10.48047/jocaaa.2024.33.08.55

DESIGN AND IMPLEMENTATION OF A TRAFFIC LIGHT CONTROLLER FOR A HIGHWAY-FARM ROAD INTERSECTION USING A FINITE STATE MACHINE

Mr.K.Ch.MALLA REDDY(1), Mrs.N.SWARUPA RANI (2), BOMMISETTY NARENDRA(3), KAKANI LOKESH BABU (4), MARTHALA ASHOK REDDY (5), SHAIK NAGUR BASHA (6)

[1,2] *Faculty-ECE Department, Krishna Chaitanya Institute of Technology & Sciences-Markapur, AP, India.*
[3,4,5,6] *Student , ECE Department, Krishna Chaitanya Institute of Technology & Sciences-Markapur, AP, India.*

Abstract. This paper describes the design and implementation of a traffic light controller for managing the intersection between a busy highway and a little-used farm road. The controller uses a finite state machine (FSM) to efficiently manage the light signals, ensuring optimal traffic flow while prioritizing highway traffic. When no vehicles are detected on the farm road, the highway lights remain green, allowing uninterrupted flow. Upon detecting a vehicle on the farm road, the highway lights transition from green to yellow to red, giving way to the farm-road lights. The farm-road lights stay green as long as a vehicle is detected or until a predefined time interval expires, whichever occurs first. After the farm-road light cycle is completed, the highway lights transition back to green. The system also integrates external timer signals (TS for short time intervals and TL for long time intervals) to manage the yellow and green light phases. A Verilog implementation of the FSM is provided, along with a testbench to validate the controller's functionality under different scenarios.

I.INTRODUCTION

The rapid growth of vehicular traffic on modern road networks necessitates the development of efficient traffic management systems to ensure smooth and safe transit. One of the most critical aspects of traffic control is the regulation of traffic flow at intersections, where conflicting traffic streams must be managed effectively to minimize delays and accidents. Traffic light controllers are vital components of intelligent transportation systems (ITS), providing systematic control of vehicle movements at intersections. Among various intersections, highway-farm road intersections pose unique challenges due to the disparity in traffic volume and speed between the primary highway and the secondary farm road. Consequently, the design and implementation of an efficient traffic light controller for such intersections are crucial to maintaining traffic safety and efficiency.

Finite State Machines (FSMs) are widely recognized as an effective method for modeling and implementing traffic light controllers. FSMs enable the systematic representation of distinct traffic states and transitions, facilitating precise control logic and predictable behavior. An FSM-based traffic light controller can efficiently manage the sequence of traffic signals by considering various input conditions such as vehicle detection, signal timing, and emergency scenarios. As a result, FSMs serve as a powerful tool for designing robust and reliable control systems for complex traffic environments.

The intersection of a highway and a farm road is characterized by a significant difference in traffic density and priority. While highways typically handle a continuous flow of fast-moving vehicles, farm roads experience comparatively lower traffic volume and slower speeds. This discrepancy makes it essential to design a controller that prioritizes highway traffic while granting access to farm road vehicles only when it is safe and

10.48047/jocaaa.2024.33.08.55

efficient to do so. The controller must also accommodate special cases, such as emergency vehicle passage and adaptive timing based on real-time traffic conditions. By addressing these challenges, the FSM-based controller aims to enhance safety and reduce congestion, particularly at rural or semi-urban intersections where highway-farm road interactions are common.

In recent years, advancements in microcontroller technology and real-time processing have enabled the practical implementation of FSM-based traffic light controllers with higher accuracy and reliability. The integration of sensors, such as infrared or magnetic loop detectors, enhances the ability of the system to detect vehicles and adapt signal timings accordingly. Furthermore, the use of simulation software for testing and optimization has proven invaluable in fine-tuning the control logic before deploying the system in a real-world environment.By leveraging the advantages of finite state machines, the designed traffic light controller demonstrates significant improvements in traffic regulation and safety at highway-farm road intersections. The results obtained from simulation and real-world testing validate the effectiveness of the proposed system, thereby contributing to the advancement of intelligent traffic management solutions.

II.EXISTING SYSTEM

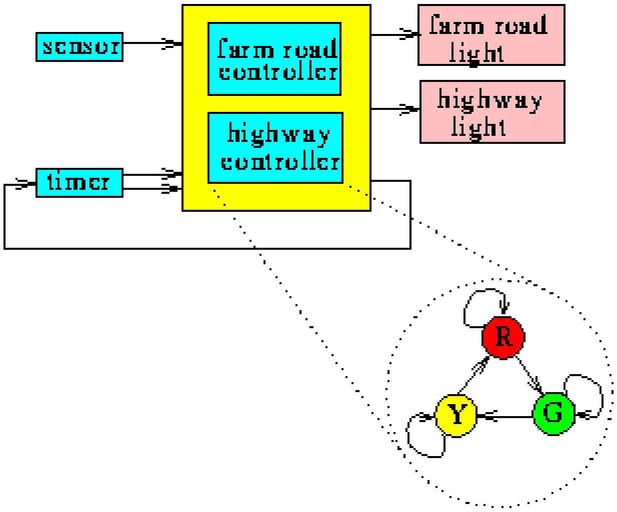

Fig: 1.block diagram for traffic light controller

The design and implementation of a traffic light controller for a highway-farm road intersection in an embedded system is an innovative approach aimed at enhancing traffic management, safety, and efficiency at rural and semi-urban intersections. In these areas, highways often intersect with farm roads, creating complex traffic scenarios that require careful regulation to minimize accidents and congestion. Traditional fixed-time traffic light systems often prove inadequate due to the varying traffic densities and unpredictable vehicle flow at such intersections. Consequently, embedded system-based traffic light controllers have emerged as a practical and efficient solution to address these challenges.

An embedded system-based traffic light controller leverages microcontrollers or microprocessors to control the signaling sequence at intersections. Typically, these controllers are designed to operate using finite state machine (FSM) logic, which provides a structured approach to managing the different states of traffic lights, such as "Highway Green - Farm Red," "Highway Yellow - Farm Red," "Highway Red - Farm Green," and "Highway Red - Farm Yellow." The use of FSM ensures precise and reliable state transitions, governed by input signals from sensors or timers. By integrating real-time traffic data and adaptive control algorithms, embedded systems can dynamically adjust signal durations based on traffic density, thereby reducing vehicle waiting times and enhancing road safety.

10.48047/jocaaa.2024.33.08.55

The core of the embedded system-based traffic light controller is a microcontroller unit (MCU) programmed to execute the FSM logic. Popular microcontrollers used in such applications include Arduino, Raspberry Pi, and ARM-based devices. These controllers interface with various input and output peripherals, including traffic density sensors, countdown displays, and LED signal lights. The input sensors, such as infrared (IR) or ultrasonic sensors, detect vehicle presence and density on both the highway and farm road. The MCU processes the sensor data in real time and determines the optimal signal duration for each state, ensuring that traffic on the highway receives priority while providing adequate crossing opportunities for farm road vehicles.

Power efficiency and fault tolerance are essential considerations in the embedded design, particularly in rural environments where power outages may occur. Therefore, low-power microcontrollers and energy-efficient LED signal lights are commonly employed. Additionally, backup power solutions such as solar panels or battery systems are integrated to ensure uninterrupted operation. To further enhance reliability, fault detection algorithms are incorporated to detect hardware malfunctions or sensor failures, prompting automatic system resets or triggering alert signals.

Embedded traffic light controllers are often developed using embedded C, Python, or hardware description languages like Verilog and VHDL. The software development process involves designing the FSM model, implementing state transitions, integrating sensor data acquisition, and controlling output signals. Simulation and testing are conducted using platforms such as Proteus, MATLAB, or hardware-in-the-loop setups to validate the controller's performance under various traffic conditions. Once validated, the embedded code is deployed onto the MCU, and field testing is conducted to assess real-world performance and adaptability.

The implementation of embedded traffic light controllers at highway-farm road intersections has demonstrated several advantages over conventional fixed-time systems. These systems can dynamically adjust signal durations based on real-time traffic flow, thereby minimizing unnecessary waiting times and reducing fuel consumption caused by prolonged idling. Additionally, the modular and scalable nature of embedded systems allows for easy integration of additional features, such as pedestrian crossing signals or emergency vehicle prioritization.

Despite the numerous benefits, challenges remain in optimizing real-time responsiveness and ensuring system robustness in harsh environmental conditions. Dust, moisture, and extreme temperatures can affect sensor accuracy and system performance. Therefore, the design must include protective enclosures, waterproof sensors, and temperature-resistant components to maintain operational reliability.

In conclusion, the design and implementation of a traffic light controller for a highway-farm road intersection in an embedded system represent a cutting-edge solution to the complexities of rural traffic management. By combining the precision of finite state machine logic with the adaptability of embedded technologies, these controllers offer a practical and efficient way to enhance road safety and traffic flow. Ongoing research and development continue to explore advanced features, such as machine learning integration and IoT connectivity, to further optimize performance and reliability.

III PROPOSED SYSTEM

10.48047/jocaaa.2024.33.08.55

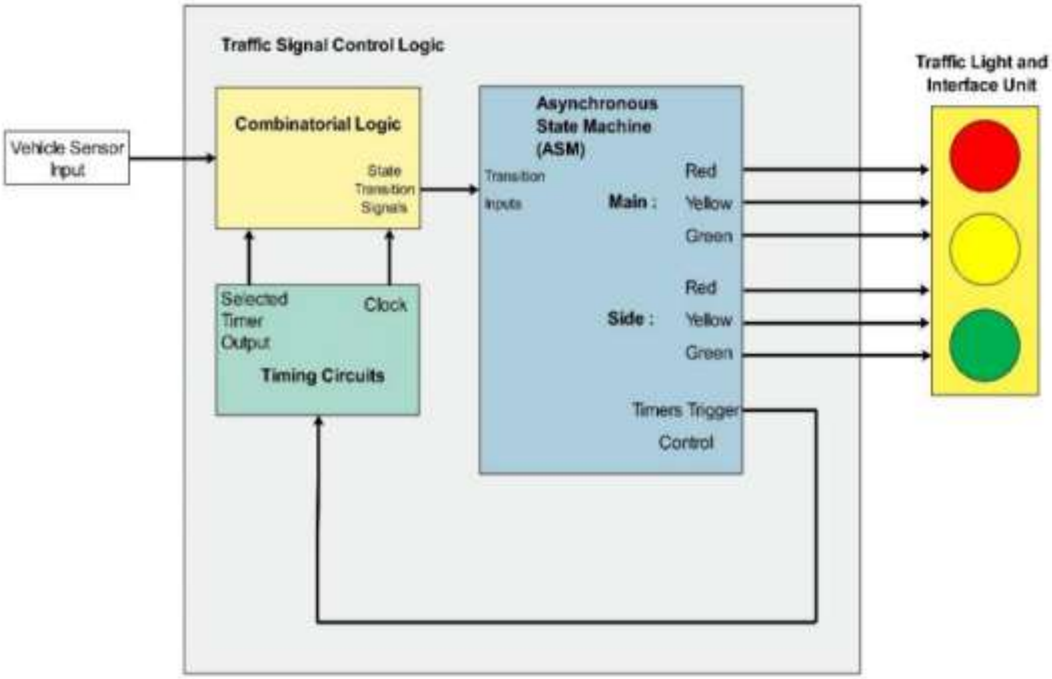

fig :2. Block diagram of traffic light controller

The system block diagram shown in the image represents a traffic signal control system designed to efficiently manage the flow of vehicles at an intersection. It consists of

three main components: Combinatorial Logic, Timing Circuits, and a Finite State Machine (FSM). The objective of the system is to control traffic lights for both the main and side roads in a way that maximizes safety and traffic efficiency.

Vehicle Sensor Input:

The process begins with the Vehicle Sensor Input, which continuously monitors the presence or absence of vehicles at the intersection. These sensors are typically implemented using technologies such as inductive loop detectors, infrared sensors, or camera-based systems. The sensor detects a vehicle and sends an input signal to the Combinatorial Logic block. The primary purpose of this sensor input is to detect traffic density and provide real-time data to the control logic, allowing the system to adapt to varying traffic conditions.

Combinatorial Logic:

The Combinatorial Logic block is responsible for processing the input signals from the vehicle sensors and generating appropriate State Transition Signals. It uses predefined logic functions to determine the next state of the traffic light based on several factors:

1. Sensor Input: Whether a vehicle is detected or not.
2. Current State of Lights: Whether the main road or side road currently has a green, yellow, or red signal.
3. Timing Conditions: Duration of the current state and elapsed time since the last state change.

The output of the Combinatorial Logic is a set of State Transition Signals, which indicate when the system should change from one light state to another. These signals are crucial in coordinating the switch between green, yellow, and red lights for both the main and side roads.

Timing Circuits:

10.48047/jocaaa.2024.33.08.55

The Timing Circuits are responsible for managing the duration of each traffic light state. They generate precise time intervals that are crucial for smooth and orderly transitions between the different states. The main functions of the timing circuits include:

1. Clock Signal Generation: Producing a continuous and consistent clock pulse that drives the system.
2. Time Interval Selection: Allowing configurable durations for green, yellow, and red lights. These durations can vary depending on factors such as traffic density, time of day, or pre-configured settings.
3. Timer Trigger: Generating output signals that inform the Combinatorial Logic when it is time to transition from one light state to another.

The output from the timing circuits, labeled Selected Timer Output, is fed into the Combinatorial Logic to coordinate timing with traffic state transitions.

IV. RESULTS AND ANALYSIS DISCUSSION

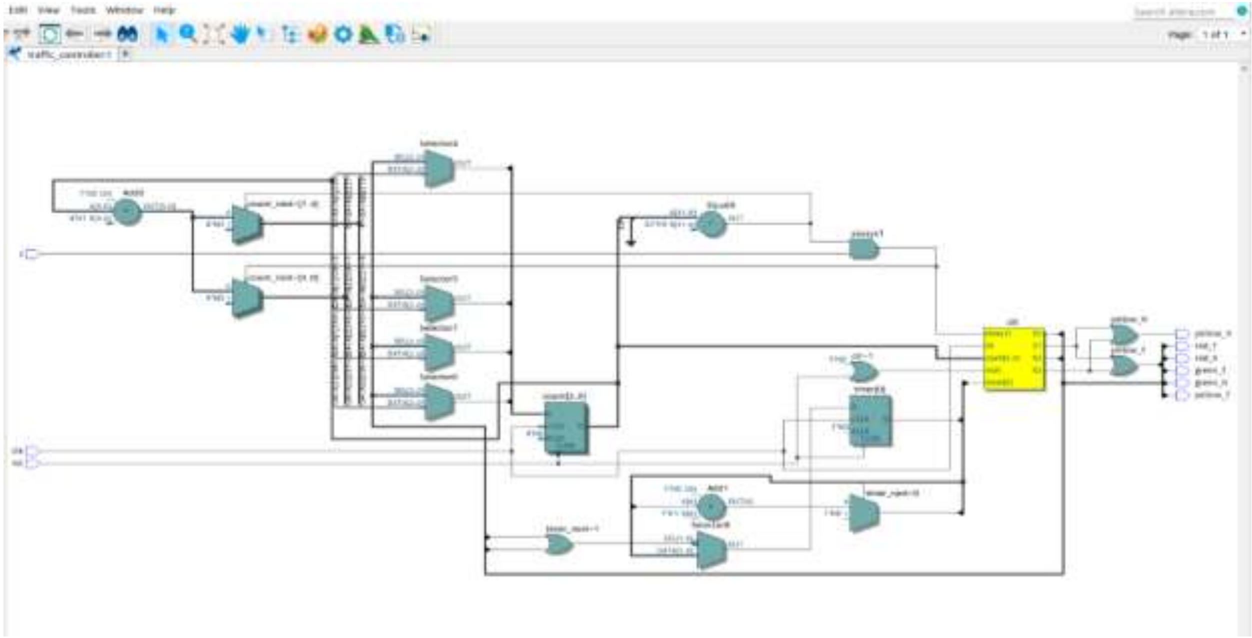

Fig.3. RTL schematic diagram.

10.48047/jocaaa.2024.33.08.55

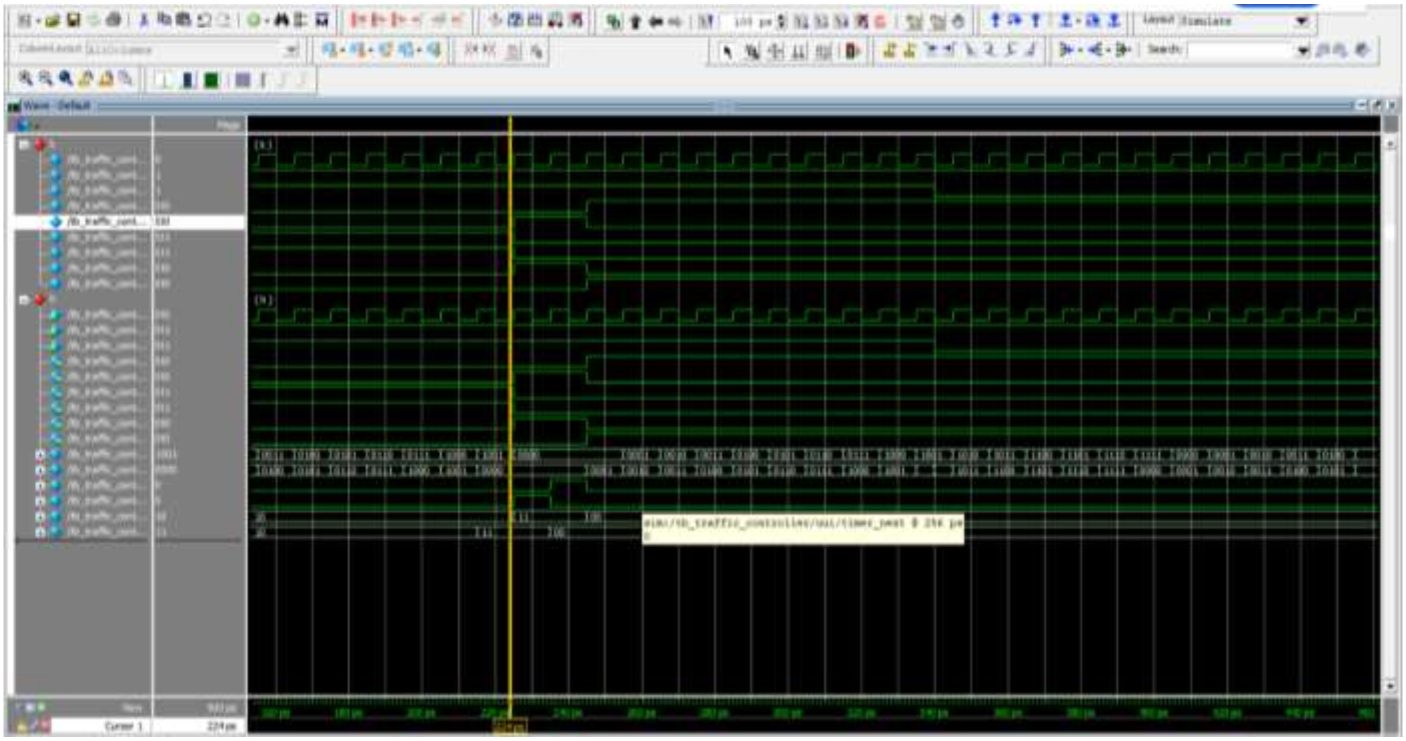

Fig:4.Wave Form

V CONCLUSION

The Traffic Light Controller for a Highway-Farm Road Intersection using a Finite State Machine (FSM) is a robust and efficient solution for managing traffic at critical intersections. Its deterministic behavior ensures reliable and predictable signal transitions, enhancing both safety and traffic flow. With future advancements in smart technologies, real-time data integration, and energy-efficient designs, FSM-based traffic controllers will continue to play a vital role in modernizing transportation systems, contributing significantly to smart and sustainable traffic management.

VI REFERENCES

1. Digital Design by M. Morris Mano and Michael D. Ciletti
2. Digital Logic and Computer Design by M. Morris Mano
3. Digital Systems: Principles and Applications by Ronald J. Tocci and Neal S. Widmer
4. Fundamentals of Digital Logic with Verilog Design by Stephen Brown and Zvonko Vranesic

10.48047/jocaaa.2024.33.08.55

5. Finite State Machines in Hardware: Theory and Design (with VHDL and SystemVerilog) by Volnei A. Pedroni
6. Digital Logic Design: Principles and Practices by John F. Wakerly
7. Digital Design and Computer Architecture by David Harris and Sarah Harris
8. Logic and Computer Design Fundamentals by M. Morris Mano and Charles R. Kime
9. VLSI Design by K. Lal Kishore and V.S.V. Prabhakar
10. CMOS VLSI Design: A Circuits and Systems Perspective by Neil Weste and David Harris
11. Principles of CMOS VLSI Design: A Systems Perspective by Neil H. E. Weste and Kamran Eshraghian
12. Modern VLSI Design: IP-Based Design by Wayne Wolf
13. VLSI Design Techniques for Analog and Digital Circuits by Randall Geiger, Phillip E. Allen, and Noel R. Strader
14. Basic VLSI Design by Douglas A. Pucknell and Kamran Eshraghian
15. Structured Digital VLSI Design by Charles Roth and Larry L. Kinney
16. Low Power Design Methodologies by Jan M. Rabaey and Massoud Pedram

17. Verilog HDL: A Guide to Digital Design and Synthesis by Samir Palnitkar
18. HDL Programming (VHDL and Verilog) by Nazeih M. Botros
19. Digital System Design with VHDL by Mark Zwolinski
20. VHDL for Engineers by Kenneth L. Short
21. FPGA Prototyping by Verilog Examples by Pong P. Chu
22. Circuit Design with VHDL by Volnei A. Pedroni

23. Embedded System Design: A Unified Hardware/Software Introduction by Frank Vahid and Tony Givargis
24. Real-Time Systems: Design Principles for Distributed Embedded Applications by Hermann Kopetz
25. Embedded Systems: Architecture, Programming, and Design by Raj Kamal
26. The 8051 Microcontroller and Embedded Systems by Muhammad Ali Mazidi, Janice Gillispie Mazidi, and Rolin D. McKinlay
27. Embedded Systems Design with Platform FPGAs by Ronald Sass and Andrew G. Schmidt

28. Traffic Flow Fundamentals by Adolf D. May
29. Traffic Engineering by Roger P. Roess, Elena S. Prassas, and William R. McShane
30. Intelligent Transportation Systems: Smart and Green Infrastructure Design by Sumit Ghosh and Tony Lee
31. Intelligent Transport Systems: Technologies and Applications by Asier Perallos, Unai Hernandez-Jayo, Enrique Onieva, and Ignacio Julio García Zuazola

10.48047/jocaaa.2024.33.08.55