iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: VII Month of publication: July 2023

DOI: https://doi.org/10.22214/ijraset.2023.54671

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

Significance of Mathematics in VLSI Circuit Design

Dr. K. L. Sudha,
Prof, Dept of ECE, DSCE, Bengaluru, INDIA

Abstract: VLSI (Very-large-scale integration) circuit design requires a solid foundation in mathematics, particularly in linear algebra, calculus, and graph theory. For VLSI design, knowledge of matrix algebra and vector calculus is also important in analyzing and optimizing the performance of integrated circuits. Familiarity with algorithms and data structures is helpful for designing efficient digital logic circuits and for computer-aided design (CAD) tools. Knowledge of circuit analysis, Boolean algebra, and abstract algebra would be beneficial as well. Vedic mathematics helps to design few mathematical operations on FPGA with simple circuits. This paper brings out importance of various fields of mathematics in designing circuits for VLSI by considering examples.

Index: VLSI, FPGA, Boolean algebra, Vedic maths

I. INTRODUCTION

This paper is envisioned at enlightening the reader with the importance of mathematics in VLSI circuit design. A very deep mathematics built into the circuit, would bring automation in the design process. Mathematics becomes more important as we go to high complexity designs involving millions of transistors, high frequencies and systems-on-chip. Some mathematical topics which are relevant from the VLSI perspective are the uses of graphs and hyper graphs for modeling contemporary life problems mainly in terms of connection, flow of information, and hierarchical structure. Use of linear algebra in VLSI and in particular the use of eigenvalues in various contexts such as system reduction and partitioning is also seen. Sub modular functions, differential equations, the symbolic methods used in modeling behavior for analysis and verification and the ubiquitous `decomposition theme' are also important. Some areas in VLSI design automation involve use of sub modular functions for optimization problems in VLSI, partitioning large scale systems which arise in many forms throughout this area, the use of differential equations in modeling and simulation, logical and algebraic techniques in verification and applications of classical automaton decomposition theory. The Vedic mathematics is a type of mathematics which is derived from the ancient literature Veda of India, which provides one line and superfast solution to any problem along with quick cross checking mechanism. All elementary operations, the addition, subtraction, multiplication and division for any processor can be performed efficiently by Vedic mathematics. At the end of the paper, conclusion is drawn on how mathematics is important, relevant and critical for developing circuits for VLSI design.

II. LINEAR ALGEBRA

A branch of mathematics that deals with the manipulation and analysis of linear equations and matrices. It plays an important role in the field of Very-Large-Scale Integration in various ways. Circuit analysis, logic synthesis, placement and routing in VLSI circuit design are based on linear algebra. Boolean algebra plays important role in designing any digital circuit. Different digital circuits are used in FPGA implementation of any larger circuit. Paper [2] shows how a cryptographic system can be implemented on FPGA using Boolean algebra.

A. *Circuit Analysis*

Linear algebra is used to solve systems of equations that model the behavior of circuits. This can be used to analyze the performance of a circuit and to optimize its design. One of the examples of using linear algebra in circuit analysis is circuit matrix analysis. This method is used to analyze the behavior of electric circuits represented by a system of linear equations. The system of equations can be represented in matrix form, where the elements of the matrix represent the resistances, capacitances and inductances in the circuit and the elements of the vector represent the voltage and current sources. By using matrix inversion, the solution of the circuit can be obtained by solving the matrix equation AX = B, where X represents the unknown voltages and currents in the circuit, A is the circuit matrix, and B is the vector of known sources.[3] Gaussian elimination or LU decomposition can be used to solve the system of equations and obtain the voltages and currents in the circuit.

Linear algebra also plays an important role in the optimization of the circuit design, for example, the placement and routing of the components on the chip. Linear programming can be used to minimize the total wire length or the total power consumption in the circuit while satisfying a set of constraints.

In all these examples, linear algebra is used to model the electrical behavior of the circuit and solve the system of equations to obtain the voltages and currents in the circuit. It also plays an important role in the optimization of the circuit design.

B. Logic Synthesis

Linear algebra is used to generate Boolean expressions that can be used to synthesize digital circuits. The Quine-McCluskey algorithm is one example of this method which minimizes the Boolean function to implement in a circuit.

Use of Linear Algebraic Methods (LAMs) for logic minimization is another area where linear algebra is used. In logic minimization, the goal is to find the smallest possible Boolean function that implements a given set of logical constraints. This can be done by using LAMs to represent the Boolean function as a matrix and then using linear algebra techniques to simplify and minimize the matrix. For example, using Gaussian elimination, it is possible to simplify the matrix by combining rows and columns to eliminate redundant terms. This can lead to a smaller and more efficient logic circuit.

Linear algebra is used in high-level synthesis also [4]. In this process, a high-level behavioral description of a circuit is transformed into a gate-level implementation. Here the circuit is represented as a matrix, and then linear algebra techniques are used to optimize the matrix and find the most efficient gate-level implementation of the circuit.

Thus, linear algebra is used to model the behavior of the circuit and optimize the design by simplifying and minimizing the matrix representation of the circuit. This can lead to a smaller and more efficient logic circuit.

C. Placement and Routing

Linear algebra is used in the placement and routing of circuit components on an integrated circuit [5]. It is used to optimize the position and connectivity of components in order to minimize the overall area and power consumption of the circuit.

In placement, linear programming is used to minimize the total wire length or the total power consumption of the circuit while satisfying a set of constraints. Linear programming techniques such as the Simplex method can be used to find the optimal placement of the components on the chip by minimizing an objective function subject to a set of constraints.

Linear algebra is used in routing where graph-based techniques such as the shortest-path algorithms are used. In this case, the placement of the components on the chip is represented as a graph, where each component is a node and the connections between the components are edges. The shortest-path algorithm can then be used to find the shortest path between two nodes, which corresponds to the optimal routing of the wires between the components.

In both examples, linear algebra is used to model the placement and routing of the circuit on the chip and optimize the layout by minimizing an objective function subject to a set of constraints. This can lead to a more efficient and compact layout of the circuit on the chip.

D. Graph Theory

Linear algebra is useful for solving graph-based problems in VLSI. This includes solving problems involving graphs that represent circuits, as well as graphs that represent the underlying geometry of the integrated circuit.[6]

An example of using linear algebra to solve graph-based problems in VLSI is in the use of spectral graph theory for floor planning. Floor planning is the process of arranging the components of a chip on a two-dimensional plane to minimize the area and wire length of the circuit. Spectral graph theory can be used to represent the placement of the components as a graph, where each component is a node and the connections between the components are edges. The Laplacian matrix can then be used to study the properties of the graph and find the optimal placement of the components. The Eigen vectors of the Laplacian matrix can be used to find the coordinates of the components that minimize the total wire length of the circuit.

In this example, linear algebra is used to model the floor planning problem as a graph and optimize the layout by studying the properties of the graph and finding the optimal placement of the components. This can lead to a more efficient and compact layout of the circuit on the chip, reducing the total wire length and power consumption.

E. Signal Processing

Linear algebra is used in the processing of signals that travel through integrated circuits. This includes the manipulation of matrices and vectors representing the signal values, as well as the solving of systems of equations that model the behavior of signals.[9]

In VLSI design, linear algebra can be used in signal processing to optimize the design of analog circuits such as filters.One example of using linear algebra in signal processing is in the use of state-space representation for designing filters. State-space representation is a method for representing linear systems using a set of first-order differential equations. Linear algebra can be used to transform the equations into a matrix form, where the elements of the matrix represent the system's dynamics and the elements of the vector represent the system's inputs and outputs. By using linear algebra techniques such as matrix inversion, it is possible to design filters that meet specific performance criteria, such as cutoff frequency, stop band attenuation and stability.

Linear algebra is also used in the realization of filter structures. In this process, different filter structures such as Butterworth, Chebyshev, and elliptic can be represented as a linear system. Linear algebra techniques such as matrix factorization can be used to decompose the filter transfer function into a cascade of simpler filter sections, each with specific transfer function and coefficients.

Linear algebra is helpful to model the behavior of the signal processing circuits and optimize the design by studying the properties of the system and designing the filter that meets specific performance criteria.

As we can see, linear algebra provides powerful tools for solving problems in VLSI and is an important field to have a good understanding of when working in this area.

III. GRAPHS AND HYPERGRAPHS

These can be used to model various problems in VLSI (very-large-scale integration) design, such as circuit layout, routing, and testing. [7]

1) In circuit layout, a graph can be used to represent the placement of transistors and other components on a chip, with nodes representing the components and edges representing the connections between them. For example, in a simple circuit with a battery, a resistor, and a lamp, the vertices of the graph would represent the battery, the resistor, and the lamp, and the edges would represent the wires connecting them. Hypergraph can also be used to represent the same circuit, but with an additional layer of abstraction. In this case, the vertices of the hypergraph would represent the components of the circuit, and the hyperedges would represent the groups of components that are connected in a certain way. For example, in a simple circuit with a battery, a resistor, and a lamp, the vertices of the hypergraph would still represent the battery, the resistor, and the lamp, but the hyperedges would represent the groups of components that are connected in series or in parallel. Both graphs and hypergraphs can be used to represent the same circuit, but with different levels of abstraction and detail. The choice of which one to use depends on the specific problem you are trying to solve and the level of detail you need.
2) In routing, a graph can be used to represent the paths that signals must take through the chip, with nodes representing the components and edges representing the interconnects between them. For example, a graph can be used to represent the routing of wires on a chip as a collection of nodes and edges. The nodes of the graph represent the pins of the chip's components and the edges represent the wires that connect the pins. For example, in a simple circuit with a few gates, each gate can be represented as a node and the wires connecting the gates can be represented as edges. This can be used to find the shortest path between two pins (nodes) in terms of the number of edges that need to be traversed. A hypergraph can also be used to represent the routing of wires on a chip. In this case, the vertices represent the components and the hyper-edges represent the groups of pins that need to be connected. This can be useful for solving more complex routing problems, such as those involving multiple nets (groups of pins that need to be connected) that share a common resource, like a bus or a switch. For example, a hypergraph can be used to model a bus with multiple nets (hyper-edges) that share a common routing resource (vertices) and then use graph algorithms to find an optimal solution for routing the nets on the bus.
3) In testing, a hypergraph can be used to represent the relationships between different components and test patterns, with nodes representing the components and hyperedges representing the groups of components that are tested together. These mathematical structures allow modeling the problem efficiently and optimizing it. One way of using a graph for testing in VLSI is to represent the chip's design as a graph, with the nodes representing the gates and the edges representing the connections between the gates. The graph can then be traversed to test the functionality of the chip by simulating the flow of signals through the gates and checking that the output is as expected. This can be done by using graph algorithms such as depth-first search or breadth-first search to traverse the graph and simulate the logic of the circuit. Hypergraph can be used for testing in VLSI to represent the chip's design as a hypergraph, with the vertices representing the components and the hyperedges representing the groups of components that need to be tested together. This can be useful for testing complex chips with multiple interacting components, such as a microprocessor. In this case, a hypergraph can be used to model the interactions between the different components of the chip and then use graph algorithms to find an optimal sequence of tests that cover all the interactions. Graphs and hypergraphs provide a way to model the chip's design and test the functionality by traversing the graph or hypergraph and simulating the logic or interactions.

IV. SUB MODULAR FUNCTIONS

A sub modular function is a function that satisfies the property of diminishing returns. This means that the marginal increase in the function's value decreases as the input set grows. More formally, given a set X and an element x not in X, the function f(X) is said to be submodular if for any subset $Y \subseteq X$, it holds that $f(X) - f(Y) \geq f(X \cup \{x\}) - f(Y \cup \{x\})$.

Sub modular functions have several useful properties:

1) They can be minimized efficiently using greedy algorithms, which have a polynomial time complexity.
2) They are naturally motivated by many combinatorial optimization problems, such as set cover, maximum cut, and facility location.
3) They can be used to model various real-world problems such as sensor placement, image segmentation, and natural language processing.

Sub modular functions are also related to the concept of convexity, but unlike convex functions, sub modular functions are not necessarily continuous or differentiable, and their global minimum may not be unique.

Sub modular functions can be used in VLSI circuit design to optimize the placement and routing of components on a chip. These functions have the property of diminishing returns which allows for efficient optimization algorithms to be used to find near-optimal solutions. One example of using sub modular functions in VLSI design is in the context of physical synthesis, where the goal is to find an optimal placement of the components on the chip while minimizing the total wire length. Sub modular functions can be used to model the objective function, such as the total wire length, and then used to optimize the placement of the components on the chip. These functions have the property of diminishing returns, meaning that adding a new component to a cluster would result in a smaller increase in the objective function than adding the same component to a smaller cluster. This allows for efficient optimization algorithms such as greedy algorithms to be used to find near-optimal solutions.

Sub modular functions can be used in the context of functional synthesis, where the goal is to find an optimal mapping of the logic gates of a circuit onto the physical components of a chip. Sub modular functions can be used to model the objective function such as the total power consumption of the circuit and then used to optimize the mapping of the gates onto the components.

Hence, sub modular functions are used to model the objective function and optimize the placement and routing of the components on the chip. These functions have the property of diminishing returns which allows for efficient optimization algorithms to be used to find near-optimal solutions.

V. DIFFERENTIAL EQUATIONS

Calculus, in particular differential equations can be used in VLSI design to model and analyze the behavior of analog circuits. [8].

Differential equations are used in VLSI design for small-signal analysis to study the linearized behavior of a circuit around an operating point. Small-signal analysis can be used to model the circuit using a set of differential equations, where the elements of the equations represent the resistances, capacitances, and inductances of the circuit and the elements of the vector represent the voltage and current sources. By solving the differential equations, it is possible to obtain the small-signal transfer function of the circuit, which describes how the circuit responds to small perturbations around the operating point.

It is also used in circuit simulation tools such as SPICE (Simulation Program with Integrated Circuit Emphasis) that uses differential equations to model the behavior of the circuit. SPICE uses a set of nonlinear differential equations to model the circuit, where the elements of the equations represent the resistances, capacitances, and inductances of the circuit and the elements of the vector represent the voltage and current sources. These equations are solved numerically to obtain the voltage and current in the circuit over time. Thus, differential equations can be used to model the behavior of analog circuits in VLSI design and analyze the circuit's performance and stability.

VI. ABSTRACT ALGEBRA

It is a branch of mathematics that studies mathematical structures that are defined by a set of operations and relations. These structures can include groups, rings, fields, and vector spaces.

In Very-Large-Scale Integration circuit design, abstract algebra is used to design and analyze digital circuits. For example, Boolean algebra, which is a type of algebra used to model and analyze logical systems, is used to design and optimize digital logic circuits. Group theory, another branch of abstract algebra, is used in the study of symmetries in digital circuits, which can help simplify and optimize the circuit design. In addition, algebraic coding theory, which is the application of abstract algebra to the study of error-correcting codes, is used in the design of digital circuits to ensure that the data transmitted is received correctly.

Overall, abstract algebra provides a powerful toolset for modeling and analyzing digital circuits, which can help simplify and optimize the design process.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

VII. SYMBOLIC METHODS

Symbolic methods in modeling use mathematical symbols and equations to represent the behavior of a system, rather than using numerical simulations. These methods can be used to analyze and verify the behavior of a system, by providing a way to mathematically reason about the system's properties.[10]

One important theme in symbolic methods is decomposition, which refers to breaking down a complex system into simpler components that can be analyzed separately. This theme is ubiquitous in the field of formal verification, where the goal is to prove that a system behaves correctly. By decomposing a system into simpler components, it is often possible to prove properties of the system more easily.

Decomposition in symbolic methods is used in abstraction model checking. Model checking is a technique used to formally verify the behavior of a system by constructing a mathematical model of the system and checking it against a set of desired properties. Abstraction is a way to simplify the model by ignoring certain details that are not relevant to the properties being checked. By abstracting away unnecessary details, it becomes easier to check the properties of the system and often allows the analysis of much more complex systems.

Use of refinement in the context of design and verification of hardware systems can also be done with symbolic methods. Refinement is a process of transforming a high-level model of a system into a lower-level one, while preserving the properties of the system. This allows to reason about a more abstract and simpler representation of the system, and then gradually refines it to a more concrete one, while ensuring that the important properties are preserved.

In summary, symbolic methods in modeling use mathematical symbols and equations to represent the behavior of a system, and decomposition theme is a common theme in these methods which allows simplifying and reasoning about complex systems by breaking them down into simpler components.

VIII. VEDIC MATHEMATICS

With development in technology, it has become the need to have a multiplier and divisor that should be robust in design and logic yet ensuring speed and accuracy that accomplish the demands of VLSI technology.[1] Conventional method used for multiplication operation such as array method, Wallace tree or booths etc are not very efficient in speed and power consumption**.** Similarly, division operation is one of the expensive and most time consuming operations of the modern processors amongst all elementary operations. Vedic Mathematics proposed various sutras that can be applied for binary multiplication and division, which are time efficient and well optimized methods as compared to conventional one. Urdhva Tiryakbhayam Sutra, Nikhilam Sutra are efficient multiplication algorithms while Paravartya, Nikhilam and Dhvajanka are few division algorithms which are more efficient algorithms compared to conventional division algorithm. Floating point multiplication, linear convolution, cryptography, FFT calculation can be done with Vedic mathematics with high speed and low power consumption. Though research in this field is very less, it has great potential to grow in VLSI field.

IX. CONCLUSION

Mathematics plays a crucial role in VLSI (Very-Large-Scale Integration) circuit design, as it provides the tools and methods for modeling, analyzing, and verifying the behavior of digital circuits. Boolean algebra is used to design and optimize digital logic circuits, which are the building blocks of VLSI systems. Boolean algebra provides a way to represent and manipulate logical expressions and equations, which can be used to model the behavior of digital circuits and to optimize their performance. Linear algebra is used for the analysis of large systems of equations and for the manipulation of matrices. Linear algebra is used in the analysis of signal-flow graphs and in the study of network theory, which are important tools for the analysis and optimization of digital circuits. Abstract algebra is used to study error-correcting codes, which is used in the design of digital circuits to ensure that the data transmitted is received correctly. Group theory, another branch of abstract algebra, is used in the study of symmetries in digital circuits, which can help simplify and optimize the circuit design. Furthermore, optimization techniques, such as integer linear programming and convex optimization are essential in the circuit design. These techniques are used to find the optimal solutions to problems that arise in the design of digital circuits, such as minimizing power consumption or maximizing performance. Graphs and hypergraphs can be used to model various problems in VLSI (very-large-scale integration) design, such as circuit layout, routing, and testing. In summary, mathematics plays a critical role in VLSI circuit design, providing the tools and methods for modeling, analyzing, and verifying the behavior of digital circuits, which are essential to the design and optimization of VLSI systems.

REFERENCES

[1] Anuradha Singh, Neetu Gupta, “VEDIC MATHEMATICS FOR VLSI DESIGN: A REVIEW“ International Journal of Engineering Sciences & Research Technology, March, 2017, ISSN: 2277-9655

[2] S.Ammu, Remya Ajay “VLSI implementation of Boolean algebra based cryptographic algorithm” March 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) DOI: 10.1109/ICEEOT.2016.7755059

[3] MUHAMMAD SAJJAD , MOHD ZUKI YUSOFF , NORASHIKIN YAHYA AND ALI SHAHBAZ HAIDER “An Efficient VLSI Architecture for FastICA by Using the Algebraic Jacobi Method for EVD” VOLUME 9, 2021, IEEE Access

[4] Tomasz S. Czajkowski, Member, IEEE, and Stephen D. Brown, “Functionally Linear Decomposition and Synthesis of Logic Circuits for FPGAs” IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 27, NO. 12, DECEMBER 2008

[5] Tamas Terlaky, Anthony Vannelli, Hu zhang On Routing in VLSI Design and Communication Networks, March 2005, Proceedings of the 16th international conference on Algorithms and Computation DOI:10.1007/11602613_104

[6] Elizabeth Kalinina, Gennady M Khitrov, Srrgei Pogozhev, “Linear algebra methods in graph theory” October 2015 DOI: 10.1109/SCP.2015.7342228 Conference: 2015 International Conference "Stability and Control Processes" in Memory of V.I. Zubov (SCP)

[7] George Karypis, Rajat Aggarwal, Vipin Kumar, and Shashi Shekhar “Multilevel Hypergraph Partitioning: Application in VLSI Domain” DAC 97 - 06/97 Anaheim, CA, USA

[8] Hussein Chiblè and Ahmad Ghandour, “DIFFERENT MATHEMATICAL RELATIONS WITH CMOS VLSI CIRCUITS” Lebanese Science Journal, Vol. 8, No. 1, 2007

[9] Mohanapriya Marimuthu, Santhosh Rajendran, Reshma RadhaKrishnan etal “Implementation of VLSI on Signal Processing-Based Digital Architecture Using AES Algorithm” January 2023 CMC **Computers, Materials & Continua** DOI:10.32604/cmc.2023.033020

[10] Xiang-Dong Tan, C J. Richard Shi, “Hierarchical Symbolic Analysis of Analog Integrated Circuits via Determinant Decision Diagrams” IEEE TRANSACTIONS ON COMPUTER AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 19, NO. 4, APRIL 2000