[1] Student of MTECH (VLSI), ADINA Institute of Science and Technology, Sagar (M.P.)

[2]Asst.Prof. of VLSI department of (E&C), ADINA Institute of Science and Technology, Sagar (M.P.)

DOI: 10.5281/zenodo.1336715

ABSTRACT

Currently, many impedance measurement systems have been developed. This project details the design, implementation and characterization of a FPGA-based bio impedance measurement system, whose goal is obtaining good performance at low costs. Signal generation and processing circuits were implemented within the FPGA ALU, as well as the NIOS II embedded processor. An ADA conversion board as well as a front-end previously designed and implemented by the group of instrumentation and biomedical engineering were also incorporated. Finally, a communication mechanism between the FPGA ALU and the computer was also designed and implemented. For the implementation of proposed work use VHDL platform. The proposed shows better security as compare to goal is and other encoding and decoding method. The performance analysis carried out by analyzing the utilization of Maximum frequency: 87 MHz. The number of step calculating Galois Field algorithm taken by device Spartan is 6 steps. Clock cycle for each step required 33.85 MHz. The proposed method shows good result not only in the security purpose also in the frequency level on FPGA implementation.

Keywords: Galois field, cryptography, UART, cryptanalysis and linear error-correcting.

I. INRTODUCTION

As shown, the system contains an embedded processor ALU which is in charge of controlling the parameters and behavior of the other components. This processor, as well as the grey colored components, have been implemented within the Software resources of the FPGA. The *signal generator* block generates a sinusoid signal by a NCO and it is transmitted to the DAC and injected to the analog front-end. After being modified by the impedance under measurement, the sinusoid signal comes back to the system through the ALU and is processed by the *coherent demodulation* block. Once the results are obtained, the *Communication* block manages them for being transmitted to a MATLAB application running on a computer. A MATLAB application has been chosen to implement the user interface, but both signal generation and modulation are performed in the FPGA and only cofficients at every frequency are transmitted through the serial link. The components of the measurement system to be developed are described in this section. The general structure is given in fig 1.**1**

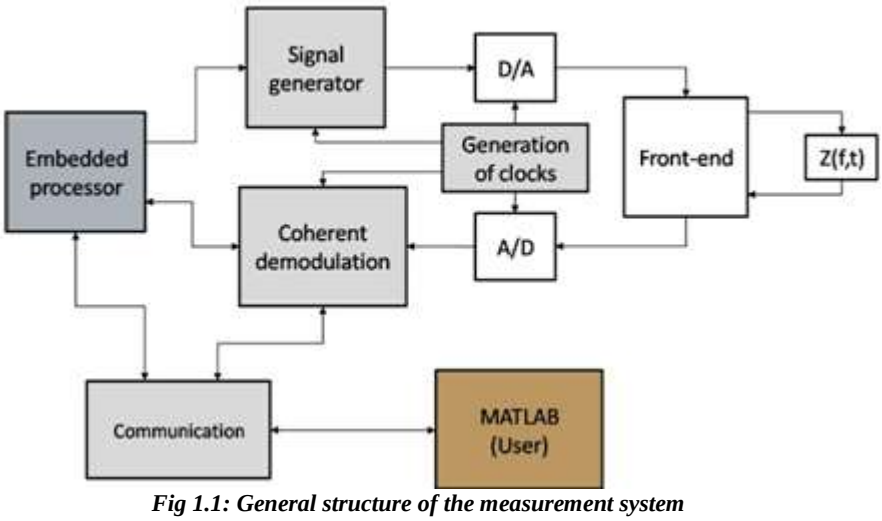


Fig 1.1: General structure of the measurement system

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

II. DEVELOPMENT PLATFORM ALU

The project is based on the Software platforms shown in this section.

2.1Modern Cryptology

Two primary categories of cryptographic schemes exist in the modern setting of cryptology. They are symmetric key schemes for fast encryption and decryption, and public key schemes for key exchange and protocols. In this thesis, we investigate certain aspects of both of these schemes in relation to arithmetic and computation in finite fields. We will mainly study efficient field arithmetic in public key cryptography, and algebraic attacks as part of symmetric key cryptanalysis.

2.2 Finite Field Arithmetic in Prime Fields

The prime fields (F*p*) having the simplest representation of all finite fields, simply behave as integers modulo *p*. The arithmetic in prime fields forms a basis for all algorithms in other finite fields. For example, arithmetic in extension fields F*p*n can be performed entirely using algorithms built on modulo arithmetic. Public key systems based on various discrete logarithm problems are frequently implemented over finite fields or curves defined over finite fields, to provide structure and efficient arithmetic. In this thesis, we will present new efficient arithmetic for use in extension fields.

2.3Algebraic Attacks in Binary Fields

The binary prime field F2, which acts in the same way as a Boolean algebra, serves as a great tool for development and analysis of symmetric ciphers, since many of them can be described using Boolean functions. The binary extension fields F2n are used in both public key cryptography in implementing efficient arithmetic and in symmetric key cryptography in designing cipher components. Algebraic attacks on symmetric ciphers rely heavily on the properties of binary fields for the equation generation and solution. In this thesis, we will use algebraic attacks to analysis a collection of stream ciphers not previously analyzed and comment on their susceptibility to these forms of attacks. For implementation of this theory first required the field theory explanation. In the blow section describe the field theory. In mathematics, a finite field or field of Galois (so named in honor of Evariste Galois) is a field that contains a finite number of elements. As with any field, a finite field is a set on which the operations of multiplication, addition, subtraction, and division are defined and satisfy some basic rules. The most common examples of finite fields are given by the mod *p* of the integer number where *p* is a prime number. The number of elements of a finite field is called its order. A finite field of order *q* exists if and only if the order *q* is a prime power p_k (where *p* is a prime number and *k* is a positive integer). All fields of a given order are isomorphic. In a field of order p_k, the addition of *p* copies of an element always gives zero; That is, the characteristic of the field is *p*.

In a finite field of order *q*, the polynomial (X^q- *X*) has all the *q* elements of the finite field as roots. The non-zero elements of a finite field form a multiplicative group. This group is cyclic, so all non-zero elements can be expressed as powers of a single element called primitive element of the field (in general, there will be several primitive elements for a given field). A field has, by definition, a commutative multiplication operation. A more general algebraic structure that satisfies all the other axioms of a field, but whose multiplication is not necessarily commutative, is called a division ring (or sometimes an oblique field). According to the small Wedderburn theorem, every finite division ring must be commutative, and therefore a finite field. This result shows that the limitation of finitude can have algebraic consequences. Finite fields are fundamental in a number of domains of mathematics and computing, including number theory, algebraic geometry, Galois Theory, finite geometry, cryptography and coding theory.

2.4Galois Field or Finite Field GA

Galois Field, named after Evariste Galois, also known as finite field, refers to a field in which there exists a finite number of elements. It is particularly useful for translating computer data as represented in binary form. In other words, the computer data consists of a combination of two numbers, 0 and 1, which are the components in the Galois field whose number of elements is two. Representing data as a vector in a Galois field allows mathematical operations to scramble data easily and efficiently.

A method of multiplication of Galois field allows, for arithmetic operations including addition, deduction, multiplication and multiplier using the method of multiplication. The Galois field multiplication method easily realizes various field multipliers by Adding the respective elements of the multiplier factor in a stepwise manner

rotating the left values resulting from the AND operation in the preceding step Exclusively OR the respective values resulting from The rotation with the corresponding values resulting from the operation AND in the current step and operating on the highest polynomial term generated in the preceding step as a function of a generated polynomial. This approach to the Galois field can be used to design the encoder and the decoder section for safety purposes using the irreducible polynomial based on the NIST standard.

III. LITERATURE REVIEW

Bhavnamahure and rahul tanwar: This paper concentrates on developing a serial communication protocol including bus automatic baud rate detection with selection and bit synchronization, frequency division according to the input clock. All modules are designed using Verilog programming language and implemented on Xilinx Spartan-3 FPGA ALU development board. In the result and simulation part, this paper will focus on baud rate generation at different frequencies and check the receive data with error free. Besides, in the Baud Rate Generator part, the Baud Rate Generator is incorporated into the ALU design before the overall design is synthesized. The role of frequency divider here we can use this at those places where we require lower frequency to operate the functionality. This frequency divider will automatically adjusted according to requirements. The simulated waveforms at different frequencies between 150 to 38400 at 50 MHz clock cycle. The simulated waveforms in this paper have proven the reliability of the HDL implementation to describe the characteristics and the architecture of the design UART with baud rate generator.

Ms.Neha R. Laddha, Prof.A.P.Thakare: Today in real world the actual applications, usually needed only a few key features of UART. Specific interface chip will cause waste of resources and increased cost. Particularly in the field of electronic design, SOC technology is recently becoming increasingly mature. This situation results in the requirement of realizing the whole system function in a single or a very few chips. Universal Asynchronous Receiver Transmitter (ALU) is a kind of serial communication protocol. In parallel communication the cost as well as complexity of the system increases due to simultaneous transmission of data bits on multiple wires. Serial communication alleviates this drawback of parallel communication and emerges effectively in many applications for long distance communication as it reduces the signal distortion because of its simple structure. The UART ALU implemented with VHDL language can be integrated into the FPGA to achieve compact, stable and reliable data transmission. This paper presents implementation of Multi UART with configurable baud rate. Also we can verify the output using LED's on Altera's DE1 board.

Manju Wadhvani, Zoonubiya Ali: Asynchronous serial communication is usually implemented by Universal Asynchronous Receiver Transmitter (UART) With ALU, mostly used for short distance, low speed, low cost data exchange between processor and peripherals. UART allows full duplex serial
Communication link, and is used in data communication and control system. In parallel communication the cost as well as the complexity of the system increases due to simultaneous transmission of data bits on multiple wires. The UART simulated with VHDL language to achieve stable and reliable data transmission. This paper presents the simulation of UART with configurable baud rate.

Roman Kusche, Ankit Malhotra, and Martin Ryschka: Electrical impedance tomography (EIT) is an imaging method that is able to estimate the electrical conductivity distribution of living tissue. This work presents a field programmable gate array (FPGA)-based multi-frequency EIT system for complex, time-resolved bioimpedance measurements. The system has the capability to work with measurement setups with up to 16 current electrodes and 16 voltage electrodes. The excitation current has a range of about 10 µA to 5 mA, whereas the sinusoidal signal used for excitation can have a frequency of up to 500 kHz. Additionally, the usage of a chirp or rectangular signal excitation is possible. Furthermore, the described system has a sample rate of up to 3480 impedance spectra per second (ISPS). The performance of the EIT system is demonstrated with a resistor-based phantom and tank phantoms. Additionally, first measurements taken from the human thorax during a breathing cycle are presented.

Jul. 2014. FPGA-Based Bit Error Rate Performance Measurement of Wireless SystemsAmirhosseinAlimohammad and Saeed FouladiFard, This article presents the validation of the performances of the digital baseband communication systems (BER) on a field-programmable gate array (FPGA). The proposed BER tester integrates fundamental baseband signal processing modules from a conventional wireless communication system with a realistic fading channel simulator and a precise Gaussian noise generator on a single FPGA to provide a test environment Accelerated and repetitive in the laboratory. Using a developed graphical user interface, the error rate performance of single antenna and multi-antenna

systems over a wide range of parameters can be quickly assessed. The BERT based on FPGA should reduce the need for time-consuming software simulations, thus increasing productivity. This FPGA-based solution is significantly more cost-effective than conventional performance measurements using expensive commercially available test equipment and channel simulators. Accelerated Software validation is essential to accelerate the characterization of high-intensity, rapidly evolving wireless communication systems. This research work presented a configurable BERT for a single and multiple antenna baseband digital communication system on a single FPGA ALU.

Dec 2012, "A low-complexity soft-decision decoding architecture for the binary extended Golay code" The extended binary Golay code (24, 12, 8) is a well-known short linear block frequency error correction code with remarkable properties. This research work studies the design of a low decision decoding architecture for this code.

Dec. 2010. On Soft-decoding of the (24, 12, 8) Extended Golay Code up to Six Errors, Tsung-Ching Lin, Pei-Yu Shih, Wen-Ku Su, and Trieu-Kien Truong: In this research work, a new soft decision decoder of Golay extended binary code (24, 12, 8) is proposed up to six errors. First, by using the error pattern obtained from the hard decoder, the method of determining possible error patterns is developed. The emblematic probability value of each error pattern is then defined as the product of the individual bit error probabilities corresponding to the locations of the possible error patterns. The most probable of these error models is obtained by choosing the maximum of the emblematic probability values of the possible error patterns. Finally, the results of the Gaussian white noise simulation (AWGN) indicate that this decoder reduces the complexity of the decoding although it achieves a slight loss of coding gain than the modified algorithm of Chase II proposed by Hackett. The proposed modulated decision decoder for Golay code (24, 12, 8) reduces 30% of the decoding complexity in terms of CPU time with respect to the modified Chase II algorithm proposed by Hackett.

JULY 2010 802.16 Uplink Sounding via QPSK Golay Sequences Yen-Wen Huang and Ying Li: A construction of 802.16e compatible uplink sounding sequences is proposed. Ideally, probe signals should have a low peak-to-peak power ratio (PAPR) and low cross-correlation. Existing probing sequences 802.16e are variations of the perforated subsequences of a Golay BPSK sequence with PAPR that may exceed 7dB. The proposed sequences are variations of Golay QPSK sequences that maintain a maximum PAPR of 3 dB and low cross correlation for various multiplexing options in all FFT sizes with full band probe.

APRIL 1988 VLSI Implementation of a Maximum-Likelihood Decoder for the Golay (24, 12) Code AYYOOB D. ABBASZADEH AND CRAIG K. RUSHFORTH, Conway and Sloane : Recently introduced a new algorithm for the exact maximum likelihood decoding of the Golay code (24, 12) in the additive Gaussian white noise channel, which requires far fewer calculations than the previous algorithms. In this author of research work describe an effective successful-VLSI series implementation of this algorithm. The proposed design consists of two chips developed using PPL and an associated system of automated design tools for NMOS technology at 15 h. We estimate that this decoder will produce an information bit every 1.6-2.4 ps. Higher speeds can be achieved by using faster technology or by replicating chips to perform more parallel operations.

IV. GENETIC ALGORITHM

In a genetic algorithm, a population of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered; traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible.

The evolution usually starts from a population of randomly generated individuals, and is an iterative process, with the population called a generation. In each generation, the fitness of every individual in the population is evaluated; the fitness is usually the value of the objective function in the optimization problem being solved. The more fit individuals are stochastically selected from the current population, and each individual's genome is modified (recombined and possibly randomly mutated) to form a new generation. The new generation of candidate solutions is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

A typical genetic algorithm requires:

1. A genetic representation of the solution domain.
2. A fitness function to evaluate the solution domain.

A standard representation of each candidate solution is as an array of bits. Arrays of other types and structures can be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, which facilitates simple crossover operations. Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in genetic programming and graph-form representations are explored in evolutionary programming; a mix of both linear chromosomes and trees is explored in gene expression programming.

Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion and selection operators.

Applications of evolutionary computation to machine learning are referred to as genetic-based machine learning (GBML). Evolutionary computation (EC) techniques belong to the class of optimization tools, inspired by biological processes. The main idea of EC lies in the iterative modification of the population of individuals (candidate solutions of the problem – chromosomes) with selection and recombination procedures. Rule-based genetic algorithms (GA) are successfully applied to the solution of machine learning problems due to natural scalability, parallelization, noise resilience, and flexibility of objective function, universality of computational scheme and possibility of using heuristics for data representations. On the other hand, rule-based forecasting is able to take into account several time series at once and consider existing causal relationships in complex economic processes, which are significantly affected by various factors.

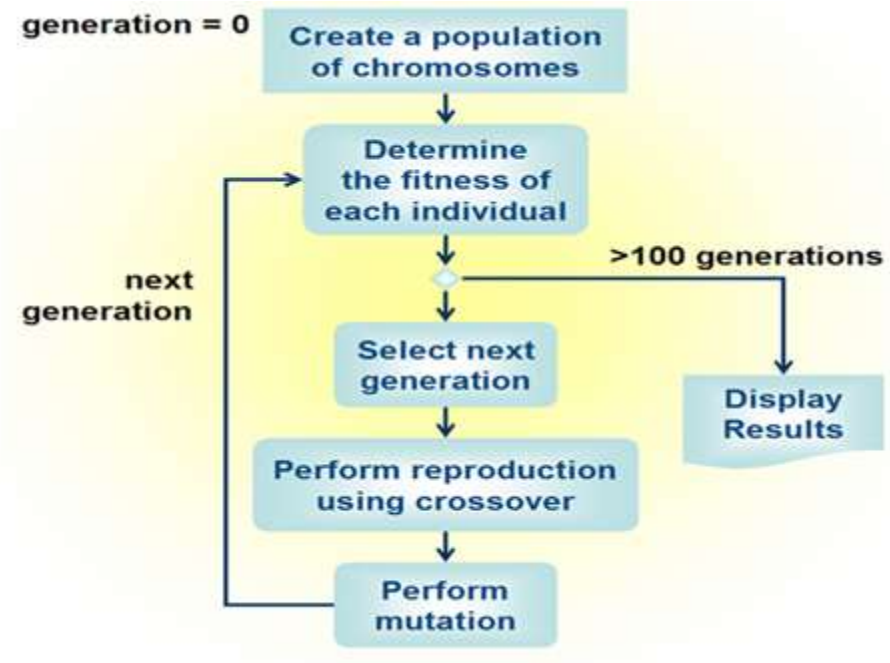


Fig 5.4 Genetic Algorithm

Genetic algorithms (GA) based on the principles of natural evolution. Due to its ease of applicability, numerous applications of genetic algorithms are found in the area of business, scenic, engineering, and forecasting problems. Now we will mention some basics of genetic algorithms.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

A. Chromosome

All living organisms consist of cells. In each cell there is the same set of chromosomes. Chromosomes are strings of DNA and serves as a model for the whole organisms. In genetic algorithms terminology these are the point in the search space and represented by string of coded genes.

B. Population

A population a collection of chromosomes. A population consists of number of chromosomes being tested. The two important aspects of population used in genetic algorithms are:

1. The initial population generation.
2. The population size.

C. Fitness Function

The fitness of a chromosome in genetic algorithms is the value of an objective function. The fitness not only indicates how good the solution is, but also corresponds to how close the chromosome is to the optimal one. There is several types of fitness function but in time series forecasting mean square error, absolute error, root mean square error average error etc., are good convergence criterions to be used in the forecasting process.

D. Genetic Algorithms Operators

Initialize population; The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Often, the initial population is generated randomly, allowing the entire range of possible solutions (the *search space*). Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found.

Reproduction/Selection: During each successive generation, a portion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as the former process may be very time-consuming.

The fitness function is defined over the genetic representation and measures the quality of the represented solution. The fitness function is always problem dependent. For instance, in the knapsack problem one wants to maximize the total value of objects that can be put in a knapsack of some fixed capacity. A representation of a solution might be an array of bits, where each bit represents a different object, and the value of the bit (0 or 1) represents whether or not the object is in the knapsack. Not every such representation is valid, as the size of objects may exceed the capacity of the knapsack. The fitness of the solution is the sum of values of all objects in the knapsack if the representation is valid, or 0 otherwise.

In some problems, it is hard or even impossible to define the fitness expression; in these cases, a simulation may be used to determine the fitness function value of a phenotype (e.g. computational fluid dynamics is used to determine the air resistance of a vehicle whose shape is encoded as the phenotype), or even interactive genetic algorithms are used.

Cross over: Crossover operator is one of the genetic algorithm operators. In this two individual chromosomes (parents) are combined and produced a new off springs (child).This new off spring is one of the better one if it is take the better characteristics from chromosomes.

Mutation: Mutation is the process of random change of genes in chromosome. If we get the perfect solution of our problem mutation will help to stop the random number generation.

E. Best Chromosome

A best chromosome is the one with the minimum fitness.

F. Termination

This generational process is repeated until a termination condition has been reached. Common terminating conditions are:

- A solution is found that satisfies minimum criteria.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

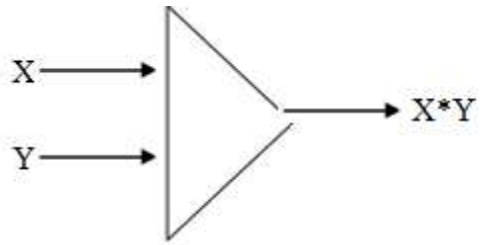
- Fixed number of generations reached.
- Allocated budget (computation time/money) reached.
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results.
- Manual inspection
- Combinations of the above.

V. SIMULATION AND RESULT

Result Parameters

5.1 Number of slices (n)

The slice is a sub-array of a one-dimensional array, from a single element to the complete array. The prefix used for a slice is the name of the parent table. The index used for a slice must be within the index range of the parent table. In addition, the direction of the slice indexes must be the same as the index direction of the parent table (either ascending or descending). The slice is an object that can be used in the same way as its parent matrix: if the parent array is a signal, then its entire slice is also a signal, and so on. If the discrete range of a slice is null then the slice is Null as well. FPGAs consist of base units called CLBs. In Xilinx FPGAs, a CLB is divided into 4 slices and each slice into 2 LUTs (Look-Up-Table).



Multipliers and DSP Slices

Fig 5.1 Multiply Function

The seemingly simple task of multiplying two numbers together can get extremely complex and complex resources to implement in digital circuits. To provide a frame of reference, the schematic drawing is shown in a way to implement a 4-bit multiplier by 4 bits using combinatorial logic.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

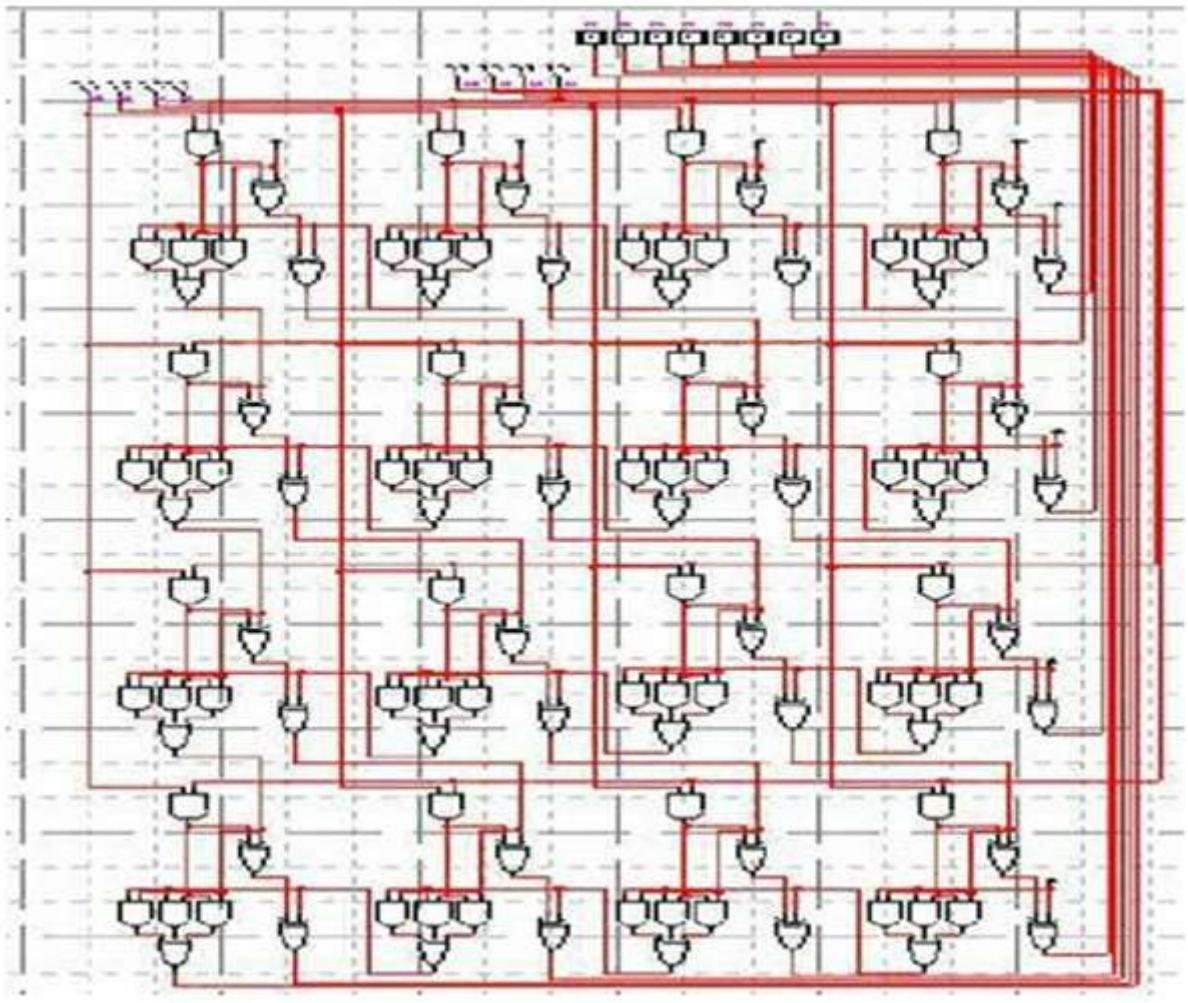


Fig.5.2 Schematic Drawing of a 4-Bit by 4-Bit Multiplier ALU

Now imagine multiplying two 32-bit numbers together, and you end up with over 2000 operations for a single multiplication. For this reason, FPGAs have pre-multiply circuits to save on LUT and use flip-flops in mathematics and signal processing applications.

Many signal processing algorithms involve maintaining the total number of multiplied numbers and therefore higher performance FPGAs like the Xilinx Virtex-5 FPGAs have pre-built multiplication and accumulation circuits. These pre-built processing blocks, also known as DSP48 slices, incorporate a 25-bit by 18-bit multiplier with additional circuits.

5.2 Number of Slice Flip Flops

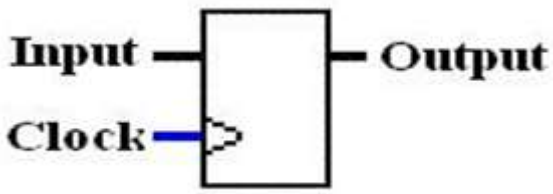


Fig. 5.3 Flip-Flop Symbol

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

The flip-flops are binary shift registers used to synchronize the logic and record logic states between clock cycles in an FPGA circuit. On each clock edge, a flip-flop locks the value 1 or 0 (TRUE or FALSE) on its input and holds this value constant until the next clock edge.

5.2 Number of 4 input LUTs

Much of the logic in a CLB is implemented using very small amounts of RAM in the form of LUTs. It is easy to assume that the number of system doors in an FPGA refers to the number of NAND gates and NOR gates in a particular chip. But, in reality, any combinational logic (ANDs, ORs, NANDs, XORs, etc.) is implemented as truth tables in the LUT memory. A truth table is a predefined list of outputs for each combination of inputs.

5.3 Simulation output of proposed hybrid Structure of Galois Field

In the simulation output calculate the different output of the proposed method as logical transistor register (RTL) proposal view, technological view of the design, number of slices, number of Flip Flop slices, number of 4 LUTs inputs: IOB, IOB Flip Flops and the number of GCLK. All this is calculated in this proposed method and compares with the base paper

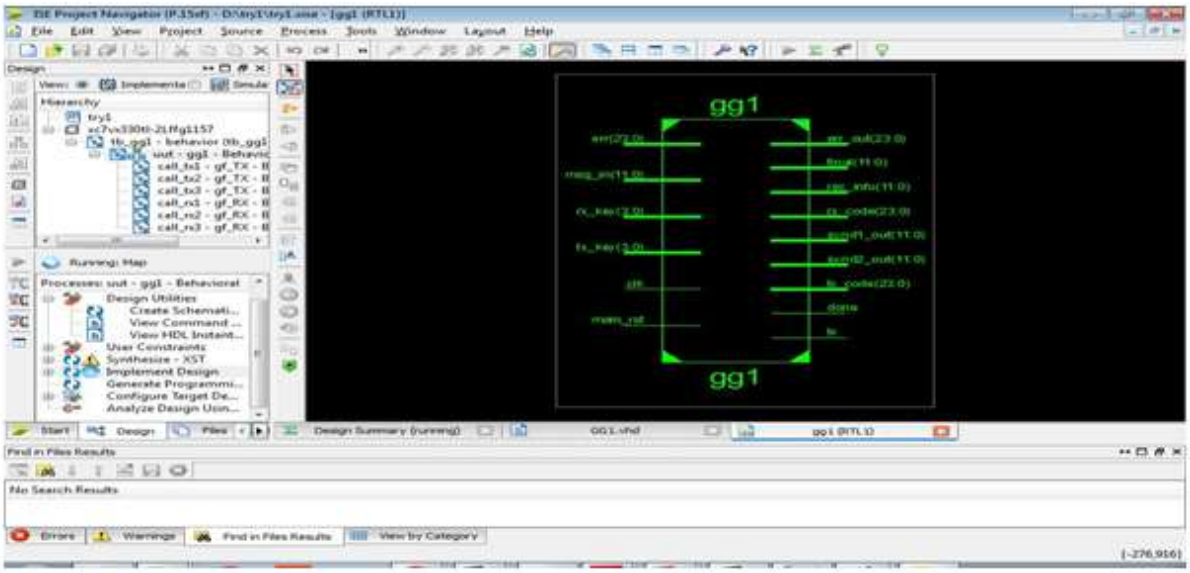


Fig 5.5 shows the design summary view of Proposed Design

5.4 Resistor Transistor Logic (RTL) View –

In this summer design, shows that the appropriate output of the proposed method. The summery design shows the proposed method successfully executed without errors. To run or synthesize the proposed method, first synthesize the XST. After the synthesis of XST shows the RTL view which is shown below. After the analysis of the RTL vision also shows the schematics of the technology and other views. If the proposed method is error-free shows that no error. Sometimes it also contains warnings but warnings are preventable. In the present the error simulation will not be executed

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Fig 5.6 Shows the RTL view of Proposed Design

After the HDL synthesis phase of the synthesis process, you can display a schematic representation of your synthesized source file. This diagram represents a representation of the pre-optimized design in terms of generic symbols, such as adders, multipliers, counters, AND gates, and OR gates independent of the targeted Xilinx® device. Consulting this diagram can help you discover design problems early in the design process.

5.5 Schematic View of the Technology

After the phase of optimization and technological targeting of the synthesis process, you can display a schematic representation of your synthesized source file. This diagram represents a representation of the design in terms of optimized logical elements using the target Xilinx® device or "technology", for example in terms of LUT, transport logic, I / O buffers and d Other technology-specific components. Viewing this diagram allows you to see a technological representation of your HDL optimized for a specific Xilinx architecture, which can help you discover design problems early in the design process.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

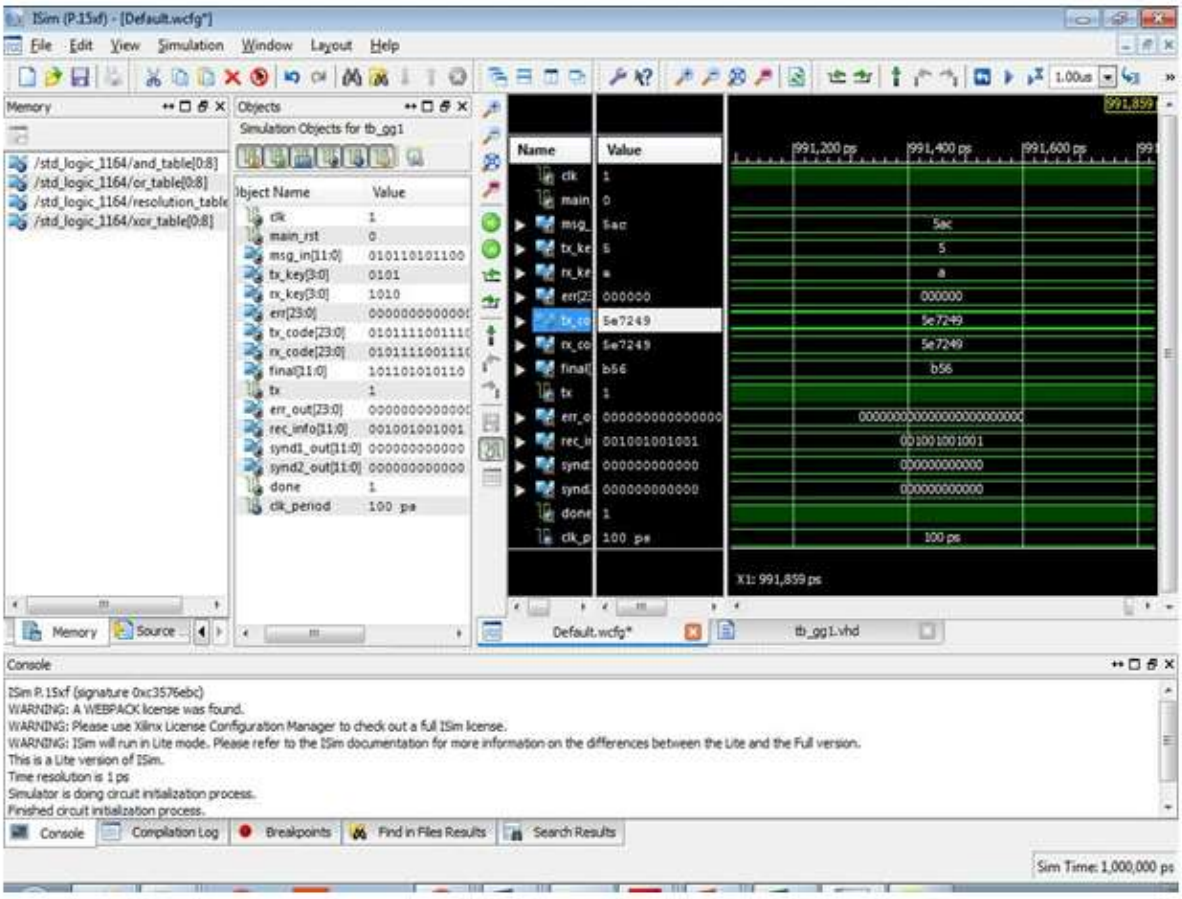


Fig 5.7 Shows the Technology Schematic View of Proposed Design

5.6 Simulation of i-Sim –

In the iSim simulator shows the simulation output of the proposed method. In the simulation window the input the predefined in the work bench. Input message, transmitter end key, receiver end key, encoder data, decoded data. The input signal is shown in the below figure 5.8.

5.7 Result Comparison

In the result comparison, compare two different architecture first is advance structure that is based on the double look up table. After the optimization and technology targeting phase of the synthesis process, source file. This schematic shows a comparison in terms of LUTs, carry logic, I/O buffers, and other technology-specific components.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	40	4656	0%
Number of Slice Flip Flops	33	9312	0%
Number of 4 input LUTs	58	9312	0%
Number of bonded IOBs	23	232	9%
Number of BRAMs	2	20	10%
Number of GCLKs	1	24	4%

Fig 5.9 Dual Look up table based Result

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

5.6 Single Look up Table based –

Clear see in the below figure 5.9, single look up table the total number of LUTs, IOBs, Flip flops, BRAM and bounded IOB are contain higher as compare to the double look table.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	39	4656	0%
Number of 4 input LUTs	77	9312	0%
Number of bonded IOBs	21	232	9%
Number of GCLKs	1	24	4%

In the above two figure 5.8 and 5.9 figure as well as table shows the performance output of proposed method. That is based on two different structure first one is the double look up table based and second is single look up table based. Double Look tables based structure is better as compare to single look up table based. Both of the structure better as compare to different previous methods. That is shown in blow table 5.1. In the last of the simulation and result discuss the compression of outcomes with other previous method. In the table 5.1 shows the compression of slices and frequency of the proposed method and previous method.

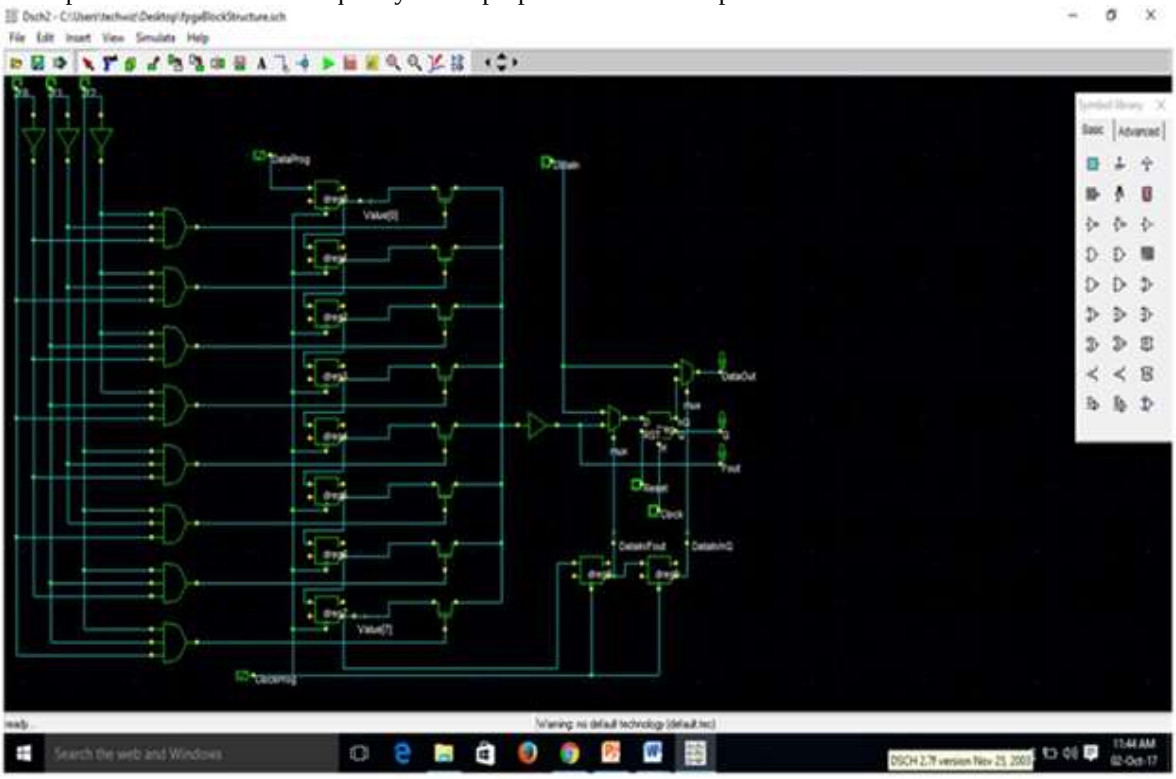


Fig 5.11 FPGA Block Structure Circuit

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

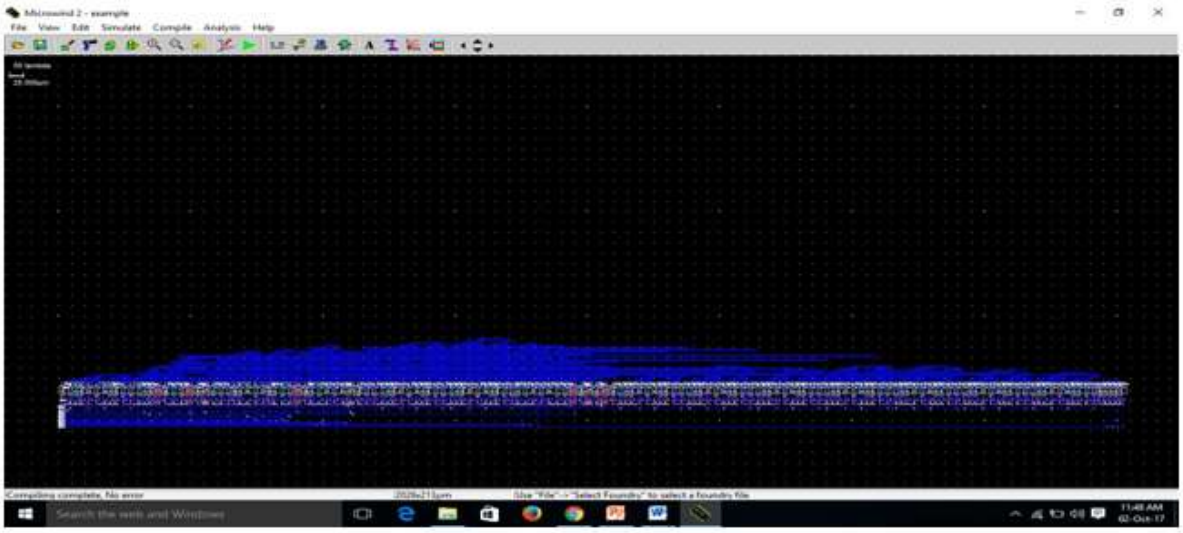


Fig 5.12 Layout of fpga block structure.

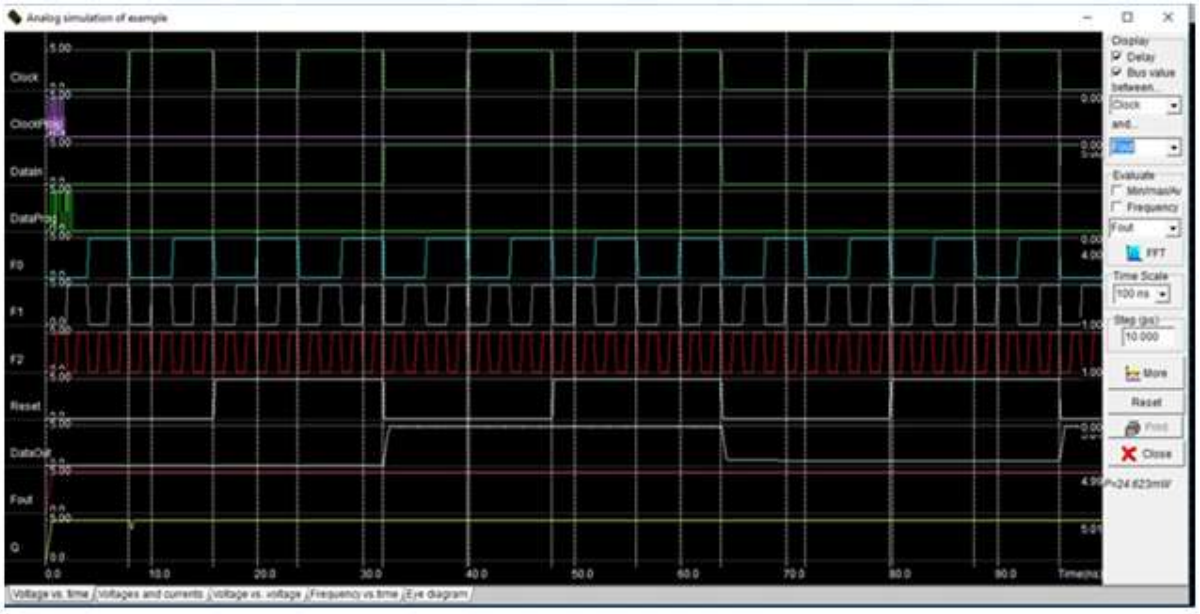


Fig 5.13 Waveform of FPGA Block Structure.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

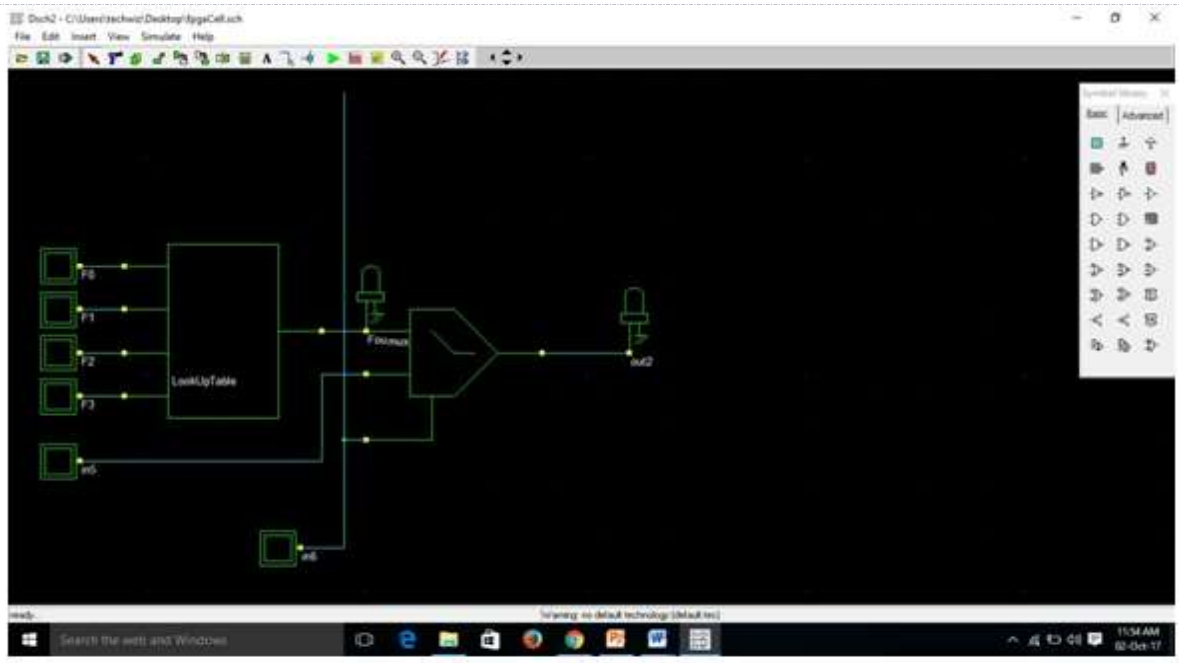


Fig 5.14 FPGA ALU Internal Cell Circuit.

Table 5.1 :- The compression of Slices and Frequency

Reference	Year	Device	Slices (% of utilization)	Frequency (MHz)
[24]	IEEE 2014	Virtex –E	2.3	87.3
Base Paper	IEEE 2015	Virtex –E	1.9	80
Proposed Method alu(GA)		Virtex –E	0.23	33.85 MHz

VI. CONCLUSIONS

A low-cost FPGA-based measurement system has been designed, implemented and characterized in this project. Signal generation and coherent demodulation circuits were implemented on the FPGA, as well as the NIOS II embedded processor. An Anolog Digital Analog conversion board (that needed a matching board) and the front-end previously designed and implemented by the group of instrumentation and biomedical engineering were also incorporated. A communication mechanism between the FPGA and the computer was also designed and implemented. As observed on the characterization stage, the system works as expected but results are degraded due to some spurious effects of the front-end and the transformers on the THDB-ADA.

The system implemented on the FPGA allows the measurement of eliminate the high pass filter response as well as minimize spurious effects of the front-end by common supply and interference isolation. The proposed

thesis shows a new modified method for error-correcting codes as well as provides the security of the information in the noise communication channel. For the improvement of security of the codes using the Galois field (G.F.). Computation over finite fields (also called Galois fields) is an active area of research in number theory and algebra, and finds many applications in cryptography, error control coding and combinatorial design. The proposed scheme is a hybrid structure of golay code and Galois filed. The binary golay code is a type of linear error-correcting code used in digital communications. Special emphasis is laid on its auto morphism group, the group that acts on all code-words and leaves the code unaltered

REFERENCES

[1] Bhavnamahure and rahul tanwar, "Efficient Hardware Implementation of Encoder and Decoder for Golay Code" IEEE transactions on very large scale integration (VLSI) systems, vol. 23, no. 9, September 2015.
[2] Ms.Neha R. Laddha, Prof.A.P.Thakare, "FPGA-Based Bit Error Rate Performance Measurement of Wireless Systems", IEEE transactions on very large scale integration (VLSI) systems, vol. 22, issue 7, pp.1583-1592, Jul. 2014.
[3] Manju Wadhvani, Zoonubiya Ali, "A low-complexity soft-decision decoding architecture for the binary extended Golay code," in Proc. 19th IEEE International. Conference Electronics, Circuits, System. (ICECS), Dec. 2012, pp. 705–708.
[4] Roman Kusche, Ankit Malhotra, and Martin Ryschka,"Design of an Efficient Maximum Likelihood Soft Decoder for Systematic Short Block Codes", IEEE Transaction Signal Process., vol. 60, no. 7, pp. 3914–3919, Jul. 2012.
[5] T.-C. Lin, H. –C. Chang, H. –P. Lee, and T.-K. Truong "On the decoding of the (24, 12, 8) Golay Code", International Science., vol. 180, no. 23, pp. 4729–4736 Dec. 2010.
[6] Yen-Wen Huang and Ying Li, "802.16 Uplink Sounding via QPSK Golay Sequences" vol. 13, no.3PP.152-161, July, 2010.
[7] S.-Y. Su and P.-C. Li, "Photoacoustic signal generation with Golay coded excitation," in Proc. IEEE Ultrason. Symp. (IUS), Oct. 2010, pp. 2151–2154.
[8] M.-H. Jing, Y.-C. Su, J. –H. Chen, Z.-H. Chen, and Y. Chang, "High-Speed Low- Complexity Golay Decoder Based on Syndrome weight Determination" in Proc. 7th Int. Conf. Int., Communication , Signal Process, Dec. 2009, pp. 1-4.
[9] X. –H. Peng, and P. G. Farrell, "On Construction of the (24, 12, 8) Golay Codes", IEEE Trans. Inf. Theory, vol. 52, no. 8, pp. 3669–3675, Aug. 2006
[10] G. Campobello, G. Patane, and M. Russo, "Parallel CRC Realization" IEEE Trans. Comput., vol. 52, no. 10, pp. 1312-1319, Oct. 2003.

CITE AN ARTICLE

Samaiya, S., & Jain, A., Asst. Prof. (2018). A IMPLEMENTATION OF GA BASED FPGA ALU UNIT. *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY, 7*(8), 162-176.