Design and Functional Verification of a Car Parking Management System Using Verilog

Achha Vamshi[1],Dr M. Shashidhar [2]
[1]M.Tech Student,[2]Professor, VLSI System Design , ECE department,
Vaagdevi College Of Engineering, Bollikunta, Warangal, Telangana - 506005.
[1]Email: vamshiachha73@gmail.com
[2]Email: sasi47004@gmail.com

***Abstract*—**Efficient parking management has become an important requirement for handling the growing number of vehicles in residential, commercial, and institutional environments. This work presents the development and simulation of a digital car parking control system using Verilog Hardware Description Language (HDL). The proposed design focuses on implementing a modular controller capable of monitoring vehicle entry and exit, maintaining an accurate occupancy count, and regulating access based on parking availability. The system is organized into independent functional modules, including vehicle detection logic, parking occupancy management, gate control, and parking status indication, enabling simplified design verification and future scalability. Access to the parking area is automatically permitted only when vacant spaces are available, whereas additional vehicles are restricted once the predefined parking capacity is reached. A comprehensive verification environment is created using Verilog testbenches to evaluate the controller under multiple operating conditions, including normal vehicle entry, vehicle exit, continuous traffic flow, parking-full operation, simultaneous sensor events, and system reset. Functional simulations performed in ModelSim validate the correctness of the control logic by analyzing occupancy count, gate control signals, parking status indicators, and state transitions under different test scenarios. The simulation results demonstrate stable and reliable controller operation with accurate parking occupancy management and deterministic FSM behavior. The proposed architecture provides a scalable digital framework that can be extended in future to support intelligent parking features such as multi-level parking control, vehicle authentication, automatic slot allocation, and smart parking guidance while maintaining a hardware-independent simulation environment.

***Keywords*—** *Car Parking Control System, Verilog HDL, Finite State Machine (FSM), Digital Controller, Functional Verification, ModelSim, Testbench, Parking Occupancy Management.*

I. INTRODUCTION

I. INTRODUCTION

The rapid increase in vehicle ownership has created a growing demand for efficient parking management systems in residential communities, commercial complexes, educational institutions, hospitals, and public transportation facilities. Conventional parking management methods, which rely on manual monitoring or basic counting mechanisms, often suffer from inaccurate vehicle tracking, inefficient space utilization, increased waiting time at entry points, and traffic congestion within parking facilities. These limitations reduce operational efficiency and negatively affect the overall user experience, highlighting the need for automated digital parking control systems.

Several parking management approaches have been proposed to improve parking efficiency. Sensor-based systems employ infrared, ultrasonic, or camera-based technologies to detect vehicle presence and estimate parking occupancy, while IoT-enabled solutions provide real-time parking information through cloud platforms and mobile applications. Machine learning and image processing techniques have also been investigated for vehicle identification and parking space monitoring. Although these approaches offer advanced functionality, they generally require additional hardware components, communication infrastructure, and complex software integration, resulting in increased implementation cost and system complexity. Furthermore, many existing studies focus primarily on hardware realization rather than the design and verification of the underlying digital control logic responsible for reliable parking operations.

Digital hardware design using Hardware Description Languages (HDLs) provides an efficient methodology for developing reliable parking controllers before hardware implementation. Verilog HDL enables the creation of modular, synthesizable, and reusable digital circuits that can be functionally verified under different operating conditions using simulation tools. Simulation-based verification allows design errors to be detected and corrected at an early development stage, reducing implementation cost while improving design reliability and functional correctness.

This paper presents the development and simulation of a Car Parking Control System using Verilog Hardware Description Language (HDL). The proposed system employs a modular architecture consisting of vehicle detection, occupancy counting, gate control, parking availability indication, and a

Finite State Machine (FSM)-based controller to coordinate parking operations. Vehicle entry is permitted only when vacant parking spaces are available, whereas additional vehicles are restricted after the parking facility reaches its predefined capacity. The complete controller is modeled in Verilog HDL and verified using comprehensive testbenches under multiple operating scenarios, including vehicle entry, vehicle exit, parking-full conditions, simultaneous sensor activation, and system reset. Functional simulation results obtained using ModelSim demonstrate the correctness of the proposed control logic and confirm reliable parking occupancy management under different operating conditions. The developed architecture provides a scalable digital framework that can be extended to support advanced parking functionalities, including intelligent slot allocation, multi-level parking management, vehicle authentication, and smart parking guidance.

Simulation-driven verification plays a significant role in the development of digital hardware systems. Instead of directly implementing the design on physical hardware, simulation enables designers to evaluate functional correctness, identify design flaws, and validate system performance during the early stages of development. Comprehensive testbenches can reproduce various parking scenarios, including normal operation, boundary conditions, invalid input combinations, and parking capacity limits. This methodology minimizes development effort, reduces implementation cost, and increases confidence in the correctness of the final design before hardware realization.

Contributions of this Work

- Using Verilog Hardware Description Language (HDL), a modular car parking control system is created, offering an effective digital framework for automated parking management via simulation.
- To provide deterministic and dependable system performance, a Finite State Machine (FSM)-based control architecture is built to coordinate vehicle entrance, vehicle exit, parking occupancy monitoring, gate control, and parking status notification.
- To improve design flexibility, maintainability, and scalability, the controller is divided into separate functional modules, such as vehicle detection, occupancy counting, gate control, and display management, using a modular design technique.
- To assess the controller under various operating conditions, such as vehicle entry, vehicle exit, parking-full conditions, simultaneous sensor activation, continuous parking operations, and system reset, a thorough verification environment is created using Verilog testbenches

.II. RELATED WORK

Research on automated parking management systems has progressed significantly over the past decade and can generally be classified into three major categories: sensor-based parking systems, IoT-enabled smart parking solutions, and digital controller-based parking management systems.et The first category focuses on sensor-based parking detection techniques. Several researchers have employed infrared (IR), ultrasonic, and magnetic sensors to detect vehicle presence at parking slots. These systems continuously monitor parking occupancy and generate signals indicating whether a parking space is occupied or vacant. Although sensor-based approaches provide accurate vehicle detection, they primarily concentrate on data acquisition and often require additional hardware components and dedicated sensing infrastructure, increasing both installation cost and maintenance requirements.

The second category includes Internet of Things (IoT)-based parking management systems. These solutions integrate sensor networks with wireless communication technologies to transmit parking occupancy information to cloud servers, mobile applications, or web-based platforms. Drivers can remotely monitor parking availability and, in some cases, reserve parking spaces before arriving at the destination. While IoT-based systems improve user convenience and enable real-time parking information, they rely heavily on continuous network connectivity, cloud infrastructure, and communication modules, resulting in increased system complexity, higher implementation cost, and additional security considerations.

In contrast to previous approaches, the proposed work focuses on the development and simulation of a modular Car Parking Control System using Verilog Hardware Description Language (HDL). The controller integrates vehicle detection logic, occupancy counting, gate control, parking availability indication, and FSM-based supervision within a unified digital architecture. A comprehensive verification environment is developed using Verilog testbenches to evaluate the controller under multiple operating scenarios, including normal vehicle entry, vehicle exit, parking-full conditions, simultaneous sensor activation, continuous traffic flow, and system reset operations. Functional verification using ModelSim confirms the correctness of the proposed control logic and demonstrates reliable parking occupancy management in a hardware-independent simulation environment..

Table I summarizes the characteristics of the major categories of existing parking management systems and compares them with the proposed digital controller.

Approach	Major Features	Limitations
Sensor-Based parking system	Detect vehicle occupancy using sensors	Higher maintenance requirements
Conventional digital parking controllers	Vehicle counting and gate control using digital logic	Limited verification scenarios
Proposed Verilog based parking system	FSM based control	Reduced scalability
Smart parking system	Realtime monitoring cloud connectivity	Internet dependent
Parking system	Easy way park	Required large area

III. PROPOSED SYSTEM ARCHITECTURE

A. Overview of the Proposed car parking control system

The proposed Car Parking Control System is developed using Verilog Hardware Description Language (HDL) and follows a modular digital architecture to automate parking operations through simulation. The complete system consists of independent functional modules responsible for vehicle detection, occupancy counting, gate control, parking status indication, and supervisory control. These modules communicate with one another through well-defined control and status signals to ensure reliable parking management under different operating conditions.

The system receives vehicle entry and exit requests through digital input signals representing the corresponding sensor outputs. Whenever an entry request is detected, the controller verifies the current parking occupancy using the parking counter module. If vacant parking spaces are available, the controller activates the gate control module, permitting vehicle entry while simultaneously incrementing the occupied parking count. Conversely, when an exit request is received, the controller decrements the occupancy count, updates the available parking spaces, and refreshes the parking status indication. When the parking facility reaches its predefined maximum capacity, the controller automatically disables further entry requests until at least one parking space becomes available.

A Finite State Machine (FSM) serves as the supervisory controller for coordinating all parking operations. The FSM manages the sequence of vehicle entry, vehicle exit, occupancy updates, gate activation, and parking status monitoring while preventing invalid state transitions and incorrect counting operations. This structured control methodology ensures deterministic system behavior and improves the reliability of the parking controller.

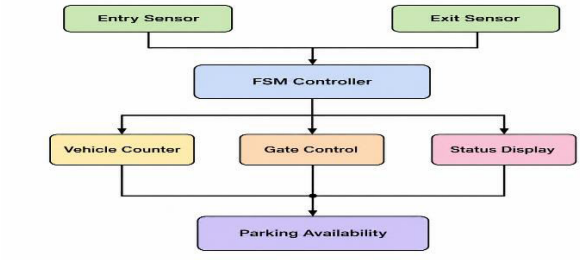


Fig. 1. Parking system control

The entire controller is implemented using a modular coding methodology in Verilog HDL, allowing each functional block to be independently designed, simulated, and verified before system integration. Functional verification is performed using comprehensive Verilog testbenches in ModelSim, where different operating scenarios—including normal vehicle entry, vehicle exit, parking-full operation, simultaneous sensor events, continuous vehicle flow, and system reset—are simulated. Waveform analysis confirms correct occupancy counting, accurate gate control, proper parking status indication, and reliable FSM state transitions, demonstrating the effectiveness of the proposed digital parking control architecture.

Describes the FSM operation

The Finite State Machine (FSM) serves as the central control unit of the proposed Car Parking Control System, coordinating all parking operations in a predefined sequence. The controller begins in the Idle state, where it continuously monitors the entry and exit sensor signals. Upon detecting a valid vehicle entry request, the FSM verifies the current parking occupancy before transitioning to the Entry state. If parking spaces are available, the controller activates the gate control signal, increments the occupancy counter, updates the parking availability status, and then returns to the Idle state. Similarly, when a vehicle exit is detected, the FSM enters the Exit state, decrements the occupancy count, refreshes the available parking information, and resumes normal monitoring. When the parking occupancy reaches the maximum permitted capacity, the FSM transitions to the Parking Full state, where additional entry requests are blocked until a vehicle exits the parking facility. A dedicated Reset state initializes all internal registers and counters, restoring the controller to its default operating condition whenever a reset signal is applied. By executing well-defined state transitions based on sensor inputs and parking occupancy, the FSM ensures deterministic system behavior, prevents invalid operations, and provides reliable coordination of all parking control functions.

C. Discusses the verification methodology

Under normal type-2 ac charging, the active rectifier of an on-board charger is controlled in the d-q reference frame to achieve unity-power-factor rectification at the grid terminals. During V2V charging, this control loop is disabled and the active rectifier is instead operated as a fixed switch network: once the two type-2 ports are connected, the gating signals for S1 and S6 of active rectifier-1, and for S1′ and S6′ of active rectifier-2, are held active-high for the complete duration of the transfer in every mode described above. Fig. 2 summarizes the control flow used to select and enter a V2V mode, starting from a read of the two battery voltages and the provider/receiver preference set by the users.

Model Sim simulation and Verilog testbenches are used to verify the proposed Car Parking Control System's functionality. To assess how the controller behaves in various operating scenarios, such as normal vehicle entry, vehicle exit, parking-full operation, continuous vehicle movement, simultaneous sensor activation, and system reset, an extensive set of test cases is created. To verify the accuracy of the occupancy counter, gate control logic, parking status indication, and Finite State Machine (FSM) transitions, the input signals are applied in a predetermined order during each simulation. To make sure the controller

generates the desired outputs for each input condition without logical conflicts or inappropriate state transitions, the generated simulation waveforms are evaluated.

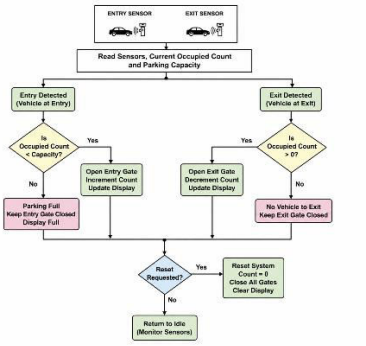


Fig. 2. Proposed system impllementation control flow.

D. Occupancy Counter Logic

he parking occupancy counter is responsible for maintaining an accurate record of the number of vehicles currently occupying the parking facility. The counter is continuously updated based on the entry and exit events generated by the corresponding sensor inputs. Whenever a valid vehicle entry is detected, the occupancy count is incremented by one, whereas a valid vehicle exit decrements the count by one. The updated parking occupancy is determined using

$$P_{new} = P_{current} + E - X$$

where $P_{current}$ represents the current number of occupied parking spaces, P_{new} denotes the updated parking occupancy, E is the entry event (assigned a value of 1 when a vehicle enters and 0 otherwise), and X is the exit event (assigned a value of 1 when a vehicle exits and 0 otherwise). To ensure reliable system operation, the occupancy count is constrained within the permissible parking capacity such that

$$0 \leq P_{new} \leq P_{max}$$

where P_{max} denotes the maximum parking capacity of the facility. This constraint prevents counter overflow and underflow conditions while ensuring that the parking occupancy information remains accurate throughout the parking operation.

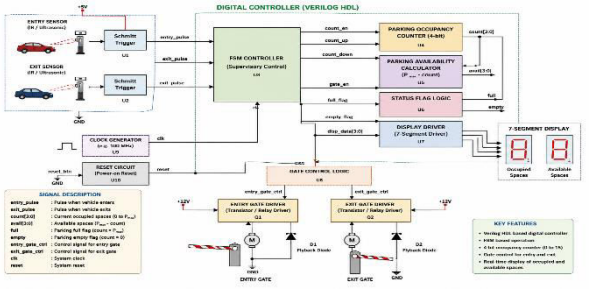


Fig. 3. Digital Controller (Verilog HDL)

the parking availability module continuously determines the number of vacant parking spaces by comparing the current parking occupancy with the maximum parking capacity. the available parking spaces are calculated asiv. simulation results and discussion.

$$A=P_{max}\ P_{current}$$

where Arepresents the number of available parking spaces, P_{max} denotes the maximum parking capacity, and $P_{current}$represents the current number of occupied parking spaces.The gate control decision is based on the parking availability. If at least one parking space is available, the entry gate is enabled; otherwise, the entry gate remains closed to prevent additional vehicles from entering.

Gate= {OPEN,CLOSE,A>0A=0.

E. vehicle entry control logic

The vehicle entry control module processes the signal generated by the entry sensor. Before permitting vehicle entry, the controller verifies that the parking occupancy has not reached the maximum allowable capacity. The entry gate control signal is expressed as

$$Gate_{Entry} = EntrySensor \times (P_{current} < P_{max})$$

where the vehicle entry request signal is represented by EntrySensor. The controller increases the occupancy counter and opens the gate if there is at least one empty parking spot in the parking complex. If not, the parking-full signal is generated and the vehicle entry request is denied.

f. vehicle exit control logic

Whenever a vehicle leaves the parking facility, the exit sensor generates a digital signal that is processed by the controller. The exit gate is activated according to $Gate_{Exit} = ExitSensor$

The parking occupancy is then updated as

$$P_{new} = \begin{cases} P_{current} - 1, & P_{current} > 0 \\ 0, & Otherwise \end{cases}$$

This logic prevents counter underflow while ensuring that the occupancy count always represents the correct number of parked vehicles.

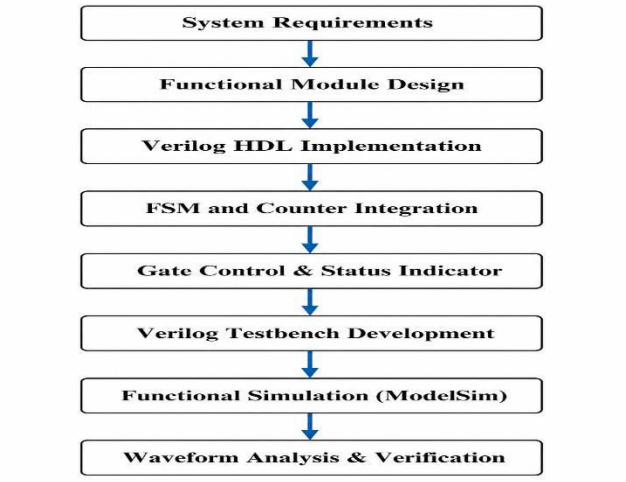
Finite State Machine (FSM) Operation

The proposed parking controller employs a Finite State Machine (FSM) to coordinate the complete parking operation. The controller consists of four primary states:

- S_0: Idle State
- S_1: Vehicle Entry State
- S_2: Vehicle Exit State S_3: Parking Full State The state transition function is defined as

METHODOLOGY

The development methodology adopted for the Car Parking Control System follows a structured digital design approach comprising system specification, module development, controller implementation, functional integration, and simulation-based verification. Each functional block is independently designed using Verilog Hardware Description Language (HDL) before being integrated into the complete parking controller. The overall design methodology is illustrated Digital input signals from the entry and exit sensors are supplied to the vehicle detection module. The Finite State Machine (FSM), which serves as the main controller for organizing all parking actions, processes these signals. The FSM creates the proper control signals for the occupancy counter, gate controller, and parking status indicator based on the sensor inputs and the current parking occupancy



IV. SIMULATION RESULTS AND DISCUSSION

A. Simulation Setup

The proposed Car Parking Control System is developed and functionally verified using Verilog Hardware Description Language (HDL). The complete design is modeled as a modular digital controller consisting of a Finite State Machine (FSM), vehicle detection logic, parking occupancy counter, gate control unit, and parking status indicator. Each module is individually verified before integrating the complete parking controller to ensure correct functional behavior. A comprehensive verification environment is developed using Verilog testbenches, where digital input signals are applied to emulate vehicle entry, vehicle exit, parking-full conditions, continuous vehicle movement, simultaneous sensor activation, and system reset operations. The controller continuously monitors the input signals and updates the parking occupancy count, available parking spaces, gate control signals, and parking status indicators according to the implemented control logic.

B. simulation Tool

The Car Parking Control System is modeled using Verilog Hardware Description Language (HDL) and functionally verified in the ModelSim simulation environment. The complete digital controller consists of a Finite State Machine (FSM), vehicle detection logic, occupancy counter, gate control unit, and parking status indicator, all of which are integrated to perform automated parking management. Functional verification is carried out using Verilog testbenches that emulate various parking scenarios, including vehicle entry, vehicle exit, parking-full conditions, continuous vehicle movement, simultaneous sensor activation, and system reset. Simulation waveforms are analyzed to evaluate the correctness of state transitions, parking occupancy updates, gate control signals, and parking availability indicators. The verification results confirm that the controller operates reliably under different operating conditions while maintaining accurate parking occupancy management.

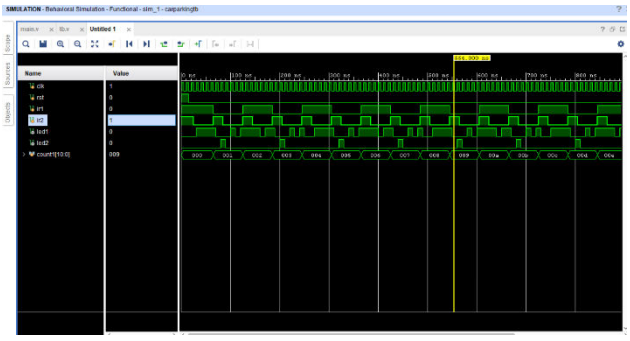


Fig. 4(a). result image

C. Vehicle entry operation

In this scenario, a vehicle arrival is represented by activating the entry sensor while parking spaces are available. The controller verifies the current parking occupancy before permitting vehicle access. Upon successful validation, the gate control signal is asserted, allowing the vehicle to enter the parking facility. Simultaneously, the occupancy counter is incremented by one, and the available parking spaces are updated accordingly. The FSM transitions from the **Idle** state to the **Entry** state and returns to the **Idle** state after completing the parking operation. The simulation waveform confirms correct gate activation, accurate occupancy counting, and proper state transitions without any invalid control signals.

Results

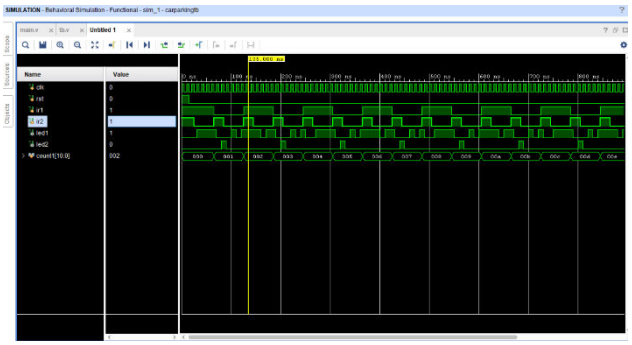


Fig 4(b). Simulation waveform

.

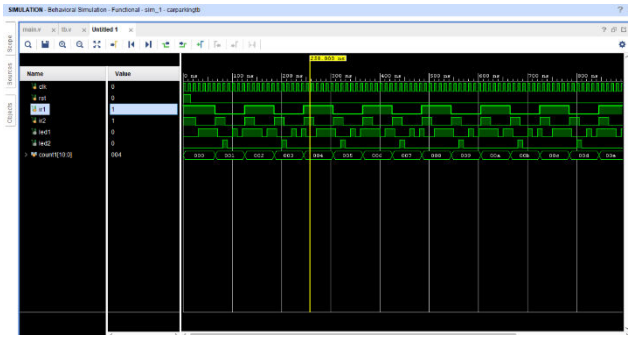


Fig. 4(c). verification waveform results.

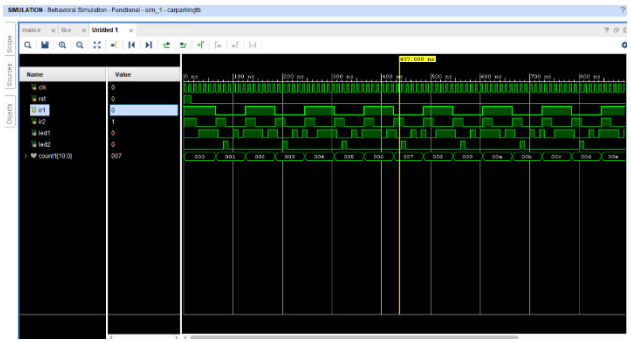


Fig $(d):vehicle entry exit simulation

E. Discussion

The simulation results demonstrate that the developed Car Parking Control System performs reliable parking occupancy management under all evaluated operating conditions. The modular architecture enables accurate coordination between the vehicle detection logic, occupancy counter, gate control unit, parking status indicator, and the Finite State Machine (FSM), ensuring synchronized operation throughout the parking process. The occupancy counter correctly updates the number of parked vehicles during both entry and exit operations while preventing counter overflow and underflow by restricting the count within the predefined parking capacity. Consequently, the parking availability information remains consistent with the actual parking occupancy during continuous vehicle movement. The FSM-based control methodology significantly improves the reliability of the digital controller by providing deterministic state transitions and well-defined operational sequences. The controller successfully manages vehicle entry, vehicle exit, parking-full conditions, and system reset operations without generating invalid control signals or incorrect occupancy updates. Simulation waveform analysis confirms that gate control signals are generated only under valid operating conditions, thereby preventing unauthorized vehicle entry when the parking facility reaches its maximum capacity.

V. CONCLUSION AND FUTURE SCOPE

The work presented in this paper demonstrates the successful design and functional verification of a digital Car Parking Control System using Verilog Hardware Description Language (HDL). A modular architecture incorporating vehicle detection logic, an occupancy counter, gate control, parking status indication, and a Finite State Machine (FSM) was developed to automate parking operations in a structured and reliable manner. Functional verification performed using ModelSim confirmed that the controller accurately manages vehicle entry and exit operations, maintains correct parking occupancy information, and enforces parking capacity constraints under various operating conditions. The simulation results further validate the effectiveness of the FSM in coordinating system activities while ensuring deterministic state transitions and eliminating invalid counting conditions. The modular implementation improves code organization, simplifies functional verification, and provides a flexible framework for future expansion without increasing design complexity. Overall, the developed controller offers a reliable and efficient digital solution for parking occupancy management and demonstrates the practicality of simulation-driven hardware design using Verilog HDL Future enhancements will focus on extending the controller with intelligent parking functionalities that improve automation and operational efficiency. The existing architecture can be expanded to support automatic parking slot allocation, multi-level parking management, secure vehicle authentication, and dynamic parking guidance for drivers. Additional research may also include integration with wireless communication and cloud-based monitoring systems to enable real-time parking information and centralized management. Furthermore, the controller can be synthesized and validated on FPGA platforms to evaluate its hardware performance, resource utilization, and timing characteristics. These enhancements will enable the developed digital controller to evolve into a comprehensive smart parking management solution suitable for modern transportation infrastructures.

REFERENCES

[1] M. M. Mano and M. D. Ciletti, *Digital Design*, 6th ed. Pearson, 2018.

[2] S. Brown and Z. Vranesic, *Fundamentals of Digital Logic with Verilog Design*, 3rd ed. McGraw-Hill Education, 2014.

[3] D. M. Harris and S. L. Harris, *Digital Design and Computer Architecture*, 2nd ed. Morgan Kaufmann, 2013.

[4] S. Palnitkar, *Verilog HDL: A Guide to Digital Design and Synthesis*, 2nd ed. Pearson Education, 2003.

[5] C. H. Roth Jr. and L. L. Kinney, *Fundamentals of Logic Design*, 7th ed. Cengage Learning, 2014.

[6] P. J. Ashenden, *The Designer's Guide to Verilog HDL*, 3rd ed. Morgan Kaufmann, 2008.

[7] N. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, 4th ed. Addison-Wesley, 2011.

[8] M. Morris Mano, *Digital Logic and Computer Design*, Pearson Education, 2004.

[9] M. Ali, S. Hossain, and M. Rahman, "Design and implementation of an FPGA-based smart parking system," *International Journal of Computer Applications*, vol. 179, no. 20, pp. 24–30, 2018.

[10] R. Yusnita, F. Norbaya, and N. Basharuddin, "Intelligent parking space detection system based on image processing," *International Journal of Innovation, Management and Technology*, vol. 3, no. 3, pp. 232–235, 2012.

[11] S. Revathi and T. Dhivya, "Smart parking management system using IoT," *International Journal of Engineering Research & Technology (IJERT)*, vol. 8, no. 5, pp. 356–360, 2019.

[12] M. Idris, Y. Leng, E. Tamil, N. Noor, and Z. Razak, "Car park system: A review of smart parking system and its technology," *Information Technology Journal*, vol. 8, no. 2, pp. 101–113, 2009.

[13] S. Khanna and A. Anand, "IoT-based smart parking system," in *Proc. International Conference on Internet of Things and Applications (IOTA)*, Pune, India, 2016, pp. 266–270.

[14] A. R. Al-Ali, I. Zualkernan, and F. Aloul, "A mobile GPRS-sensors array for air pollution monitoring," *IEEE Sensors Journal*, vol. 10, no. 10, pp. 1666–1671, Oct. 2010.

[15] Xilinx Inc., *Vivado Design Suite User Guide: Synthesis (UG901)*, Version 2024.

[16] Siemens EDA, *ModelSim User's Manual*, Siemens Digital Industries Software, 2024.

[17] IEEE Standard for Verilog Hardware Description Language, **IEEE Std 1364-2005**, IEEE, New York, NY, USA, 2006.

[18] IEEE Standard for SystemVerilog—Unified Hardware Design, Specification, and Verification Language, **IEEE Std 1800-2023**, IEEE, 2023.

[19] Xilinx Inc., *Vivado Design Suite User Guide: Logic Simulation (UG900)*, Version 2024.

[20] Intel Corporation, *Quartus Prime Pro Edition User Guide: Design Compilation*, Version 24.1, 2024.