A PROJECT REPORT
ON

TRAFFIC LIGHT CONTROLLER USING VERILOG

By: Dhananjay Singh

ABSTRACT

Transformer is a static device, which converts power from one level to another level. The aim of the proposed work is to protect the transformer under overload condition by load sharing. Due to over load on transformer, the efficiency drops and winding get overheated and may get burnt. Thus, by sharing load on transformer, the transformer is protected. This will be done by connecting another transformer in parallel through a microcontroller on the other hand through the Arduino. The both controllers compare the load on the first transformer with the reference value. When the load exceeds the reference value, the second transformer will share the remaining load. If the load exceeds the rating of both transformers, then system is going to be shut down. Whenever the sharing of load on transformer occurs, the operator gets message through the GSM. An IOT is also used to inform the control station about sharing load. This arrangement will provide proper maintenance facility for both transformers. Hence, temperature of both transformers, load shared to another transformer along with timing are recorded.

FIGURE:

Contents

CHAPTER 1.

INTRODUCTION

Introduction of Project

Traffic control is a challenging problem in many cities. This is due to the large number of vehicles and the high dynamics of the traffic system. Poor traffic systems are the big reason for accidents, time losses. In this method of approach, it will reduce the waiting time of the vehicles at traffic signals. The hardware design has been developed using Verilog Hardware Description Language (HDL) programming.

Verilog designing is hardware descriptive language, the name itself suggest that it deals with the hardware designing and simulation. Basically, it becomes very difficult to mount the various electronic components on breadboard or PCB circuit. It also takes too much time for the simulation and sometimes many errors occur because of improper connection of components onto the circuit. And thus, to overcome this factor hardware descriptive language comes into conclusion. we can code the process using Verilog and we can mount it on a circuit or just upload it to the circuit accordingly so that particular circuit will work as according to the code we have written.

HDL language is often used for sequential circuits like shift register, combinational logic circuit like adder, subtractor etc. Basically it describes the digital systems like microprocessor or a memory. Whatever design that is described in HDL are independent, it has its unique state of work, very much easy to simulate, designing and debugging, and very useful than schematics, especially for large circuits thus, to overcome difficulties or problems to design the circuits manually with breadboard and PCB, use of Verilog designing in this complex world is increasing a way better.

This project deals with a basic design of a T - Shaped road for traffic light control. The output of system has been tested using Xilinx 2023.2

A traffic light system is an electronic device that assigns right of way at an intersection or crossing or street crossing by means of displaying the standard red, yellow and green colored indications. In addition, it also works in conjunction with pedestrian displays to assign pedestrian crossing right of way.

A traffic light, also known as traffic signal, stop light, stop-and-go lights, is a signaling device positioned at a road intersection, pedestrian crossing, or other location in order to indicate when it is safe to drive, ride, or walk using a universal colour code. Nowadays,

1. A red light meant traffic in all directions had to stop.
2. A yellow light meant cross-town traffic would have to slow and,
3. A green light would to go or proceed.

The problem of heavy jam is happened because of never configure the level of jam in each way and set the delay time. Another problem represents when there is no jam, but the waiting still continues. The solution for these problems is to determine the level of jam and set the delay time. This problem need of evaluation of the traffic policeman, and then there is need for

manual control of the traffic. The target of this paper is to propose system provide solution for all above problems with least possible cost. Traffic light controller (TLC) can be implemented using Verilog, FPGA, and ASIC design. FPGA has many advantages over microcontroller, some of these advantages are; the speed, number of input/output ports and performance which are all very important in TLC design, at the same time ASIC design is more expensive than FPGA. Nowadays, FPGA becomes one of the most successful of today's technologies for developing the systems which require a real time operation. FPGA is a re-configurable integrated circuit that consists of two dimensional arrays of logic blocks and flip-flops with an electrically programmable interconnection between logic blocks. The reconfiguration property enables fast prototyping and updates for hardware devices even after market launch. Most of the TLCs implemented on FPGA are simple ones that have been implemented as examples of Finite State Machine (FSM).

The Verilog language has been selected for programming the FPGA to fill two important needs in the design process.

- Firstly, it gives full description of the structure of a design that is how it is decomposed into sub-designs, and how those sub-designs are interconnected.
- Secondly, it allows simulating the design before starting the manufacturing.
- Accordingly, the designers can quickly compare alternatives and test for correctness without the delay and expense of hardware prototyping.

BENEFITS OF USING VERILOG HDL (HARDWARE DESCRIPTION LANGUAGE) :

Verilog is a widely used Hardware Description Language (HDL) for designing digital circuits. It can also be used for modeling analog circuits. Verilog is a descriptive language that describes a relationship between signals in a circuit.

A Verilog model describes a unit of digital hardware in terms of :

- Interconnections of other hardware unit whose models prescribe their behavior in a simulation.
- Behavioral / procedural algorithms that abstractly describe input/output behavior that could be personified in a hardware unit.

Hardware description language (HDL) is divided by two types, Verilog and VHDL (VHSIC – Very High Speed Integrated Circuit Hardware Description Language). Both have its advantages and its disadvantage. In this project, Verilog HDL was chosen because it's used for synthesis of logic circuits (synthesizable code), used for verification purposes of a circuit (can be analog or digital or mixed signal), can be used by combining synthesis & verification (synthesizable & behavioral code) and it used for netlist representation of a synthesizable circuit (structural code).

THE ADVANTAGES USING VERILOG HDL ARE SHOWN BELOW :

- Easy to write.
- Easy to understand as it similar to C program.
- Easier to learn compared with VHDL.

Aim

Design a Verilog model for Traffic Signal Control using State Machine Diagram.

Objectives

This project focusses on the design of traffic light controller using Verilog as an open source platform for all end users . The objective of the project are as follows :

Design a Digital Controller to control traffic at an intersection of busy highway road and occasionally used country side road .

CHAPTER 2.

ABOUT TRAFFIC LIGHT

History of Traffic light

The traffic light has a rich history dating back to the 19th century. The first traffic light was installed in London in 1868, designed by railway manager J.P. Knight. It featured gas lamps and operated manually by a police officer. However, this early version was short-lived and prone to malfunctions.

In 1912, a more advanced electric traffic light was introduced in Salt Lake City, Utah, created by Lester Wire. It had red and green lights and a buzzer to signal the impending change. The addition of the amber light came later to provide a transition period between red and green.

The concept gradually evolved with improvements like automatic timers and standardized color sequences. In 1920, William Potts developed the first four-way, three-color traffic signal. Over the years, innovations such as pedestrian signals, signal synchronization, and the adoption of standardized international colors contributed to the modern traffic light system we see today.

Advancements continue with the integration of technology for traffic management, such as sensors and smart signals. The traffic light's history reflects a fascinating journey from simple gas lamps to sophisticated, interconnected systems that play a crucial role in regulating traffic flow worldwide.

What is Traffic Light

A traffic light is a signaling device positioned at intersections, pedestrian crossings, and other points on roadways to regulate vehicular and pedestrian traffic. It plays a crucial role in maintaining order and safety on the roads by controlling the flow of vehicles and pedestrians. This system typically consists of three colored lights: red, yellow (or amber), and green.

The red light indicates that vehicles and pedestrians must stop. It serves as a clear signal for everyone to come to a complete halt, allowing traffic from the perpendicular direction to proceed safely. This phase ensures orderly progression and minimizes the risk of collisions.

The green light signals the right of way for vehicles and pedestrians. When illuminated, it grants permission to proceed through the intersection or cross the road. This phase is crucial for maintaining traffic flow and preventing congestion.

The yellow or amber light acts as a transition between the red and green phases. It warns that the signal is about to change, prompting drivers to prepare to stop or proceed depending on the upcoming signal. This phase helps in smoothly transitioning between traffic movements, reducing abrupt stops and starts.

Traffic lights are often controlled by sophisticated systems that take into account various factors such as traffic volume, time of day, and pedestrian activity. Modern traffic management systems may use sensors, cameras, or other technologies to dynamically adjust signal timings based on real-time conditions. This adaptive approach enhances efficiency and reduces congestion.

The history of traffic lights dates back to the late 19th century. The first electric traffic light was installed in Cleveland, Ohio, in 1914, and it featured red and green lights but lacked the amber phase. Over the years, the design and functionality of traffic lights have evolved, incorporating innovations to improve visibility, efficiency, and safety.

In addition to the standard three-light configuration, some traffic lights include additional signals such as left-turn arrows or pedestrian walk signals. These features contribute to the overall complexity of traffic signal systems, enabling them to cater to diverse traffic scenarios.

Traffic lights are a crucial component of urban planning and transportation management. They contribute to the overall safety of road users, facilitate efficient traffic flow, and help reduce accidents. Proper adherence to traffic signals by drivers and pedestrians is essential for the effective functioning of these systems.

In conclusion, the traffic light is a fundamental element of modern transportation infrastructure, playing a pivotal role in regulating traffic and ensuring the safety of all road users. Its evolution from simple stop-and-go signals to sophisticated, adaptive systems reflects the ongoing efforts to enhance efficiency and address the challenges of urban mobility.

Figure 2-1 Traffic light

Red : Prohibits any traffic from proceeding.

Yellow : Provides warning that the warning signal is changing from green to red.

Green : Allows Traffic to proceed in the direction denoted.

CHAPTER 3.

VERILOG

VERILOG DEFITION

Verilog is a hardware description language (HDL) used to model and simulate electronic systems. It's widely employed in digital circuit design, particularly for field-programmable gate arrays (FPGAs) and application-specific integrated circuits (ASICs). Verilog allows designers to describe the behavior and structure of electronic systems.

KEY FEATURES OF VERILOG

Modules: Designs are structured using modules, which encapsulate specific functionalities. Modules can be instantiated within other modules, fostering a hierarchical design approach.

Behavioral and Structural Description: Verilog supports both behavioral and structural modeling. Behavioral descriptions focus on functionality, while structural descriptions emphasize the physical structure and connections between components.

Data Types: Verilog provides various data types, including scalar types (bit, logic), vector types (reg, wire), and composite types (arrays, structures). This versatility enables designers to represent a wide range of digital logic.

Concurrent Execution: Verilog is inherently concurrent, meaning multiple statements can execute simultaneously, reflecting the parallel nature of hardware. This is in contrast to sequential programming languages.

Simulation and Synthesis: Verilog is used in simulation environments to verify and debug designs. Additionally, it's crucial in synthesis tools that convert high-level Verilog code into a netlist of logic gates suitable for implementation on hardware.

Testbenches: Verilog testbenches are used to verify the functionality of designs. They include stimuli generation, applying inputs to the design, and monitoring the outputs for correctness.

Procedural Blocks: Verilog uses procedural blocks like always and initial to describe the execution order of statements within a module. These blocks help model sequential behavior.

Hierarchy: Designs are organized hierarchically, allowing for a modular and scalable approach. This makes it easier to manage and understand complex systems.

Timing and Delays: Verilog enables designers to model timing constraints and delays, crucial for ensuring that the design meets performance requirements.

FSM (Finite State Machine): Verilog is well-suited for modeling finite state machines, which are essential in many digital systems for controlling sequential logic.

Verilog is a powerful tool in digital design, supporting the entire design process from conceptualization and simulation to synthesis and implementation on hardware. It has become a standard in the electronics industry for developing digital circuits.

WORKING AND ITS METHODOLOGY

PROBLEM STATEMENT

Design a digital controller to control traffic at an intersection of a busy highway road and an occasionally used country side road .

The traffic signal for the main highway gets highest priority because cars are continuously present on the main highway . thus , the main highway signal remains green by default. occasionally, cars from the country road arrive added at the traffic signal . The traffic signal for the country road must term green only long enough to late the cars on the country road go.

As soon as there are no cars on the country road ,the country road traffic signal turns yellow and then red and the traffic signal on the main highway turns green again.

There is a sensor to detect cars waiting on the country road the sensor sends a signal X as input to the controller . X= 1 if there are cars on the country road , otherwise X =0.

There a delays on the transition from S1 to S2 , from S2 to S3, and from S4 to S0 . The delay must be controllable .
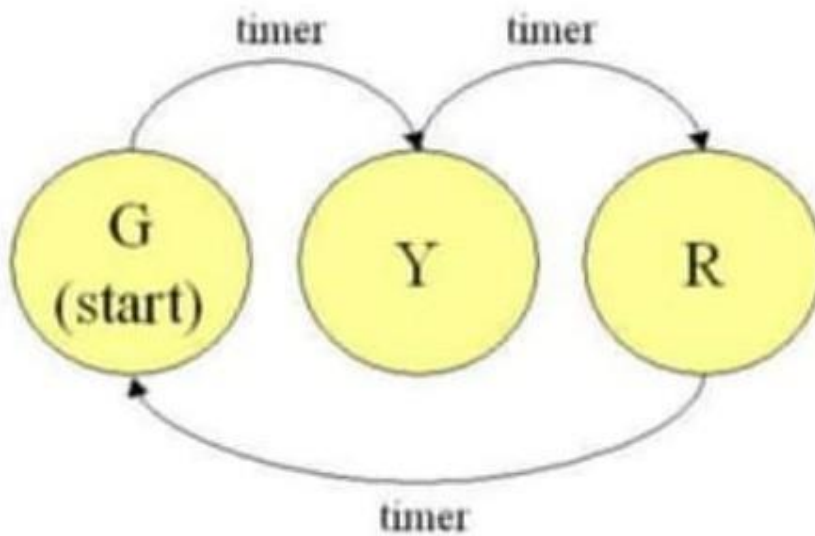

Figure 3-1 Cross road

Figure 3-2 State diagram

00 - assign for red light

01 – assign for yellow light

10 – assign for green light

STATE TABLE

State	Signals
S0	Hwy = G Cntry = R
S1	Hwy = Y Cntry = R
S2	Hwy = R Cntry = R
S3	Hwy = R Cntry = G
S4	Hwy = R Cntry = Y

Figure 3-3 State table

1. S0 is the finite state in which highway light is green and country side light is red .it is default state .

2. S1 is finite state after S0 state in which highway light is yellow and country side light is red .

3. S2 finite state ,come after S1 state in which highway light is red and country side light is also red .

4. S3 is the finite state which come after S2 state in which highway light is red and country side light is green .

5. S4 is the finite state which come after S3 state in which highway light is red and country side light is yellow .

STATE DIAGRAM

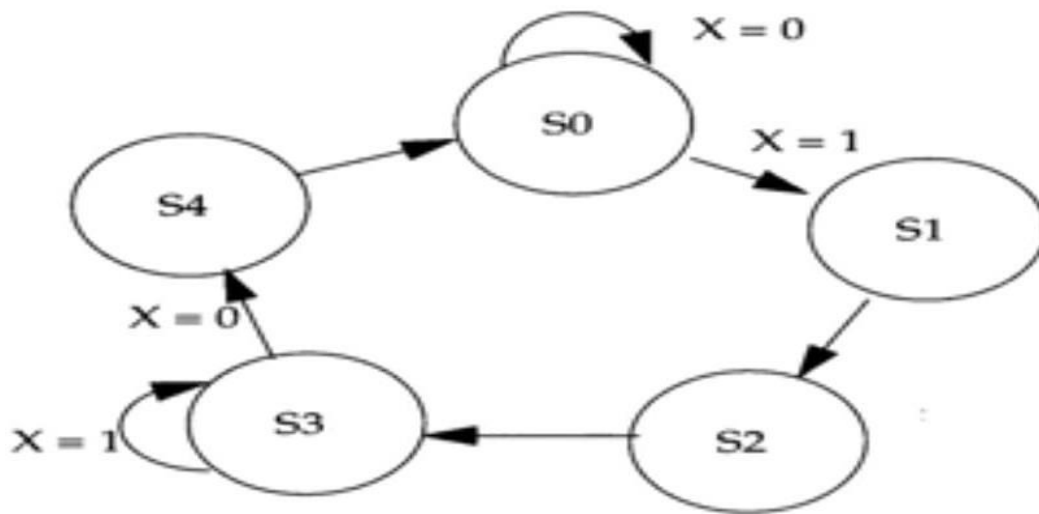Figure 3-4 state diagram

1) If sensor output X=0 , it means there is no cars present on country side road. So if machine present in S0 state remains same on that state and if present at S3 state then move to S4 state .

2) If sensor output X=1 , it means there is cars present on country side road. So if machine present in S0 state then move to S1 state and if present at S3 state then remain same at that state .

CHAPTER 4.

VERILOG CODE

CODING FOR FUNCTION

```
timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
// Design Name:
// Module Name: TLC
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////

`define true 1'b1
`define false 1'b0

`define y2rdelay 3
`define r2gdelay 2
module TLC(
    output [1:0] hwy,
    output [1:0] cntry,
    input clock,
```

```verilog
    input clear,
    input x
    );

reg [1:0] hwy,cntry;

parameter red = 2'd0,
      yellow = 2'd1,
       green = 2'd2;

parameter s0 = 3'd0,
     s1 = 3'd1,
     s2 = 3'd2,
     s3 = 3'd3,
     s4 = 3'd4;
reg [2:0] state;
reg [2:0] next_state;
always @ (posedge clock)
if(clear)
    state <= s0;
else
    state <= next_state;
always @(state)
begin
    hwy = green;
    cntry = red;
   case(state)
    s0:    ;
    s1: hwy = yellow;
    s2: hwy = red;
    s3: begin
        hwy = red;
      cntry = green;
       end
```

```verilog
            s4: begin
               hwy = red;
               cntry = yellow;
            end
         endcase
      end
   always @(state or x)
   begin
      case (state)
      s0: if(x)
            next_state = s1;
         else
            next_state = s0;
      s1: begin
            repeat(`y2rdelay) next_state = s1;
            next_state = s2;
         end
      s2: begin
            repeat(`r2gdelay) next_state = s2;

            next_state = s3;
         end
      s3: if(x)
            next_state = s3;
         else
            next_state = s4;
      s4: begin
            repeat(`y2rdelay) next_state ;
            next_state = s0;
         end
      default: next_state =s0;
      endcase
   end
endmodule
```

CHAPTER 5.

SIMULATION AND RESULT

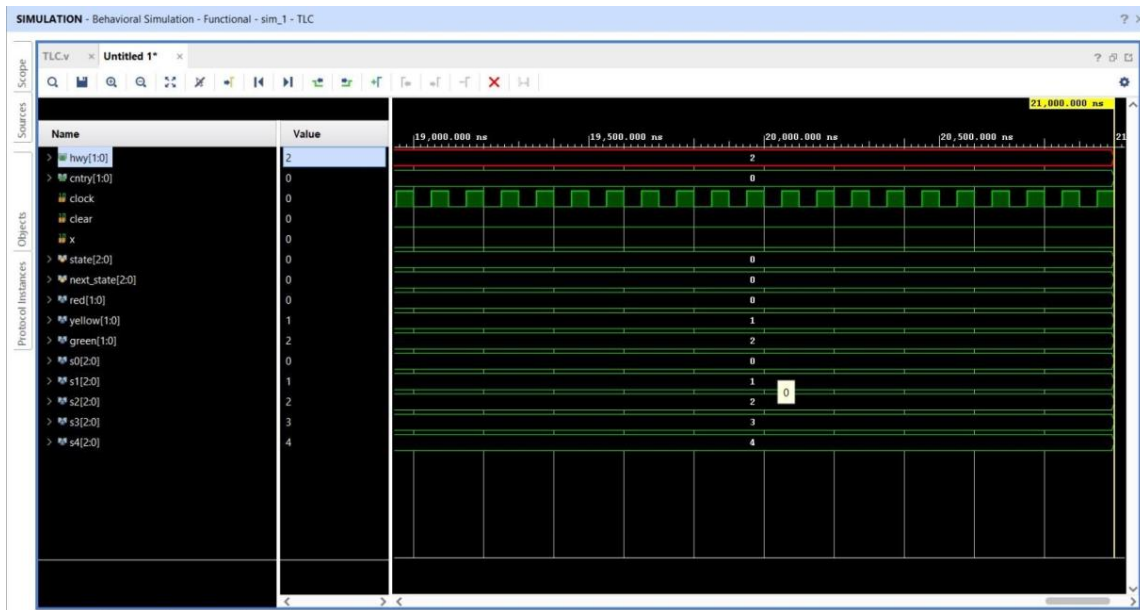CASE 1 : WHEN X=0 (no vehicle present in the country side)

Figure 5-1 Result 1
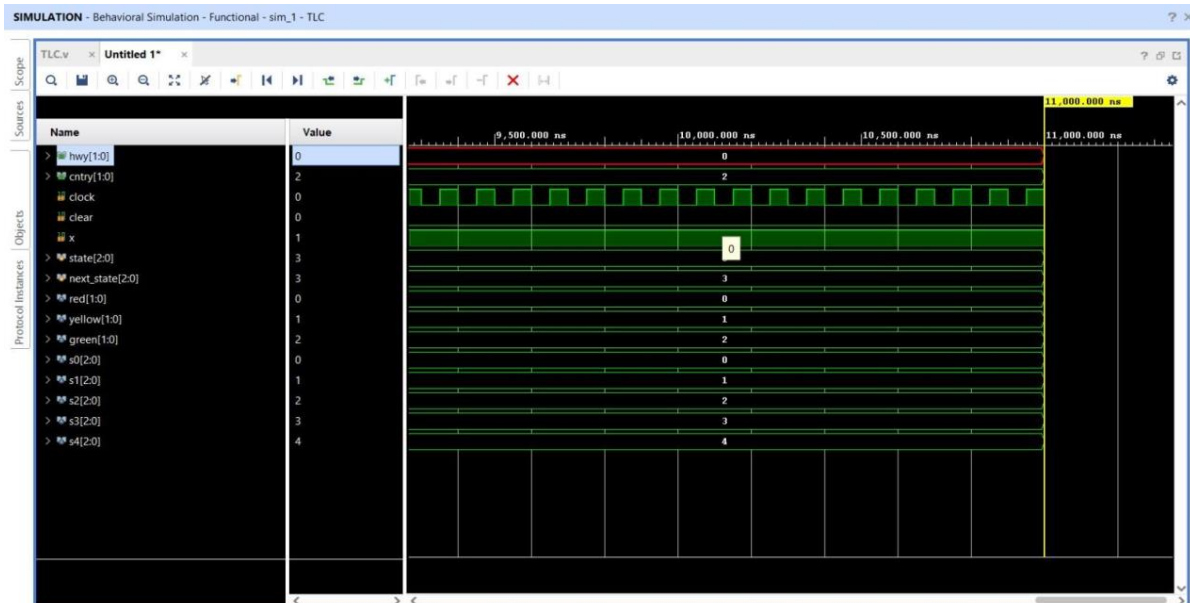
CASE 2 : WHEN X=1 (vehicle present in the country side)

Figure 5-2 Result 2

CHAPTER 6.

ADVANTAGE OF TRAFFIC LIGHT

I. Traffic Flow Control: They regulate the flow of vehicles, pedestrians, and cyclists at intersections, ensuring a smooth and organized movement of traffic.

II. Safety: Traffic lights enhance road safety by reducing the risk of collisions and accidents. They provide clear signals, minimizing confusion among road users.

III. Efficiency: Traffic lights optimize the use of intersection capacity, allowing for a more efficient allocation of green time to different directions based on traffic demand.

IV. Priority Allocation: They enable the fair allocation of right-of-way, ensuring that each direction gets its share of green time, preventing congestion and delays.

V. Pedestrian Safety: Traffic lights include pedestrian signals, allowing safe crossing for pedestrians by controlling the timing of the traffic flow.

VI. Reduced Traffic Jams: By controlling the flow of traffic, traffic lights help prevent gridlock and reduce congestion, leading to smoother and faster movement on the roads.

VII. Time Savings: Efficient traffic light systems can save time for motorists, pedestrians, and cyclists by minimizing waiting times at intersections.

VIII. Adaptability: Modern traffic lights can be equipped with sensors and adaptive systems that adjust signal timings based on real-time traffic conditions, optimizing efficiency.

IX. Emergency Vehicle Prioritization: Traffic lights can be programmed to prioritize emergency vehicles, enabling faster response times during emergencies.

X. Intersection Management: They facilitate the coordination of traffic at complex intersections, managing multiple streams of traffic to avoid conflicts and improve overall traffic management.

CHAPTER 7.

FUTURE WORK

This project can be enhanced in such away as to control automatically the signals depending on the traffic density on the roads using sensors like IR detector/receiver module extended with automatic turn off when no vehicles are running on any side of the road which helps in power consumption saving.

A lot of development ideas for work in future can be implemented, such as using solar energy (independent power supply, i.e. saving the power). Using the GPRS map as an additional step for development and choosing the best road for the emergency and police vehicles. For national highways we can also design the 8 road traffic light controllers.

CHAPTER 8.

CONCLUSION

Despite their numerous benefits, traffic lights are not without challenges. Traffic congestion remains a persistent issue in many urban areas, and poorly synchronized or outdated traffic light systems can exacerbate the problem. Additionally, the environmental impact of prolonged idling at red lights raises concerns about air quality and fuel consumption. To address these challenges, ongoing research focuses on developing smart traffic management solutions that leverage emerging technologies like artificial intelligence and the Internet of Things.

In conclusion, traffic lights are an integral part of modern transportation systems, shaping the way vehicles and pedestrians navigate intersections. Evolving from simple red and green signals to sophisticated, sensor-driven systems, traffic lights have played a pivotal role in enhancing safety and efficiency on roadways. As cities continue to grow and transportation needs evolve, the continued innovation and adaptation of traffic light systems will be essential to creating more sustainable and effective urban environments.

CHAPTER 9.

REFERENCES

1. Laboratory Exercise #12, The Traffic Light Controller Lab. "ECEN 248: Introduction to Digital Design". Department of Electrical and Computer Engineering Texas A&M University.

2. Rajeev Madhavan, Verilog HDL Reference Guide, Automata Publishing Company, CA, 1993. ISBN 0-9627488-4-6

3 E. Sternheim, Rajvir Singh, Rajeev Madhavan, Yatin Trivedi, Digital Design and Synthesis with Verilog HDL, Automata Publishing Company, CA, 1993. ISBN 0-9627488-2-X