FPGA-Implementation of Electronic Voting Machine (EVM)

Abhishek Kumar
Electronics and Electrical, Lovely Professional University, Phagwara, Punjab, India.
Email: abkvjti@gmail.com (Corresponding author)

ABSTRACT:
The voting process is one of the most significant areas that benefits from technological growth, and the development of digital technology has changed many other industries. Electronic voting machines (EVMs) are an essential part of traditional voting systems because of their effectiveness, accuracy, and ease of use. The design, architecture, and advantages of the FPGA (Field-Programmable Gate Array) implementation of an electronic voting machine over conventional voting systems have been explored in this work. EVMs based on FPGA have benefits in terms of reliability, flexibility, security, and speed. The elements of the FPGA implementation, the design process, and the difficulties in maintaining data integrity and thwarting tampering are also covered in the article. The goal is to demonstrate how FPGA technology can be applied to build a voting system that is safe, reliable, and effective, therefore enhancing the election process. FPGA-based EVM describes more complex functionalities and enhances performance and electronics voting application

KEYWORDS: EVM, Voting, FPGA, Counting, Efficient design

1. INTRODUCTION

In democratic cultures, the voting process is essential, and it is crucial to guarantee its correctness and integrity. Voting is the foundation of democratic systems, where citizens elect representatives based on majority opinion. Historically, elections have used paper ballots, a process vulnerable to issues like vote tampering, errors in counting, and time inefficiency [1]. Electronic voting machines (EVMs), which offer automated, precise, and quick voting and counting, were created to overcome these problems. To address these challenges, the shift toward electronic voting has gained prominence, and the EVM has emerged as a viable solution. EVMs not only streamline the voting process but also ensure better data integrity and security, which is critical for fair election outcomes [2]. Despite the advantages of EVMs, concerns remain regarding security, particularly the risks associated with removable storage devices like memory cards, which can be tampered with. To address these security concerns, we present a digital EVM system designed using Verilog HDL and implemented on FPGA [3], thereby eliminating physical storage vulnerabilities and introducing digital security measures like password protection and voter verification. Field-Programmable Gate Arrays (FPGAs) have become a viable platform for EVM implementation in recent years. By reconfiguring logic and processing components, FPGAs provide the flexibility of hardware programming, allowing for the creation of adaptable and effective systems. FPGA-based EVMs are the perfect option for safe voting applications as they not only increase the voting process's dependability and security but also provide real-time processing and improved hardware control. A democratic society must prioritize the efficiency, security, and integrity of the election process [4]. Ensuring a dependable and safe voting process has become crucial due to the growing usage of electronic voting systems. FPGA-based systems are more secure than software-based ones, they can still be attacked via reverse engineering or side-channel attacks. The EVM can be protected using countermeasures including secure boot sequences, tamper-resistant designs, and the application of cryptographic algorithms. The voting process's security and integrity are crucial. In this regard, FPGA-based EVM are beneficial as they enable the incorporation of strong security features at the hardware level [5].

(a) Data Encryption: Encryption methods can be implemented directly on the FPGA to guarantee that vote data is safely sent and saved. As a result, attackers will find it more difficult to intercept and alter vote data.

Paper type: Research paper
https://10.71822/mjtd.2025.1194981
Received: 30 December 2024; Revised: 22 January 2025; Accepted: 23 February 2025; Published: 1 March 2025
How to cite this paper: A. Kumar, **"FPGA-Implementation of Electronic Voting Machine (EVM)"**, *Majlesi Journal of Telecommunication Devices*, Vol. 14, No. 1, pp. 31-38, 2025.

(b) Tamper Detection: FPGA-based systems can contain elements to detect physical tampering, such as voltage, temperature, or logic state sensors, to warn authorities of potential security breaches.

(c) Audit Trails: Election results may be transparently verified by using FPGAs to create audit trails, which guarantee that every vote is documented together with its timestamp and related security hash.

Processing massive volumes of data in real-time, FPGAs may use a lot of electricity. Low-power FPGAs or power-efficient design strategies, including lowering clock frequencies or improving the logic architecture, can reduce this problem. Compared to conventional voting machines, FPGA-based solutions might be more costly and complicated to build. However, the higher initial cost is justified by the long-term advantages, which include reconfigurability, scalability, and improved security. The remainder of the article is organized as follows, literature review is included in section 2. The design and implementation of EVM with FPGA using Verilog HDL is illustrated in section 3. The result and analysis is preented in section 4 and finally concluded in section 5.

2. LITERATURE REVIEW

The need of speeding up the vote-counting process and minimize human error pushed the early deployment of EVMs. First-generation electronic voting systems relied on embedded microcontrollers, however security problems soon emerged due to these systems' susceptibility to software flaws. . A more reliable and secure solution was provided by the use of FPGA technology into the EVM architecture. The shift from software-based to hardware-based EVM systems has been covered in a number of studies. According to research in [6], FPGA-based EVMs might address several issues with microprocessor-based systems, including high maintenance costs because of the need for regular software upgrades and susceptibility to software hacking. FPGAs provide an alternative by putting into practice extremely configurable, safe, and tamper-resistant hardware-based solutions. FPGAs offer significant advantages for designing voting systems due to their customizable nature and ability to perform parallel processing. Because of their versatility and capacity for parallel processing, FPGAs provide several benefits when it comes to voting system design. FPGAs are configured to perform certain functions, such as vote counting, encryption, and security protocols, directly within the hardware thanks to the flexible design of FPGAs. In recent years, several FPGA implementations for EVMs have been investigated. One remarkable example is the work of [7] suggested an electronic voting system based on FPGA, in which the votes were safely saved in on-chip memory and the FPGA counted the votes in real-time. An encryption module was also included in their design to ensure the votes' privacy. Research in [8] conducted additional research that concentrated on creating an FPGA-based EVM with a built-in real-time voter validation mechanism. This concept had a counting device that offered real-time vote tallying, a secure memory block for storing votes, and a control unit to oversee voting sequences. According to their research, using FPGA gave more flexibility in meeting different election needs, such as supporting several candidates and using different vote-counting techniques.

The capacity of FPGAs to include tamper detection and encryption directly into the hardware is one of their advantages. The application of FPGA in establishing secure communication protocols within EVMs was investigated in research conducted by [9]. To safeguard vote data while it is being sent and stored, they included cryptographic techniques like Advanced Encryption Standard (AES) into the FPGA hardware. Effective real-time encryption and decryption were made possible by the FPGA's built-in parallelism, guaranteeing vote integrity throughout the voting procedure. Research in [10] focused on the FPGA-based EVMs' real-time processing capabilities. The total time needed for vote tallying was significantly reduced by their FPGA implementation's ability to process several votes in parallel. Additionally, their idea showed how FPGA systems may be scaled to manage elections of different sizes, from national elections with a big voter base to local elections with fewer candidates. The concept of electronic voting has evolved significantly over the years. In authors [11] described an early voting system that could automate the recording of votes without manual handling, laying the foundation for secure digital voting systems. Electronic systems developed in [12] presented an electronic device that eliminated human handling of votes, reducing the risk of fraud and error. Further advances in EVM technology were noted in [13], who described programmable devices capable of handling complex election scenarios, including those requiring multiple votes per voter. Authors in [14] introduced the concept of biometric security in EVMs, using fingerprint recognition to further secure voting processes. Each of these studies contributed critical insights into the development of secure, efficient, and adaptable EVMs that meet the diverse needs of various electoral systems.

3. DESIGN AND ARCHITECTURE OF FPGA-BASED EVM

The goal of an FPGA-based Electronic Voting Machine's (EVM) architecture and design is to create a reliable, safe, and effective system for collecting, storing, tallying, and sending votes during an election. High-speed parallel processing and reconfigurability are two benefits of FPGA technology [15,16] that make it perfect for applications requiring fault tolerance, security, and real-time data processing. Below, we discuss the design considerations, components, and architecture involved in an FPGA-based EVM. Fig 1 presents the EVM system is developed in three distinct stages, ensuring security and accuracy throughout the voting process. Designed using Verilog HDL, this EVM can be implemented on an FPGA, allowing for easy reprogramming and cost-effective adaptation across various elections.

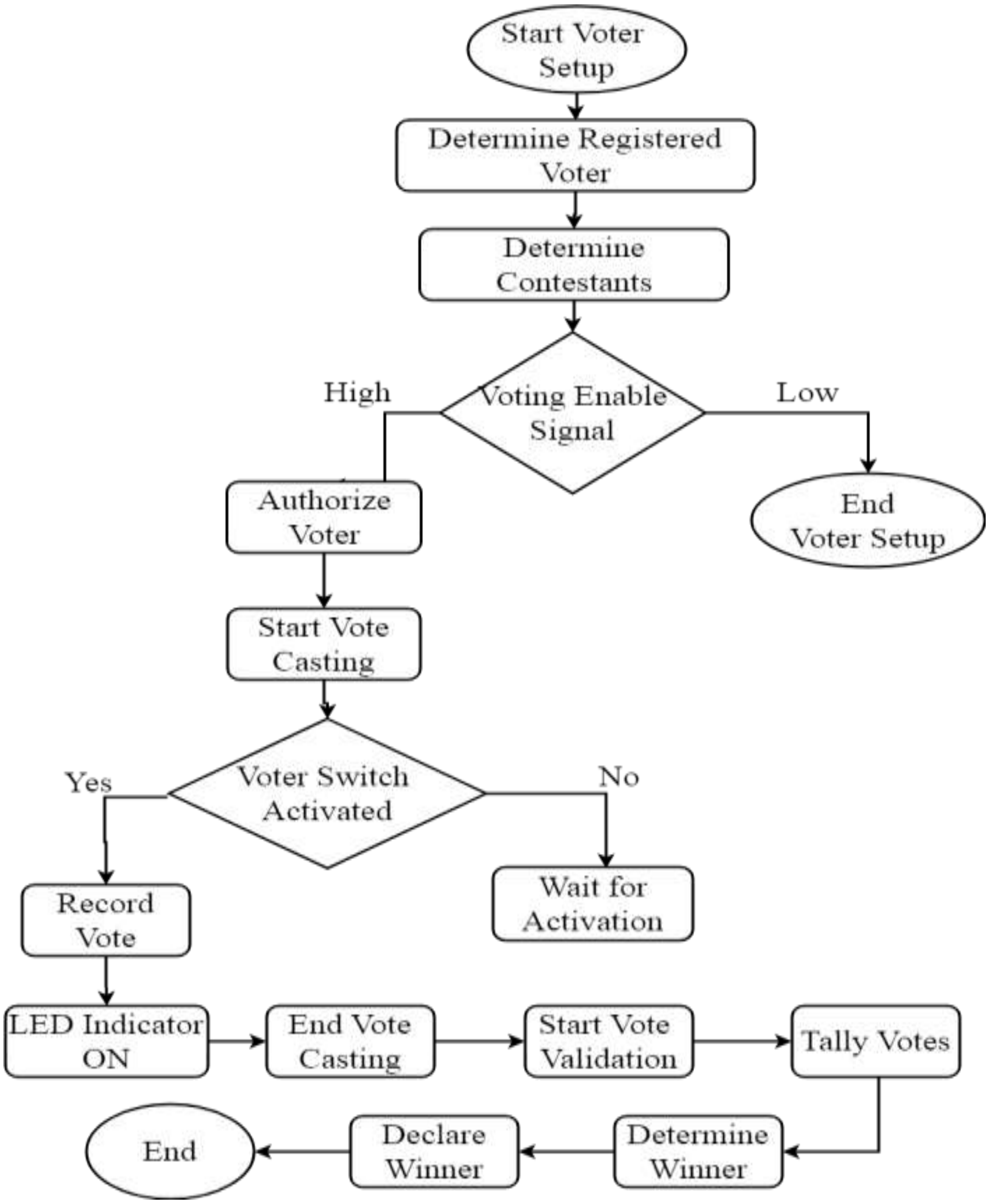


Fig. 1. Flow chart of EVM process.

Stage 1: Voter Setup At the beginning of the election, the total number of registered voters and contestants is determined. An active high input signal, termed Voting Enable, authorizes each voter to cast a single vote, ensuring the validity of the voting session.

Stage 2: Vote Casting The voting process begins when the Voting Enable signal is high. Voters select their preferred candidate by activating the Voter Switch, which records the vote in the corresponding candidate’s registry. Visual confirmation through an LED indicator assures the voter that their vote has been successfully cast.

Stage 3: Vote Validation and Result Declaration Upon completion of voting, the system verifies the results by tallying votes stored in each candidate’s registry. The candidate with the highest count is declared the winner, and process ends.

Field-Programmable Gate Array (FPGA) can be configured to carry out particular functions. In contrast to conventional microcontrollers or CPUs, FPGAs provide a high degree of parallelism and circuit design freedom. They are made up of an assortment of input/output blocks, interconnects, and programmable logic blocks (PLBs) that may be set up to carry out a broad range of tasks. FPGAs provide designers the ability to create unique hardware designs for particular uses, such as the creation of an electronic voting machine. The following are the benefits of implementing an electronic voting machine with FPGAs:

(a) Parallel Processing: FPGAs are excellent for high-speed jobs like encryption and vote counting

(b) Flexibility: The voting system may be updated and modified without requiring new hardware

(c) Security: Encryption and tamper detection methods at the hardware level,

(d) Real-time Operation: FPGAs can process data in real-time

Verilog HDL was chosen due to its hardware abstraction capabilities and compatibility with Xilinx Vivado tools. The design operates on a Artix 7 FPGA, which offers high configurability and cost efficiency, making it suitable for various election types. An FPGA-based EVM integrates multiple functionalities into a single hardware platform, from voter input to vote storage, counting, and result declaration. A polling officer controls the Voting Enable signal, which is only enabled during legitimate voting times, the system clock makes sure that the whole voting process is synchronized. The voter can choose from the three candidates after the Voter Switch is turned on. An LED indicator illuminates when a vote is cast, removing any doubt among voters and verifying that the vote was successfully registered. To visually track the number of votes cast for each candidate, the EVM uses three seven-segment displays, each of which is coupled to an active low signal (An0, An1, An2). This instant response simplifies the verification process and increases vote-counting transparency. The objective of the work is

(a) Verilog Hardware Description Language (HDL) to design and implement EVM.

(b) The EVM module mimics the voting mechanism, records votes for various parties, and shows the vote tallies.

(c) The system also allows voters to select "None of the Above" (NOTA).

(d) Additionally, LED indicators are incorporated into the design to verify legitimate votes and highlight incorrect inputs if an improper switch combination is discovered.

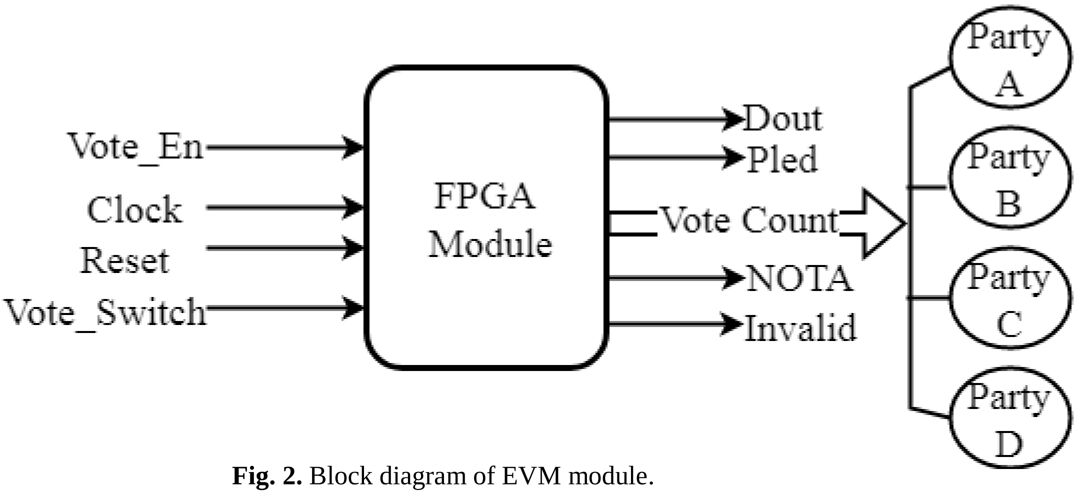


Fig. 2. Block diagram of EVM module.

The EVM module that you are designing includes a variety of features such as vote counting, LED control for visual feedback, and flag handling for invalid voting attempts. To manage the voting process, the EVM system is built with several input and output signals. Vo_en, which permits voting while it is active; first, a reset signal to set the system to a known state; clk, a clock signal for synchronization; and vo_switch, a 5-bit input that chooses which candidate or option the vote is cast for, are among the input signals. Pled, a 5-bit signal that controls the LEDs for each party and NOTA, and Dout, the overall aggregate of votes cast for all candidates and the NOTA option, make up the output signals. Furthermore, 8-bit registers hold the vote totals for the four candidates (Parties 1 through 4) as well as the NOTA option, and the invalid flag indicates if a vote was cast that is invalid. The design contains parameters, registers, and counters, such as LED_TIMER_MAX, which determines the length for which the LED should remain after a vote is cast. Pled manages the LED indicators for each candidate and NOTA, while the Party1, Party2, Party3, Party4, and Nota registers keep track of the number of votes cast. After a vote, a counter called led_timer controls how long the LEDs remain light. The voting mechanism checks the vo_switch value to see which candidate the vote is cast for when vo_en is active. The appropriate LED in Pled is activated for the amount of time indicated by

LED_TIMER_MAX, and the related vote count is increased if the vote is legitimate. The invalid flag is raised if the vote is deemed invalid. After every vote, the led_timer decreases, and when the timer hits zero, the LED is turned off. Lastly, the sum of the votes cast for each candidate plus the NOTA choice is used to get the final vote total, Dout. Additionally, the `invalid` flag is raised (set to high) if an invalid value is detected on the `vo_switch` input, indicating that the vote does not correspond to a valid candidate or the NOTA option.

4. RESULT ANALYSIS AND VALIDATION

The simulation results in fig 3 confirmed that the proposed Electronic Voting Machine (EVM) design meets the functional requirements of a secure and reliable voting system. The RTL schematic in Fig 4 validated each stage of the voting process, demonstrating accurate vote registration, signal synchronization, and output verification. The final vote count displayed in the Dout register matched the total number of participants, ensuring there were no instances of overvoting or data discrepancies. The simulations showed that every vote was correctly registered and allocated to the chosen candidate's registry, with signal timings synced to guarantee that votes were cast just once and kept safe. To ensure there were no overvotes or discrepancies and that the final result accurately represented the total number of voters, it was cross-checked before being shown in the Dout register. By removing the possibility of duplicate or incorrect votes, this precise count in Dout demonstrates how effectively the system manages votes in a clear and reliable method. This successful validation procedure shows that the suggested EVM architecture is suitable for practical applications and further validates its resilience. The design may be used in a variety of election contexts, from small-scale municipal elections to major public elections, guaranteeing its usefulness in a range of situations. The results of the simulation demonstrate that the EVM system, which was developed with Verilog HDL, operates efficiently and reliably. To ensure accuracy, security, and adherence to design criteria, every system component underwent extensive testing in the Xilinx Vivado simulation environment.

The following are a few significant findings from the simulation results:

1. Vote Count Update and LED Feedback: The voting process increases the number of votes in the appropriate register (Party0, Party1, or Party2) when the system detects the selected candidate. The voter is given visible proof that their vote was correctly recorded simultaneously and LED is turned on. This feature guarantees the system's responsiveness to legitimate inputs while simultaneously boosting voter trust.

2. Handling of Invalid Inputs: The EVM is made to handle illegal or incorrect inputs efficiently. The system generates a invalid signal whenever it detects an incorrect signal, such as a repeated vote or an input that is outside of the permitted voting parameters. This guarantees that there are no increases in the vote count registers, thus only valid votes are counted. By eliminating vote tampering and reducing mistakes brought on by illegal inputs, this feature improves the EVM's security and accuracy.

3. Total Vote Display Accuracy: The total number of votes cast for each candidate is adequately shown in the EVM's Dot output. The system's capacity to consistently track and validate the total number of votes is demonstrated by the fact that the sum of the individual vote counts kept in each candidate's register (Party0, Party1, Party2) coincides exactly with the value displayed on the Dot. To maintain openness and validate the election results after the voting process, the total vote count must be calculated accurately.

4. Timing and Synchronization: The results additionally emphasize the system's exact timing and signal synchronization, especially when the voting enable signals and system clock are used. Because every vote is timed to the clock cycles, there is a lower probability of data corruption or miscounts and voter registration is reliable and orderly. This synchronization ensures precise and efficient system operation.

5. Candidate Result Declaration: After voting is over, the EVM system counts the votes and correctly declares the winner to be the candidate with the most votes. In accordance with the system's architecture, which relies on vote totals to correctly report outcomes, this step completes the voting process. Thus, each module's ability to produce safe, dependable, and accurate voting results is validated by the system simulation.

The Artix7 FPGA-based power consumption report presented in Fig 5 shows the total on chip power consumption is 0931 W, out of which 88% is dynamic power inficated the EVM is active and has maximum time. The power consumption during ideal phase is only 12%. Fig 6 presents the resousre utilization summary for the EVM requires LUT, slice registers, and IOBs. There is a significant difference between traditional EVM and FPGA-based EVM. Presented design describes more complex functionalities and enhances performance and electronics voting application.

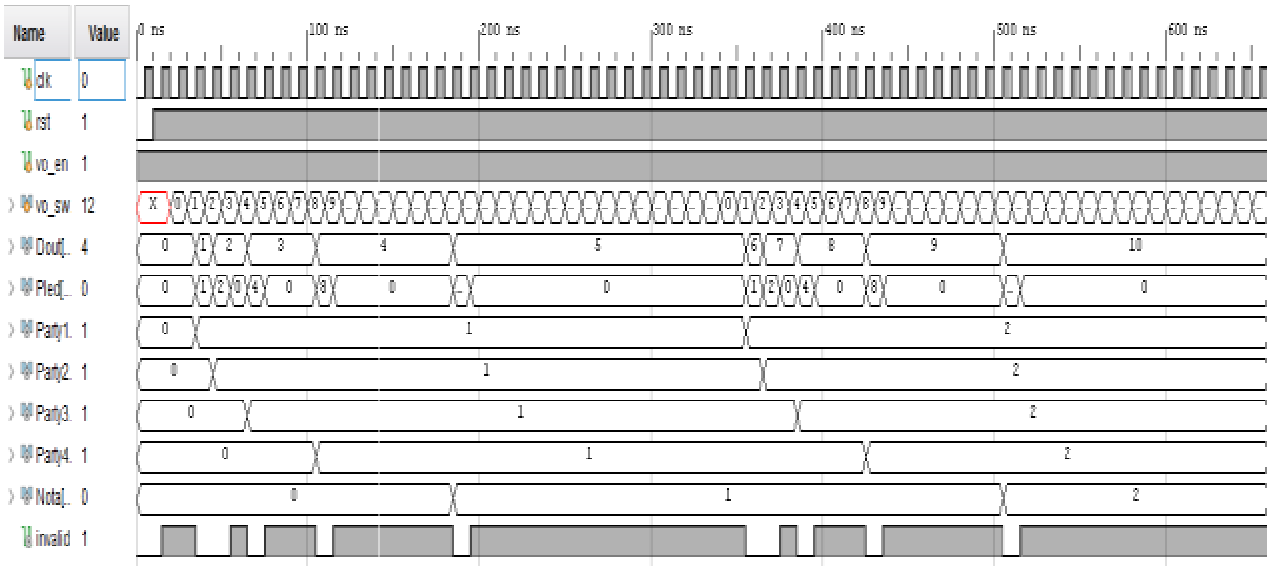


Fig. 3. Simulation waveform of EVM.

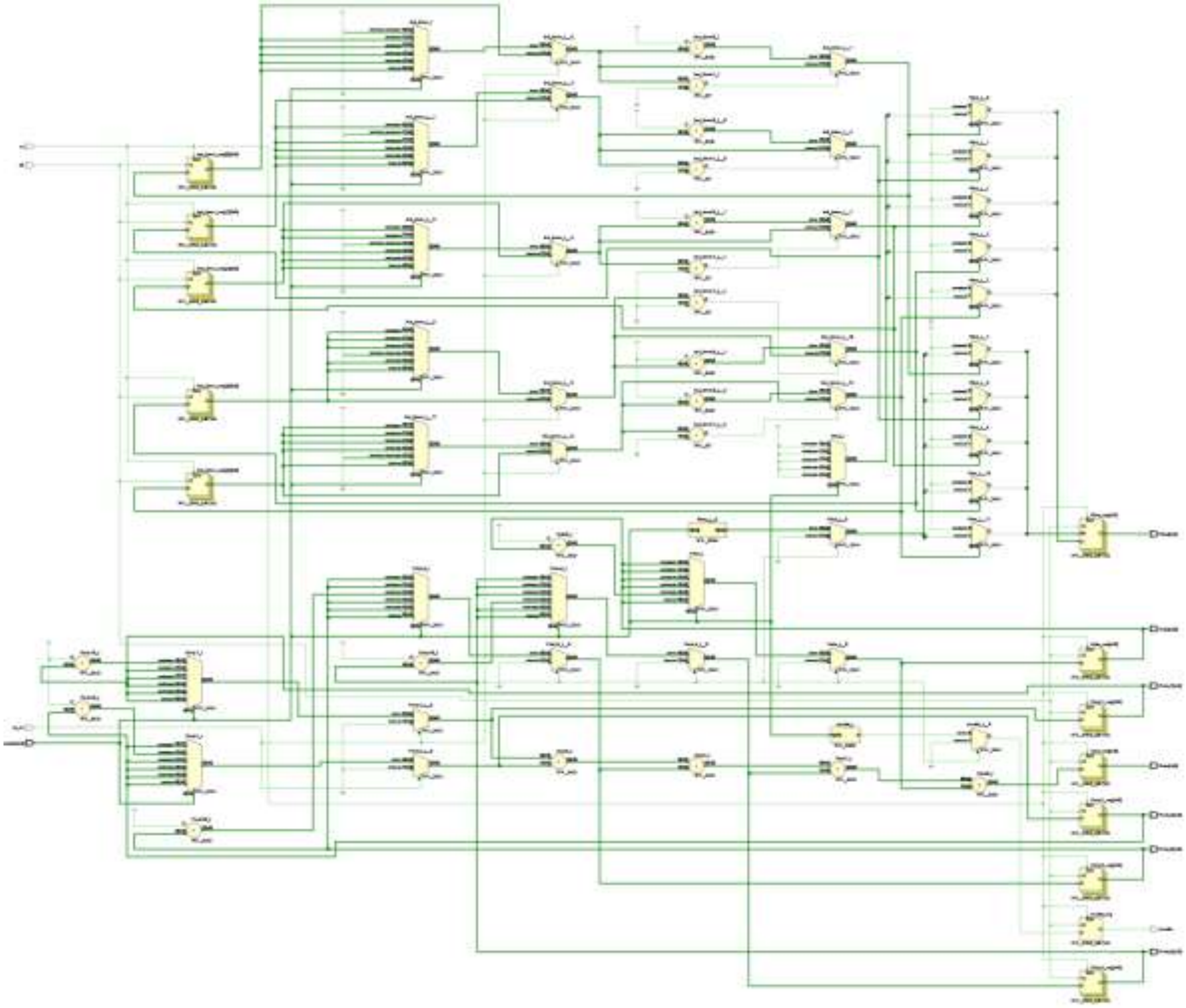
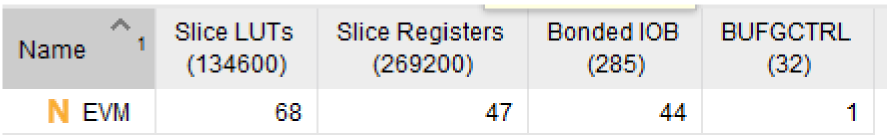
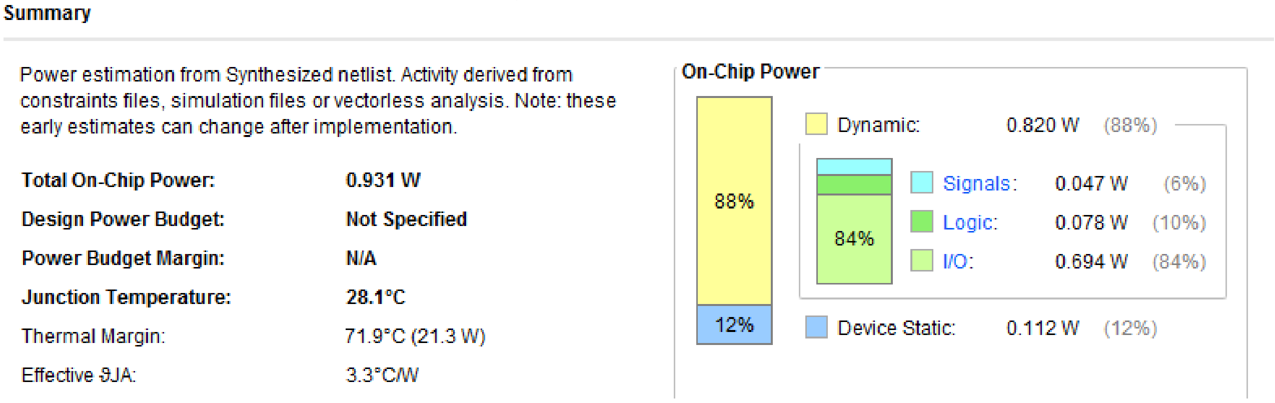


Fig .4. Synthesized RTL diagram of EVM.



Name	Slice LUTs (134600)	Slice Registers (269200)	Bonded IOB (285)	BUFGCTRL (32)
N EVM	68	47	44	1

Fig. 5. EVM power profile report.



Summary

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power:	**0.931 W**
Design Power Budget:	**Not Specified**
Power Budget Margin:	**N/A**
Junction Temperature:	**28.1°C**
Thermal Margin:	71.9°C (21.3 W)
Effective ϑJA:	3.3°C/W

On-Chip Power

Dynamic:	0.820 W	(88%)
Signals:	0.047 W	(6%)
Logic:	0.078 W	(10%)
I/O:	0.694 W	(84%)
Device Static:	0.112 W	(12%)

Fig. 6. Resource utilization summary of FPGA implementation of EVM.

5. CONCLUSION

This Verilog HDL-based EVM design meets the critical requirements for secure and efficient voting, providing an adaptable solution for various election types. Future enhancements include integrating biometric authentication to strengthen voter verification and prevent unauthorized access. Additionally, the design could incorporate real-time data transmission for remote election monitoring and statistical analysis, paving the way for fully digitized and auditable electoral processes. Implementation of EVM with FPGA technology offers benefits like increased speed, flexibility, security, and scalability. Customizable and effective voting machines that can manage complex tasks like vote counting and encryption in real time are made possible by FPGAs. Any election system must have strong security and tamper-resistance measures and FPGA-based solutions provide reliable ways to guarantee the voting process's integrity. The advantages of FPGA-based EVMs make them a viable alternative for the future of electoral systems, helping to create more secure, transparent, and effective elections in spite of the difficulties associated with power consumption and development complexity.

REFERENCES

[1] Ohize, Henry O., et al. "Blockchain for securing electronic voting systems: a survey of architectures, trends, solutions, and challenges." *Cluster Computing* 28(2),pp 132. 2025

[2] Oyelude, Olufunmilayo, and Ibukunoluwa Olojede. "**Evaluating the effectiveness of electronic voting systems in Nigeria: Challenges and opportunities**." *African Journal of Politics and Administrative Studies* 16(2), pp 84-104, 2023

[3] Swapna, Tangelapalli, Dharmavaram Asha Devi, and Manu Gupta. "**Digital Electronic Voting Machine Using Verilog**." *Disruptive technologies in Computing and Communication Systems*. CRC Press,pp 425-431, 2024.

[4] Keerthi, T., et al. "Real Time Implementation of Biometric-based EVM System for Distinct Verification." *Procedia Computer Science* 230, pp 407-416.2023.

[5] Molina, Romina Soledad, et al. "High-level synthesis hardware design for fpga-based accelerators: Models, methodologies, and frameworks." *IEEE Access* 10, pp 90429-90455, 2022

[6] Kalra, Hariom, et al. "**Impregnable Electronic Voting Machine Harnessing the Power of FPGA Zynq 7000**."IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI). Vol. 2., 2024.

[7] Soomro, Zeeshan Ahmed, et al. "**FPGA Based Real-Time Face Authorization System for Electronic Voting System**." *3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pp 1-6 2020.

[8] Khan, Areeb Adnan, et al. "DigitalNizam: Shaping the Future of Pakistani Elections through Secure Field-Programmable Gate Array-Based Voting Solutions." *Engineering Proceedings* 46(1) pp 44, 2023.

[9] Zhang, Jiliang, and Gang Qu. "**Recent attacks and defenses on FPGA-based systems**." *ACM Transactions on Reconfigurable Technology and Systems (TRETS)* 12(3), pp 1-24, 2019.

[10] Drimer, Saar. "**Volatile FPGA design security–a survey**." *IEEE Computer Society Annual Volume*, pp 292-297, 2008.

[11] Coşgun, Ercan, and Anıl Çelebi. "**FPGA based real-time epileptic seizure prediction system.**" *Biocybernetics and*

Biomedical Engineering 41(1), pp 278-292, 2021

[12] Drimer, Saar. "**Volatile FPGA design security–a survey**." *IEEE Computer Society Annual Volume*, pp 292-297, 2008

[13] Kumar, Mritunjay, et al. "**FPGA Based Voting Machine**." *Kilby*, 7th edition ,pp 100, 2023.

[14] Chaabane, Faten, et al. "Low power blockchained e-vote platform for university environment." *Future Internet* 14(9) pp 269, 2022.

[15] Tripathi, Suman Lata, Abhishek Kumar, and Mufti Mahmud. "**FPGA for secured hardware & IP ownership**." *IEEE international conference of electron devices society Kolkata chapter (EDKCON)*. IEEE, 2022.

[16] Kumar, Abhishek. "**VLSI implementation of Vedic multiplier**." *Design and Development of Efficient Energy Systems, pp* 13-30, 2022