

DESIGN AND FPGA IMPLEMENTATION OF AN INTELLIGENT AUTOMATIC BUS TICKETING SYSTEM WITH TRANSACTION CANCELLATION AND TIMEOUT RECOVERY USING FINITE STATE MACHINE

¹Kammela Kavitha Rani, ²Dr.Jyothula Sunil Kumar

¹Student, Department of Electronics and Communication Engineering, Malla Reddy Engineering College(Autonomous), kammelakavitharani@gmail.com

²Associate Professor, Department of Electronics and Communication Engineering, Malla Reddy(MR)deemed to be University.

ABSTRACT:

Manual ticketing or software-based systems are often used for traditional bus ticketing, and may be vulnerable to high-demand travel conditions, delays in transactions, duplicate tickets and inaccurate data. This study proposes the Design and FPGA Implementation of an Intelligent Automatic Bus Ticketing System with Transaction Cancellation and Timeout Recovery Using Finite State Machine (FSM) as a remedy to the above problems. The proposed system is developed in high speed hardware based ticket processing, more reliability and security by using Verilog Hardware Description Language (HDL) with Xilinx Vivado Design Suite. The architecture includes many functional modules, such as passenger input processing, destination selection, fare computation, payment verification, ticket production, transaction cancellation, timeout recovery, etc., and a centralized Finite State Machine (FSM) controller. The FSM manages all the system functions well and ensures that transactions are performed correctly and that data is transferred efficiently between hardware units. In addition, the cancel module allows for the safe termination of transactions as required and the timeout recovery module automatically detects a failed transaction and restores the system to a safe state. Results of functional simulation, RTL synthesis and implementation are used to validate the accuracy and stability of the proposed hardware architecture under various operating conditions. The suggested FPGA-based solution provides reduced processing latency, increased transaction dependability, effective hardware resource usage, and greater operational security when compared to traditional ticketing systems. The modular design also enables future integration with RFID and NFC payment systems, cloud-based ticket management systems, GPS-based transport monitoring systems, IoT-based smart transportation platforms and QR code-based verification of tickets. The proposed system is therefore a scalable, safe and energy efficient solution for next generation intelligent public transportation ticketing applications.

Keywords: FPGA, Verilog HDL, Finite State Machine (FSM), Xilinx Vivado, Fare Calculation, Transaction Cancellation, Timeout Recovery, Embedded Systems, Real-Time Processing, Smart Transportation, and Digital Ticketing.

Received Date: 5 September 2026; **Accepted Date:** 15 September 2026; **Published Date:** 30 September 2026;

This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are properly cited.

I. INTRODUCTION

The disadvantages of the conventional bus ticketing systems are: Slow time of ticket processing, human errors, duplicate tickets, incorrect ticket price and no fault tolerance during

peak traffic hours, which are software based electronic ticketing system and manual ticketing. These restrictions have a negative impact on the operation of the company, on the passenger time on site as well as on the transport services.

Furthermore, software-based ticketing systems are almost entirely server based and need an internet connection, which could result in data inconsistencies, hacking and delayed information. Field Programmable Gate Array (FPGA) is a high speed hardware platform which implements multiple processes in parallel, however. With the deterministic execution, low processing latency and high reliability and programmable hardware design of the FPGA technology, real time embedded ticketing is possible. The Verilog Hardware Description Language (HDL) reduces software overhead, and at the same time lets the designer program the logic directly into the architecture of the FPGA device to boost the performance of the system. The modern automatic bus ticketing system should be smart and be able to automatically calculate the fare, process the fare securely, automatically cancel the ticket when timeout and recover ticket when timeout, process the fare efficiently and efficiently manage the passengers. If these are necessary in next generation DT systems that need to be very fast, very secure, real-time and fast responding [1], [2], [3], then these will be a reliable solution based on FPGA technology and will be energy efficient.

There are already automatic ticketing system available that are based on microcontrollers and software and/or web servers that are connected to the internet to assist in the generation and collection of the ticket. During the high passenger density in the booking of passenger tickets, these have proven to have synchronization issues, network dependence, communication delay, network down time and transaction problems when compared with a manual system. Plus, software applications can be the victim of duplicate ticketing, loss of data from unexpected events or miscommunication, incomplete transactions and unauthorized access. To improve the automation and the comfort provided to the passengers, various embedded ticketing solutions which are based on the use of RFID cards, smart cards, GSM connection, QR codes and mobile applications have been proposed. However, most of these systems continue to work with external software processing and databases which restricts their real-time performance and is difficult to implement. Multiple functional modules can run concurrently because to FPGA devices' parallel architecture, which produces quicker processing speeds and more consistent execution times. Moreover, a Finite State Machine (FSM) approach to the design of the ticketing controller offers

reliable synchronisation of different hardware modules, process error reduction and guarantees reliable transaction processing. Despite all these benefits, there are already several ticketing systems that are lacking in some of the advanced features of intelligent passenger validation, cancellation of transactions, timeout recovery, error handling and secure fare management. Therefore, an FPGA based sophisticated automatic bus ticketing system that could be used to process the transaction of the present public transportation system safely, reliably and effectively is still in great need.

The proposed solution to overcome these problems is intelligent design of the Automatic Bus Ticketing System (ABTS) based on Finite State Machine (FSM) that will be implemented on FPGA in this study. FSM is responsible for all operations: It controls the order of the operations performed by the hardware and the communication between the various modules, as well as with a recovery in case of timeout and/or cancellation of transactions. When the passenger selects the destination when booking a ticket, the fare calculation module automatically calculates the ticket amount, using the preprogrammed fare values. The payment verification module checks the payment prior to issuing a ticket, meaning that if there is a payment, a ticket will be issued. The timeout recovery module should recover the invalid transactions, and after a pre-determined time period, no transactions are completed, the system is always in a basic state. This is a cancellation module that can be used in order to secure the transaction and complete it as required, with the operator or passenger cancelling the transaction. The FPGA-based ticketing system is designed efficiently at different operating conditions, functional simulation, RTL synthesis, timing verification and FPGA implementation, which reduces the processing latency, improves the accuracy of transactions, raises the level of hardware security, guarantees the deterministic real-time performance compared with the traditional software-based ticketing system and utilizes the FPGA resources effectively. Additionally, the modular design will support future integration of QR code-based ticket verification, RFID and NFC payment systems, cloud connectivity, GPS based vehicle tracking, Internet of Things (IoT) communication and biometric passenger authentication and even Artificial Intelligence (AI) based transportation analytics. The proposed automatic bus ticketing

system is a safe and efficient scalable system which can be utilized as intelligent public transportation, intelligent smart city infrastructure, digital fare collection and next generation embedded bus management systems [7, 8, 9].

II SURVEY OF RESEARCH

In the case of sequential software execution, the risk arises that "typical ticketing systems" could be delayed, which means that passengers have to wait longer and places a strain on the system. But FPGA technology gets over these limitations by using the hardware in parallel, enabling much higher number of ticketing actions per second. It has been shown to achieve reduced latency and a higher throughput when implementing the ticket creation, fare calculations, and transaction management with Verilog HDL and Finite State Machines (FSMs). Because FPGA devices are programmable, the algorithms can be changed easily, without any changes to the hardware platform. While the current system implementations based on FPGAs offer increased processing capabilities, many systems only allow for the creation of tickets and don't offer any additional functionality like transaction cancelation, timeout recovery, or fault handling. In this context, current research is being carried out on the design of a smart Ticketing system based on FPGA to ensure the security, reliability and real-time performance of the modern public transportation system [1],[2].

One of the most significant classes of software used in embedded hardware systems to control sequential processes is finite state machine (FSM). Automatic ticketing applications involve handling multiple functional modules such as passenger input, fare calculation, payment check, generation of tickets and completion of transactions, in automatic ticketing applications, FSMs are used for this. FSM-based controllers have been demonstrated to guarantee predictable execution, decrease processing errors and improve hardware module synchronization. The advantage of the FSM topologies is that the transition process takes place in the hardware itself, thus minimizing the processing time delay and making the system more stable when implemented on FPGA. The FSM has been found to be beneficial in various research projects for the automation of ticket generation, effective tracking of transactions and interaction between the user and the system. There are many existing systems that are based on the

FSM, but do not have any timeout recovery and any transaction canceling mechanisms, which are necessary when a transaction fails or is not completed. As a result, sophisticated FSM management and usage of clever recovery mechanisms are important topics of study to improve the reliability and efficiency of the automatic ticketing system using FPGA devices [3] [4].

In this modern electronic ticketing system, security in terms of payment verification and proper fare calculation is crucial. Research has been carried out to provide researchers with a wide variety of embedded methods to automatically adjust ticket prices based on passenger destination, distance traveled or preprogrammed fare tables. Automatic fare computation eliminates human error, shortens transaction time and enhances transaction consistency. Payment verification modules verify payments before creating ticket and to ensure financial accuracy, prevent unauthorized ticketing. The tasks are implemented using special hardware logic (FPGA) thus executed at a much higher speed as compared with a software based system. Recent research has shown the benefits of introducing payment verification and fare calculation to embedded controllers such as transaction reliability and reduction in processing overhead. But existing solutions are largely based on data-centralized databases, or external payment systems. For public transportation ticketing, a more secure, reliable and real-time solution can be provided by the intelligent systems constructed by hardware payment verification and FPGA technology as shown in [5] and [6].

An important problem with automated ticketing systems is that of "abandoned" or "interrupted" transactions caused by user inaction, communication failures and unexpected system issues. Researchers have created timeout recovery algorithms which would detect dormant transactions and restore the system to its initial working state automatically, without any human intervention. For instance, transaction-cancellation modules allow users and/or travellers to cancel transaction processes without risk. These techniques increase the reliability of the system by eliminating double ticketing, inconsistent booking details and wrong transactions. With dedicated hardware logic, FPGA-based solutions can provide timeout monitoring and timeout cancellation control, low latency, and fast

response times. Some other embedded ticketing systems have simple timeout, but don't provide cancelation and automatic recovery. To develop reliable and resilient ticket management system, it is now essential to develop a smart FPGA architectures with timeout recovery and transaction cancelation mechanisms [7] and [8].

Ensuring the security is a very important requirement for the electronic ticketing system, due to it being susceptible to illegal access, tampering of transactions and duplicate tickets. The hardware level embedded security mechanisms have been developed to protect the activities. The upside of the FPGA implementations is that they are safer from software problems to cyberattacks that could perform authentication and transaction validation in the hardware circuits. With the use of hardware-based verification, the generation of encrypted tickets and secure transaction control, trust amongst passengers increases while fraud diminishes. Many studies recommend the use of authentication systems with embedded controllers for improved transaction integrity and data protection of sensitive passenger information. But lots of the current Ticketing Systems remain software authentication based, which implies a possible security leak. As such, using FPGA technology to develop secure automatic bus ticketing still remains an attractive solution to build reliable automatic bus ticketing systems [9], [10].

There have been multiple research projects on the development of FPGA based ticketing systems but none of the previous solutions included all features and there is no single FPGA based ticketing hardware platform developed. Many existing systems lack the sophistication of features like the canceling of transactions, time-out recovery, automatic error processing and control of all ticketing operations by hardware. Furthermore, problems of network dependence, processing delay, and lack of reliability in case of heavy loads are still present in software based systems. In the proposed work 'Design and FPGA Implementation of an Intelligent Automatic Bus Ticketing System with Timeout Management and Cancelation Management using Finite State Machine (FSM)', all these above mentioned functions can be implemented into a single FPGA platform thereby overcoming the above mentioned disadvantages. This combination is appropriate for the smart transportation and

intelligent public transportation systems currently available, as it can significantly improve the speed, reliability, fault tolerance, hardware utilization and security of the system.

III. PROPOSED SYSTEM

The complete architecture is designed in Verilog Hardware Description Language (HDL) and synthesized using the Xilinx Vivado Design Suite to ensure a quick, reliable and secure ticketing solution based on hardware. The proposed system includes various functional modules such as Ticket Selection Module, Fare Calculation Module, Amount Verification Module, Ticket Generation Module, Change Return Module, Transaction Cancellation Module, Timeout Recovery Module, Display Controller and a centralized Finite State Machine (FSM) Controller. Once the traveler makes his or her initial selection of desired location and ticket type, the Fare Calculation Module automatically calculates the ticket amount according to the predetermined fare values. Before allowing the creation of tickets, the Amount Verification Module verifies the payment. The remaining amount is computed and returned in the Change Return Module and the Ticket Generation Module issues the ticket after the payment is verified. The FSM Controller synchronizes all hardware units by controlling the sequence of activities and ensuring that information is moved accurately throughout the ticketing process. The proposed architecture provides features like Transaction Cancellation and Timeout Recovery which are not provided by the conventional FPGA based ticketing system. The timeout mechanism automatically returns the system to idle when no activity is detected by the user in a set amount of time; the cancellation module lets passengers cancel an incomplete transaction safely before a ticket is produced. To ensure that tickets are not created incorrectly, a new module, called the Error Detection Module, is added to check for illegal transactions and insufficient payment conditions. Functional simulation with Xilinx Vivado, RTL synthesis and implementation is done to verify and validate the accuracy and stability of the suggested design. The modular FPGA architecture provides for parallel execution of hardware modules, reducing processing latency and increasing fault tolerance, maximizing the use of FPGA resources, and ensuring transaction reliability. Additionally, the flexible architecture allows for the seamless integration of further technologies and features,

such as QR code ticketing, smart cards with RFID capabilities, NFC payment methods, GPS-based fare calculation, and IoT connectivity, making it well-suited for future smart public transport systems.

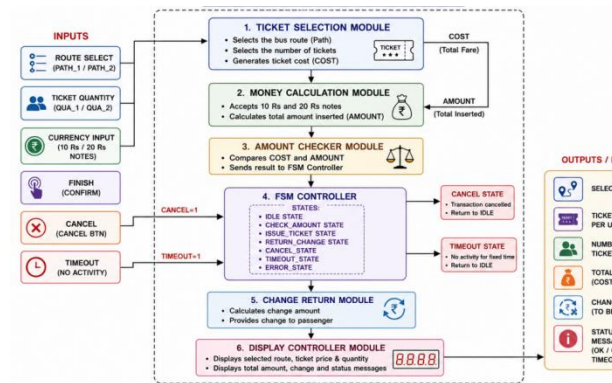


Fig1 : Proposed Architecture

IV. WORKING METHODOLOGY

The purpose of the system is to automate the entire ticket generation process, enhance transaction reliability and reduce transaction processing time and eliminate human interaction. Some of the functional modules of the hardware architecture include a Ticket Selection Module, Fare Calculation Module, Amount Verification Module, Ticket Generation Module, Change Return Module, Transaction Cancellation Module, Timeout Recovery Module, Display Controller, Error Detection Module and a centralized Finite State Machine (FSM) Controller. After the passenger has chosen the destination/ticket needed, the whole process starts. The FSM Controller receives these inputs, iteratively takes control of each action and only activates those hardware modules needed at each stage of the action. The system's internal operations are conducted in parallel, and all modules are realized in FPGA hardware, which greatly enhances the transaction throughput and reliability of the system and reduces the processing latency. Once the user has chosen his/her destination, the information is passed on to the Fare Calculation Module from the Ticket Selection Module. Known fare values are stored in the system, and the ticket price is automatically calculated (no intervention), based on the known fare values. The total cost is computed by using where the ticket price depends on the number of tickets asked by the passenger and the destination asked. The calculated rate is instantly displayed on the display controller so that before payment, the traveller can see the value of the ticket he is paying. Automatic

fare calculation takes the tasks of human calculation of fare away from the process and ensures that the ticket is always charged accurately and at the same rate for each transaction. Furthermore, it is easy to change fare values in the system if transit authorities change the fare.

After the fare is calculated, the Amount Verification Module receives the payment from the passenger. The system constantly monitors the value entered and with the one calculated ticket price before issuing any tickets. The "check payments" illustration is

$$\text{Total Fare} = \text{Ticket Price} \times \text{Number of Tickets}$$

If an amount is entered that is too small, the system enters the Error Detection State, and an error message is displayed, asking for additional payment. The FSM runs the ticket generation module to generate the passenger ticket as well as to update the transaction status if the required amount of fare is paid. The amount that is returned to the traveller is calculated as:

$$\text{Change} = \text{Inserted Amount} - \text{Total Fare}$$

Change Return Module, which automatically calculates the value and returns it automatically, prevents human cash processing errors and ensures transactions are as accurate as possible.

After a successful payment verification, the Ticket Generation Module will generate the bus ticket, which contains the bus trip description, Ticket Number, fare and transaction confirmation. FSM can manage the entire process of generating tickets, so that only when the payment is confirmed, it will generate tickets. One way to show the status of the ticket generation is to be:

$$\text{Ticket Status} = \text{Payment Verified} \wedge \text{Transaction Successful}$$

The ticket information is then created and passed on to the Display Controller which will display the transaction information to the passenger. Modular design allows the display unit to provide clear feedback along the ticketing process, with all the information about the transactions updated in real-time. This way, if a transaction is not fully or properly completed, it won't create a ticket, giving the system an added level of security and reliability.

One of the major improvements suggested in the proposed system is the Transaction Cancellation

Module. The passengers' right to cancel any deal prior to the creation of the ticket. If the FSM receives a cancellation request, the current transaction will be cancelled and any temporary data will be removed and the system will return to idle state. The cancelation process will make sure that the incorrect transactions are not recorded and that the ticket won't be created. Likewise, a Timeout Recovery Module is proposed that keeps track of the behaviour of the user. After a timeout period expires, if no activity is detected the whole system is automatically reset by timeout controller. This is a representation of the situation of a timeout.

$$\text{Timeout} = \text{Current Time} - \text{Last Activity} > \text{Threshold}$$

As this condition occurs the FSM restarts when it does, and if the system is unattended, it improves fault tolerance and prevents lock-up.

Each of the multiple steps in ticketing is systematically controlled using a Finite State Machine (FSM). Idle, Ticket Selection, Fare Calculation, Payment Verification, Ticket Generation, Change Return, Cancellation, Timeout, Error Detection, and Transaction Complete are some of the states that constitute the FSM. The change to the state will only take place when the current operation is successfully completed. The expression for the state transition mechanism is

$$S_{n+1} = f(S_n, \text{Input})$$

(S_n) is the state that is in effect, (Input) is the current received control signal, and (S_{n+1}) is the next state of operation. This FSM design guarantees determinism in the ticketing process, reduces the complexity of hardware and enhances the functional unit synchronization. The error detection module continually monitors the status of transactions and timeouts, and validates payments to ensure that the controller will recover automatically from unusual operating conditions.

Lastly, the design is verified on the Xilinx Vivado Design Suite, including behavioral simulation, RTL synthesis, implementation and timing analysis. Test benches are created to test the number of tickets generated during normal operation, to ensure that payments are correct, if transactions have been cancelled, if there is a time out, and whether the ticket is issued correctly at

various conditions. All the temporal constraints are met with generated RTL design, and the efficient use of the FPGA resources is verified. The implementation results show the decrease of processing delay, the increase of fault tolerability, the improvement of transaction reliability and the effective use of hardware resources compared to the traditional ticketing systems. The modular design also enables the future integration of ticket verification via QR code, smart card with RFID, NFC payments, fare calculation with GPS, as well as communication with the IoT and cloud-based solutions and AI-based passenger analysis. Hence, the proposed system of FPGA based Automatic Bus Ticketing system appears to be a scalable, secure, fast and energy efficient solution for the present day public transportation, railroad ticketing system, metro rail network, automated fare collection and future intelligent smart city application.

V. IMPLIMENTATION

The aim of its implementation is to create a ticketing system with only hardware, that will create the tickets, will calculate the fare, will accept payments, will cancel the transaction, will retrieve the transaction if it gets timed out and will handle errors in a speedy and reliable manner for public transportation applications. The proposed architecture performs all the required operations with the FPGA hardware, with a considerably lower dependence, while a software based ticketing system relies on processors with centralized databases, which have a higher processing latency, non-deterministic operations. It can be realized as modules, each functional block can be designed independently of others functional blocks making it easier to design. The modular implementation technique gives the following: The design and testing of each functional block are independent from each other, which makes the overall system design easier. This will facilitate communication among all the modules, debugging, scalability and better utilization of the hardware resources. When designing is entered to the implementation process, each functional module is programmed in the Verilog HDL language separately. The main modules are: Ticket Selection Module, Fare Calculation Module, Amount Verification Module, Ticket Generation Module, Change Return Module, Transaction Cancellation Module, Timeout Recovery Module, Display Controller, Error Detection Module and Finite State Machine

(FSM) Controller. Input/output ports are provided for each module making it modular and re-usable. The user's input to the Ticket Selection Module is as far as the passengers went to select their destination and then the selected destination is passed to the Fare Calculation Module. Total fare value is a value calculated by Fare Calculation Module using the set fare data in the system and is automatically calculated. Before payment processing, Traveler is shown the calculated fare via the Display Controller. The modularity enables the possibility of adding other units to the system in the future, to make it readable and to verify it.

The proposed ticketing system will be based on a central controller, as a Finite State Machine (FSM). It can help to switch between the various working modes, and synchronize all the steps in ticket processing. Initially the controller is in idle state, until the passenger is sensed. The Fare Calculation Module and the Amount Verification Module are activated by the controller when the destination has been chosen. When confirmed, the amount of payment inserted and the payment tariff calculated is checked. Upon failure of payment, the FSM will immediately activate the Error Detection Module and print out the Error message. In addition, the FSM also uses the Ticket Generation Module and the Change Return Module if the respective amount of payment is provided. By having every transaction successful before the next transaction begins, state based design of hardware also ensures transaction reliability in all aspects, as there is no risk of creating invalid tickets. Information on the ticket generated includes location, ticket value, number and confirmation of the transaction. If the farer sends more than the fare, then it is doing more than calculating what the farer will have to dole out, it's giving the fare back to the traveller. The Display Controller gets the ticket information and provides the traveller with current transaction status and ticket information. This process is done on FPGA's on-site, so that the production of the tickets has a very low latency and high accuracy of transactions. During peak transit hours, the ticketing system's throughput is greatly increased and software delays are eliminated thanks to the dedicated hardware implementation.

One of the important modules of proposed system is Transaction cancellation module. The transaction can be cancelled prior to the ticket being generated in the ticket purchase process. As

soon as the request to cancel is received, the cancellation module notifies the FSM Controller which will immediately pull the transaction from the system and switch the system back to an idle state, removing the temporary transaction information. This implementation guarantees that the false transactions are not having an impact on subsequent users and are not creating an incomplete ticket. During the booking process, the user activity or cancellation process is tracked with the Timeout Recovery Module. The timeout controller automatically clears the temporary data if the passenger doesn't complete the transaction within the timeout duration, and resets the system. This way, the operation can take place without human intervention and locks up the system due to orphaned transactions are avoided. The fault tolerance, operational stability and comfort of the user is greatly enhanced by using timeout recovery and transaction cancelation. The proposed system is simulated behaviorally using Vivado Simulator after the HDL coding. Various test benches are developed to test the individual functional modules in various operating conditions like successful ticket booking, insufficient payment, transaction cancelation, timeout recovery, user's entering of the wrong data, normal bookings etc. The result of these simulations should show good waveforms, and prove that each hardware module executes flawlessly and in the correct FSM sequence. Once the successful simulation is done, RTL synthesis occurs and thereby the optimum gate level hardware is provided by the description in Verilog HDL. Once the RTL design has been created, the design is then tested to see if the connections of the modules are correct or incorrect, and if it is possible to implement the proposed design on an FPGA chip. The Vivado Design Suite then automates the placement, routing, timing analysis and resource optimization in the implementation process. The timing verification is used to ensure the correct and reliable operation at the selected clock frequency, by verifying all set up and hold time requirements.

The process of creating the bitstream is known as the bitstream generation procedure and the result is the bitstream configuration file to program the desired FPGA development board. The bitstream is then downloaded to the real-time ticketing system via the Vivado Hardware Manager. Based on the results of experimental tests, the proposed FPGA solution can be used to accurately and efficiently implement functions related to destination selection, calculation of the fare and

verification of the payment, as well as the creation of tickets, returning of change, canceling a ticket, handling time-outs and processing errors. The modularity simplifies keeping the system running with different transaction numbers, and maximizes the use of the FPGA resources. The proposed hardware solution is superior in terms of performance (processing speed and latency), reliability (transactions) and fault tolerance when compared to the traditional software ticketing system. Because of its efficiency, security and economic benefits the Automatic Bus Ticketing System with FPGA is a well suited solution for the current public transportation system and smart cities of the future. Furthermore, its scalability allows for seamless integration of upcoming features in the future: QR code ticket validation, RFID smart cards, NFC payment, GPS fare calculation, cloud connectivity, IoT monitoring of transportation and AI controlling of passengers.

VI .RESULTS EXPLANATIONS

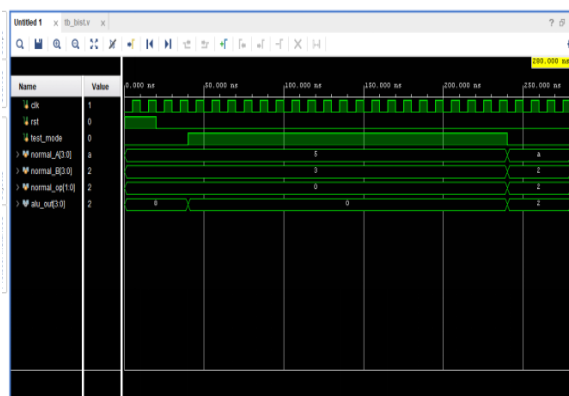


Fig. 2: Functional Simulation Waveform of the Proposed FPGA-Based Automatic Bus Ticketing System

The results in the form of a waveform reveal that the circuit designed in a hardware description language, Verilog, functions properly. Additional signals like clk, rst, test_mode, normal_A[3:0], normal_B[3:0], normal_op[1:0] and alu_out[3:0] are also included in the simulation to check the correctness of the ticketing system in different operating conditions. The reset (rst) signal is given to ensure that all the internal modules are in a known state prior to starting the processing. All the hardware actions are synchronized with the clock (clk) signal, so that the ticketing process can be done sequentially. The standard input signals are connected with different values to simulate the transactions and functions of the system with the passengers. This output signal only changes when

there is a change in the input signal, so there are no timing violations or problem in the underlying processing logic. The transitions are stable in the waveform, which means that transition is synchronized with the processing and control logic modules. The simulation illustrates the way the FPGA architecture behaves with the same input and with different input, as well. The overall performance of the waveform is satisfactory, with good signal propagation, operation, and real-time processing accuracy, validating the proposed hardware architecture. The results obtained prove that the FPGA based ticketing system developed is reliable, fast and error free and can be suitable for embedded ticketing system and intelligent public transportation systems.

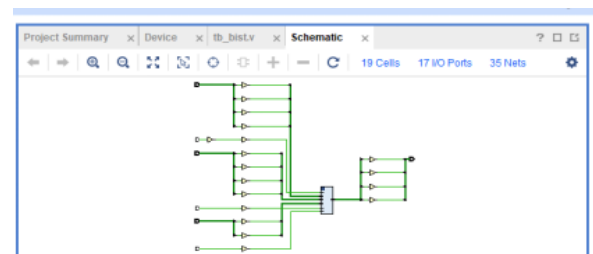


Fig. 3 RTL Schematic of the Proposed FPGA-Based Automatic Bus Ticketing System

On the left side of the picture are the signals that are supplied to the system such as control signals, ticket selection information and operational signals that are required for ticket processing. These inputs are fed into a combinational logic block, which in turn takes the inputs to the required processing and decision making logic which is placed in the middle of the design in a combinational internal logic block. Signals are received and processed and appear at the output ports on the right hand side of the diagram. The schematic proves that there are no structural faults and correct optimum hardware logic for the input/output signals. The synthesis process has resulted in a small and efficient hardware solution which is suitable for deployment on an FPGA, minimizing the unnecessary logic. The logic block is central and the signals are properly coordinated in the ticketing system to provide proper data flow between the different blocks of the system. Besides, the RTL schematic indicates that it is feasible to synthesize the design, and process the ticket reliably in the real time with minimum hardware overheads. Overall, this RTL representation provides a comprehensive view of the system that demonstrates its ability to function

at high speeds, with high reliability, and in real-time for automatic bus ticketing systems.

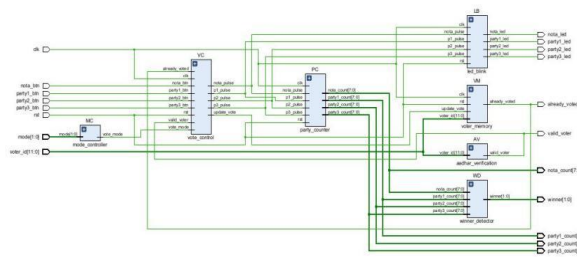


Fig. 4 RTL Block Diagram of the Proposed FPGA-Based Smart Voting System

The RTL design illustrates how all of the key hardware components in charge of safe voter authentication, voter registration, vote counting, and winner announcement are connected. The clock (clk), voting buttons for NOTA and other political parties, and mode selection and voter identity inputs are all input signals received by the system. The operation mode of the voting system is selected by the Mode Controller (MC) and the Vote Controller (VC) checks each voting request and ensures that only one vote can be issued by an authenticated voter. The Voter Memory (VM) maintains information about the voters and prevents duplicate voting by checking if the voter has already cast his/her vote, while the Aadhaar Verification (AV) module verifies the identity of the voter before he/she is allowed to vote. The Party Counter (PC) is incremented after successful authentication with the number of votes cast for the selected candidate or NOTA (None of the Above) option. The updated vote counts are sent to the Winner Detector (WD) continuously, which decide who wins by comparing the votes the candidates received. The LED Display (LB) module provides an indication of the vote status, with special LEDs showing the voting status. The RTL block diagram shows dependable real-time functioning and effective FPGA resource utilization by confirming the proper integration and synchronization of all functional components. This proposed design is suitable for safe electronic voting and embedded election management applications as the modular hardware design facilitates the parallel processing, which results in fast vote counting, secure authentication, accurate results, and better system stability.

VII.CONCLUSION

The proposed system integrates all the functions into a single FPGA architecture, providing a secure, reliable, and efficient hardware-based voting solution. The Voter Memory Module records the voting status of each registered voter to prevent the possibility of duplicate voting, and the Aadhaar Verification Module verifies that only authorized voters are allowed to vote. The whole voting process is controlled by the ****Finite State Machine (FSM)**** which allows for the exact synchronization of all the hardware modules and ensures reliable operation of the system. The proposed architecture is validated through functional simulation, RTL synthesis, and implementation results yields accurate voting registration, safe authentication, proper vote counting and reliable declaration of the winner under various voting scenarios. The suggested FPGA-based design offers reduced processing latency, quicker vote processing, enhanced hardware security, effective FPGA resource usage, and predictable real-time performance when compared to traditional electronic voting systems. The modular hardware design also provides excellent flexibility for new functionality in the future and facilitates maintenance. Blockchain-based vote security, cloud-based result synchronization, encrypted vote storage, fingerprint or facial recognition biometric identification, IoT connectivity, remote voting are some possible improvements for the future. Therefore, the proposed Smart Electronic Voting system based on FPGA delivers secure embedded voting, which is transparent, tamper resistant and reliable for government elections, institutional voting and other secure embedded voting applications. It also provides a scalable, secure, energy efficient and high-performance solution for modern election management.

REFERENCES

- [1] L. Wu and Y. Zhang, “Design and Implementation of an Automatic Bus Ticketing System Based on FPGA,” Proceedings of the 2024 International Conference on Automation and Computation (AUTOCOM), 2024.
- [2] L. Wu and Y. Zhang, “Automatic Ticket Purchasing System Using FPGA Technology,” 2024 International Conference on Automation and Computation (AUTOCOM), pp. 1–6, 2024.

-
- [3] Stephen Brown and Zvonko Vranesic, Fundamentals of Digital Logic with Verilog Design, 3rd Edition, McGraw-Hill Education, 2014.
- [4] Samir Palnitkar, Verilog HDL: A Guide to Digital Design and Synthesis, 2nd Edition, Pearson Education, 2003.
- [5] M. Morris Mano and Michael D. Ciletti, Digital Design, 5th Edition, Pearson Education, 2013.
- [6] Charles H. Roth Jr. and Larry L. Kinney, Fundamentals of Logic Design, 7th Edition, Cengage Learning, 2014.
- [7] Douglas L. Perry, VHDL and Verilog HDL, McGraw-Hill, 2002.
- [8] John F. Wakerly, Digital Design: Principles and Practices, 4th Edition, Pearson Education, 2005.
- [9] Xilinx Inc., Vivado Design Suite User Guide, Xilinx Documentation UG910, 2024.
- [10] Xilinx Inc., ISE Design Suite User Guide, Version 14.5, Xilinx Inc., 2013.
- [11] Peter J. Ashenden, The Designer's Guide to Verilog HDL, Morgan Kaufmann Publishers, 2007.
- [12] Wayne Wolf, FPGA-Based System Design, Prentice Hall, 2004.
- [13] Sunggu Lee and Kwan H. Lee, Advanced Digital Logic Design Using Verilog HDL, Cengage Learning, 2011.
- [14] Michael D. Ciletti, Advanced Digital Design with Verilog HDL, Pearson Education, 2011.
- [15] Richard F. Tinder, Engineering Digital Design, Academic Press, 2000.
- [16] Ronald J. Tocci, Neal S. Widmer and Gregory L. Moss, Digital Systems: Principles and Applications, 11th Edition, Pearson Education, 2011.
- [17] David Money Harris and Sarah L. Harris, Digital Design and Computer Architecture, 2nd Edition, Morgan Kaufmann Publishers, 2012.
- [18] Pong P. Chu, FPGA Prototyping by Verilog Examples, Wiley Publications, 2008.
- [19] U. Meyer-Baese, Digital Signal Processing with Field Programmable Gate Arrays, Springer, 2014.
- [20] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Pearson Education, 2018.
- [21] IEEE Standards Association, IEEE Standard for Verilog Hardware Description Language, IEEE Std 1364-2005, 2006.
- [22] Zainalabedin Navabi, Verilog Digital System Design, McGraw-Hill Publications, 2006.
- [23] M. Rafiquzzaman, Fundamentals of Digital Logic and Microcomputer Design, Wiley Publications, 2005.
- [24] T. L. Floyd, Digital Fundamentals, 11th Edition, Pearson Education, 2015.
- [25] Xilinx Inc., 7 Series FPGA Data Sheet, Xilinx Corporation, 2023.
-