International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 6 June 2023, pp: 828-832 www.ijaem.net ISSN: 2395-5252

Traffic Light Control System Based on FPGA

Aditya R. Girame[#1], Preeti S. Ingole[#2], Shrijivan D. Jadhav[#3], Payal R. Nanaware[#4], Aditya S.Nikam[#5], Rajveer Shastri[#6]

Department of Electronics and Telecommunication, VPKBIET Baramati , Maharashtra, India

[#1]aditya.girame.entc.2021@vpkbiet.org,[#2]preeti.ingole.entc.2022@vpkbiet.org[#3],shrijivan.jadhav.entc.2022@vpkbiet.org,[#4]payal.nanaware.entc.2021@vpkbiet.org[#5] aditya.nikam.entc.2022@vpkbiet.org[#6]rajveer.shastri@vpkbiet.org,

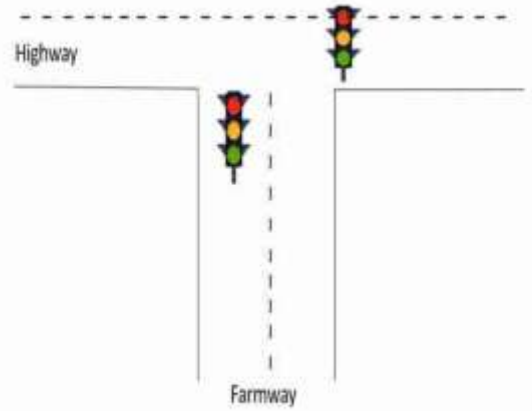
Abstract: The Traffic Light Controller plays a crucial role in regulating vehicular and pedestrian traffic, ensuring smooth and efficient flow on road intersections.This abstract presents the design and implementation of a Traffic Light Controller using the Basys 3 developmentboard. The Basys 3 board, equipped with a Xilinx Artix-7 FPGA, provides a versatile and powerful platform for real-time control applications.

The proposed Traffic Light Controller system utilizes theBasys 3 board's input/output capabilities, including buttons, switches, LEDs, and seven-segment displays, to simulate a realistic traffic scenario. The design follows a finite state machine (FSM) approach, enabling the controller to transition between different traffic lightphases accurately.

The controller's implementation involves designing and coding the FSM logic using a hardware description language (HDL) such as VHDL. The HDL code specifies the states, transitions, and outputs necessary for the correct operation of the Traffic Light Controller. The code is then synthesized and mapped onto the Basys 3 FPGA, utilizing the Xilinx Vivado design suite.

Upon successful implementation, the Traffic LightController can accurately replicate various traffic light patterns, including green, yellow, and red lights fordifferent directions, pedestrian crossings, and transition delays. The Basys 3 board's LED indicators and seven-segment displays are utilized to visualize andcommunicate the current state of the traffic lights.

Keywords: Traffic Light Controller, Basys 3, FPGA, Finite State Machine, Verilog, VHDL, Xilinx Vivado, Real-time Control, Traffic Flow.



1.Introduction:-

A traffic light controller is a crucial component in managing and regulating the flow of traffic at intersections. The Basys 3 is a development board that can be used to implement various digital designs, including a traffic light controller. In this introduction, I will provide an overview of a traffic light controller using the Basys 3 board.

The traffic light controller built using the Basys 3 board will be responsible for controlling the timing and sequencing of the traffic lights at an intersection. It will ensure that vehicles and pedestrians can safely navigate through the intersection while minimizing congestion and optimizing traffic flow.

To implement the traffic light controller, the Basys 3board can leverage its programmable logiccapabilities, such as the Xilinx Artix-7 FPGA and various input/output interfaces. The FPGA will be programmed to execute a specific logic design that handles the sequencing and timing of the trafficlights.

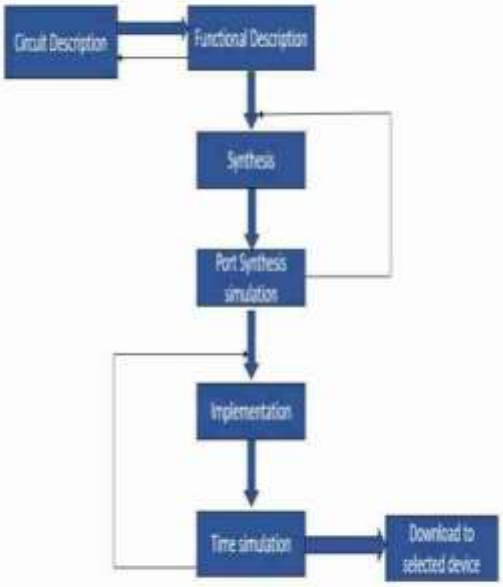


Fig. Design Flow

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 6 June 2023, pp: 828-832 www.ijaem.net ISSN: 2395-5252

3.Field Programmable Logic Array

A Field Programmable Gate Array (FPGA) is a type of integrated circuit (IC) that can be configured or programmed after manufacturing. Unlike application-specific integrated circuits which are designed for specific functions, FPGAs are flexibleand can be reprogrammed to perform different tasks or implement various digital circuits.

FPGAs consist of an array of configurable logic blocks interconnected by programmable routing.

channels. Each CLB typically contains lookup tables ,flip-flops, and other resources that can be configured to implement different logic functions. The routing channels allow signals to be routed between the CLBs, providing flexibility in designing complex digital circuits.

The programming of an FPGA is typically done using a hardware description language (HDL) suchas VHDL or Verilog. The HDL code describes the desired functionality of the circuit, and a synthesis tool converts it into a configuration file or bitstreamthat can be loaded onto the FPGA.

FPGAs are widely used in various applications,including digital signal processing, telecommunications, embedded systems, aerospace, industrial automation, and prototyping. They offer advantages such as reconfigurability, high-performance computing capabilities, parallel processing, and rapid development cycles.

In addition to the configurable logic resources, modern FPGAs often include other specialized components such as embedded processors (e.g., ARM cores), high-speed transceivers, memoryblocks, and other peripherals. This integrationallows FPGAs to handle a broader range of tasks and interface with external devices more efficiently.

Overall, FPGAs are valuable tools for implementing custom digital circuits and accelerating certain computational tasks. Theirflexibility, reprogrammability, and performance make them suitable for a wide range of applications in diverse industries.

3.1. Very High Speed Integrated Circuit Hardware Description Language(VHDL)

VHDL is a hardware description language used in the design and simulation of digital electronic systems. It is commonly used in the field of digital design and FPGA (Field-Programmable Gate Array)programming.

VHDL allows designers to describe the behavior andstructure of a digital system at various levels ofabstraction. It is a textual language that provides a standardized way to model and simulate digital circuits and systems before they are implemented inhardware.

VHDL supports the hierarchical design of digital systems, meaning that larger systems can be built bycombining smaller modules. It allows for the description of the internal structure of a module, the interconnections between modules, and the behaviorof the system as a whole.

Some common uses of VHDL include:

Designing and simulating digital circuits: VHDL is used to describe the behavior and structure of digitalcircuits, including logic gates, flip-flops, multiplexers, and other digital components. It allows designers to simulate and test their designs before fabrication.

FPGA programming: VHDL is often used toprogram FPGAs, which are programmable logic devices that can be reconfigured to implement digital circuits. By describing the desired behavior of a circuit in VHDL, it can be synthesized and programmed onto an FPGA.

Verification and testing: VHDL supports simulation, which enables designers to verify the correctness oftheir designs and test them under different scenarios and inputs. This helps identify and debug potential issues before moving to the physicalimplementation.

It's worth noting that VHDL is a complex language with its own syntax and rules.

Learning VHDL requires an understanding of digital design concepts and familiarity with the language itself. There are various tools and software available, such as Xilinx ISE, Vivado, and ModelSim, that support VHDL development, simulation, and synthesis.

Input		Output	
Sensor	Rst_n	Highway	Farmway
Initial Condition		Red	Red
0	0	Green	Red
0	1	Yellow	Red
1	0	Red	Green
1	1	Red	Yellow

4. 1. Traffic Light Controller Structure

State Machine: The traffic light controller is often implemented as a finite state machine . The FSMrepresents different states that the traffic lights can be in and defines the transitions between these states. Each state corresponds to a specific traffic light configuration (e.g., green light for one direction, red lights for others).

Inputs: The traffic light controller receives inputs fromvarious sources to determine when and how to change the lights. Common inputs include:

Timer: A timer ensures that each traffic light state remains active for the appropriate duration.

Outputs: The traffic light controller controls the outputs, typically LEDs, to indicate the current state ofthe traffic lights. Each output LED corresponds to a specific light color (red, yellow, or green) for a particular direction of traffic.

Logic and Control: The traffic light controller's logic determines when to change the traffic light states basedon the inputs. It may involve a combination of timers,sensor inputs, and predefined traffic light sequencing patterns. The control circuitry interprets the logic and generates the necessary signals to activate the appropriate LEDs.

4. . State Description

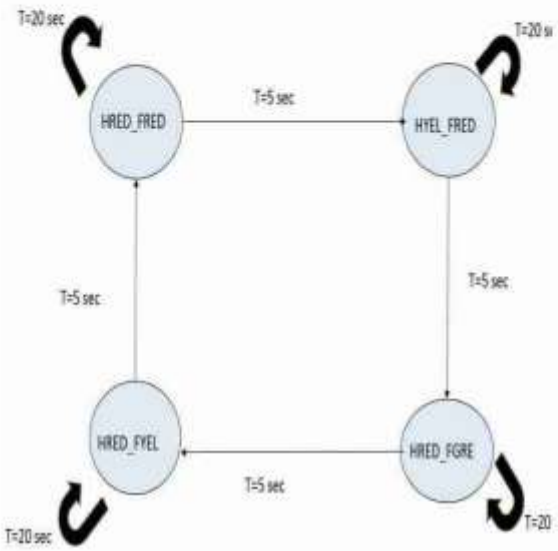
The sequence of traffic is a FSM_States: An enumeration type defining the states of the finite state machine.

HGRE_FRED: Highway green and farm red state.
HYEL_FRED: Highway yellow and farm red state.
HRED_FGRE: Highway red and farm green state.
HRED_FYEL: Highway red and farm yellow state
current_state, next_state: Signals of type FSM_States representing the current and next states of the FSM

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 6 June 2023, pp: 828-832 www.ijaem.net ISSN: 2395-5252

Sequential Logic:
A process triggered by the rising edge of the clocksignal (clk) increments the count signal on each clock cycle.
Another process triggered by the rising edge of thesclk signal assigns the next state based on the current state and the clock signal.
The sclk signal is derived from the 24th bit of thecount signal.

Fig 3: State Diagram



There are two inputs in this system: a sensor, a clock,and rst_n. The system follows a state diagram calledTLC (Traffic Light Controller) shown in Figure 3. It operates based on the changing values of these inputs.Initially, all the traffic lights are set to red, which is the initial condition. Then, the program execution begins as shown in Figure 4.

When rst_n is 0 and the sensor is 0, the green light in the Highway direction turns on for a few seconds, while the red signal light in all other directions, namelyFarmway, remains on.

When rst_n is 0 and the sensor is 1, the yellow light turns on for a few seconds, and the pedestrian Highwaylight turns on. After that, the value of the sensor is increased by one, and rst_n is set to zero.

If rst_n is 1 and the sensor is 0, the red light in the Farmway direction turns on for a few seconds, and allred lights in other directions remain on.

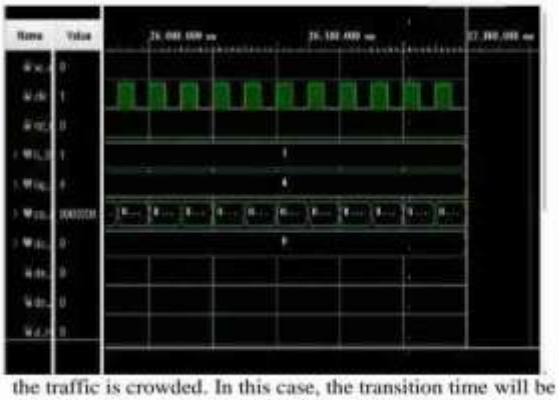
If rst_n is 1 and the sensor is 1, the yellow light turns on for a few seconds, and the pedestrian Highway light turns on.

Then, the value of the sensor is increased byone, and rst_n is set to zero.

This sequence continues, and the traffic flow is controlled by assigning time periods for the lights in both directions.

4.4 Simulation Result: -

Figure 6 shows the results of a simulation for a controller when the sensors output=logic "0" detectthat the traffic is slow. This means that the transitiontime, or the time it takes for something to change, will be shorter.
On the other hand, Figure 7 shows the results of a simulation for a controller when the sensorsoutput=logic "1" detect that



the traffic is crowded. In this case, the transition time will be longer, meaning it will take more time for something to change.

Fig 4.4.1: Simulation Result"Sensor"="0"

In above diagram shows the simulation result of input sensor is "0" and reset is also "0" so output will be shows Highway Green and Farm Red.

Fig:-7 Simulation Result"Sensor"="1"

In above diagram shows the simulation result of input sensor is "0" and reset is also "1" so output will be shows Highway Yellow and Farm Red.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 6 June 2023, pp: 828-832 www.ijaem.net ISSN: 2395-5252

Fig 8: Simulation Result"rst_n"="1"

In above diagram shows the simulation result of input sensor is "1" and reset is also "0" so output will be shows Highway Red and Farm Green.

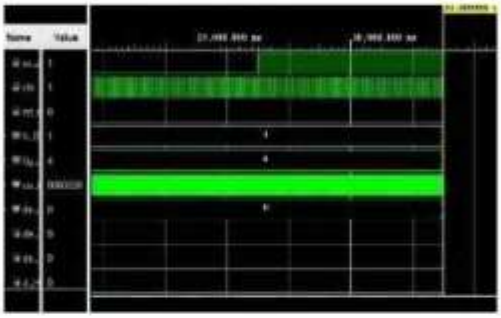


Fig 9: Simulation Result"rst_n"="1" & Sensor "1"

In above diagram shows the simulation result of input sensor is "1" and reset is also "1" so output will be shows Highway Red and Farm Yellow.

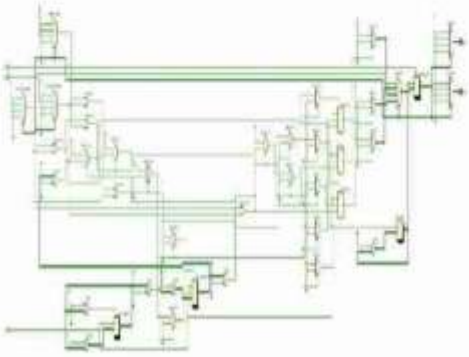


Fig. RTL view of Traffic Light Controller

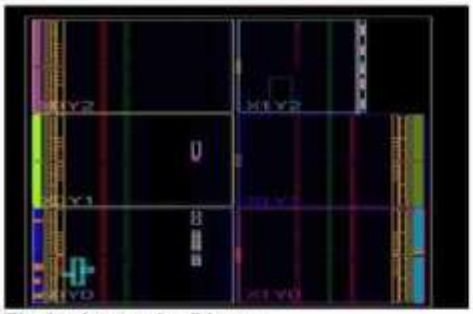
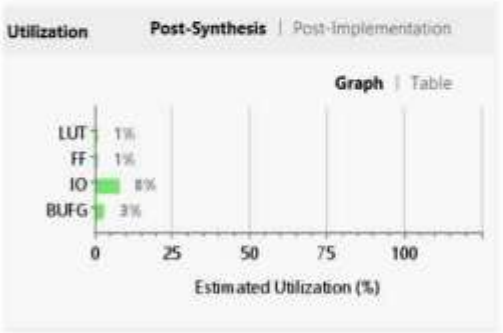


Fig. Implementation Diagram

An implementation diagram of a traffic light controller using VHDL typically consists of three main components: a state machine, input/output modules, and a timing module. The state machine defines the different states of the traffic lights and their transitions based on inputs. The input/output modules handle the signals received from the sensors and control the output signals to the traffic lights. The timing module ensures proper timing between state transitions and signal changes. These components are interconnected through signals and ports, forming a hierarchical design. VHDL code is written to describe the behavior and interconnections of these components, allowing for simulation and synthesis to obtain a physical implementation on a programmable device.



A utilization graph of a traffic light controller using VHDL showcases the resource allocation and utilization of the programmable device. It provides an overview of how different components and modules are distributed and utilized within the design.

The graph displays the percentage of resources, such as logic elements, memory, and I/O pins, occupied by each module or component. By analyzing the utilization graph, designers can optimize the design, adjust resource allocation, and ensure efficient utilization of the programmable device for the traffic light controller implementation.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 5, Issue 6 June 2023, pp: 828-832 www.ijaem.net ISSN: 2395-5252

Timing	Setup	Hold	Pul
Worst Negative Slack (WNS):	5.32 ns		
Total Negative Slack (TNS):	0 ns		
Number of Failing Endpoints:	0		
Total Number of Endpoints:	96		
Implemented Timing Report			

The utilization value of a traffic light controller using VHDL refers to the percentage of resources on a programmable device that are occupied by the design. It provides insight into how much of the device's available resources, such as logic elements, memory, and I/O pins, are utilized by the traffic light controller implementation. A higher utilization value indicates a more resource-intensive design, while a lower value suggests a more efficient use of available resources. Designers can use the utilization value to optimize the design, balance resource allocation, and ensure efficient utilization of the programmable device.

The chip power of a traffic light controller using VHDL refers to the power consumption of the programmable device on which the controller is implemented. It represents the amount of electrical power required for the device to operate the traffic light controller design. Chip power is influenced by various factors, including the complexity of the design, the frequency of operation, and the resource utilization. By analyzing the chip power, designers can assess the energy efficiency of the traffic light controller implementation and make optimizations to minimize power consumption, leading to more sustainable and cost-effective operation.

Conclusion

The implementation of a traffic light controller using the Basys 3 development board offers a practical and efficient solution for managing traffic flow at intersections. This project demonstrates the effective utilization of digital logic design principles and FPGA programming.

The traffic light controller designed with Basys 3 provides several key benefits. Firstly, it enhances traffic safety by organizing the flow of vehicles, cyclists, and pedestrians in a systematic and controlledmanner. By assigning specific time intervals to each direction, it minimizes the risk of collisions and improves overall traffic efficiency.

Reference

[1] Jose E. Ortiz and Robert H. Klenke. "Simple Traffic Light Controller: A Digital Systems Design Project," IEEE SoutheastCon 2010(SoutheastCon), Concord, NC, March 2010, pp. 85-88
[2] W.M. El-Medany and M.R. Hussain. "FPGA-Based Advanced Real Traffic Light Controller System Design," Technology and Applications, 2007. IDAACS 2007. 4th IEEE Workshop, pp. 100-105
[3] M.F.M. Sabri, M.H. Husin, W.A.W.Z Abidin, K.M. Tay and H.M. Basri "Design of FPGA-based Traffic Light Controller System," Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference, Shanghai, June 2011 pp.114-118
[4] S. Shi, T. Hongli and Z. Yandong "Design of Intelligent Traffic Light Controller Based on VHDL," Knowledge Discovery and Data Mining, 2009. WKDD 2009. Second International Workshop, pp. 272 – 275
[5] H. Taehee and L. Chiho "Design of an intelligence traffic light controller (ITLC) with VHDL," TENCON '02. Proceedings. 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering, Oct. 2002 , pp. 1749 - 1752 vol.3
[6] L. Zhenggang, X. jiaolong, Z. Mingyun and D. Hongwei "FPGA-Based Dual-Mode Traffic Lights System Design," Information Science and Engineering (ICISE), 2009 1st International Conference, Dec. 2009, pp.558-561