WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Design and Implementation of an Indian Traffic Light Controller for a Four-Lane Junction

Subbi Naidu Bora[1], *Anji Kumar Narukula* [2]

[1].Assistant Professor, Rajiv Gandhi University of Knowledge and Technology , Nuzvid,Andhra Pradesh 521202, India, subbinaidu.bora@rguktn.ac.in

[2].B.Tech-Student,Dept. of Electrical Engineering,RGUKT, Nuzvid,Eluru 521202 anjikumarnarukula@gmail.com

DOI : https://doi.org/10.55248/gengpi.6.0625.22117

ABSTRACT—

This paper presents the design and implementation of a Traffic Light Controller (TLC) optimized for Indian road conditions, utilizing the Nexys A7 FPGA board. The proposed TLC manages a four-lane junction, where vehicles drive on the left side of the road and left turns are always permitted. Employing a Finite State Machine (FSM) approach, the system controls the sequence of traffic lights for North, South, East, and West directions, regulating both straight and right-turn movements while ensuring continuous left-turn functionality. The design aims to optimize traffic management by alternating signal lights to reduce congestion and prevent accidents. The paper details the system design, simulation, implementation phases, including code and hardware prototype, and provides performance metrics and outcomes demonstrating the controller's effectiveness in improving traffic flow at intersections.

Index Terms—Traffic Light Controller, Finite State Machine (FSM), FPGA Nexys A7, Verilog HDL, Traffic Management,Intersection Control, Left Turn Allowance, Simulation, Hardware Prototype.

I. INTRODUCTION

Traffic congestion is a significant issue in urban areas, contributing to increased travel times, pollution, and accidents. Effective traffic management is crucial for optimizing vehicle flow and ensuring safety at intersections. A key component of traffic management systems is the traffic light controller, which regulates the timing and sequencing of signals to manage the movement of vehicles and pedestrians. In India, where vehicles drive on the left side of the road, traffic light systems must accommodate unique challenges, such as allowing left turns at all times while managing straight and right-turn movements. Traditionally, traffic light controllers have been implemented using analog or microcontroller-based systems. However, these methods often lack the flexibility and scalability required for modern traffic management challenges. With advancements in digital technology, Field Programmable Gate Arrays (FPGAs) have emerged as a powerful tool for implementing complex control systems. FPGAs offer high-speed processing, parallelism, and reconfigurability, making them ideal for real-time traffic control applications. This paper presents the design and implementation of a Traffic Light Controller optimized for a four-lane junction under Indian traffic conditions. The proposed system utilizes a Finite State Machine (FSM) to manage the sequence of traffic lights for four directions: North, South, East, and West. The controller is designed to optimize traffic flow by ensuring efficient signal timing and minimizing congestion, while also allowing for continuous left-turn traffic. The design ensures that opposite lanes (M1 with M3, M2 with M4) can move simultaneously, providing equal timing for all lanes. The paper is organized as follows: Section II provides an overview of the traffic light control system and the specific requirements for the FPGA-based implementation.

Section III describes the design and architecture of the FSM used in the controller. Section IV details the implementation process, including hardware setup and software coding. Section V presents the simulation results and performance metrics of the system. Finally, Section VI concludes the paper with a summary of the findings and potential future work. This research contributes to the field of traffic management by demonstrating the feasibility and effectiveness of using FPGA technology for traffic light control, offering a scalable and adaptable solution to address the challenges of modern traffic systems, particularly in regions with unique driving conditions like India.

Fig. 1. Traffic View

II. LITERATURE SURVEY

In the early stages, traffic light controllers were primarily analog, relying on mechanical timers, which lacked the ability to adapt to changing traffic conditions [1]. The transition to microcontroller-based systems marked an improvement in flexibility, with the introduction of time-of-day based adjustments to traffic signals [2]. With the advent of FPGA technology, more advanced and adaptable traffic controllers were developed, allowing for real-time adjustments using finite state machines [3]. An FPGA-based traffic control system using IR sensors and cameras optimizes signal timing and prioritizes emergency vehicles, reducing delays and improving efficiency. Implemented with Verilog HDL and tested on Xilinx14.3 [4].Further advancements in FPGA technology led to the integration of adaptive control strategies, where signal timings could dynamically adjust based on real-time traffic density [5]. In recent years, the concept of smart traffic management systems has emerged, utilizing FPGAs in conjunction with IoT devices to optimize traffic flow across entire cities [6]. The latest research has explored the potential of integrating machine learning algorithms into FPGA-based controllers, enabling predictive traffic control using real-time data [7]. In 2020, the concept of smart traffic systems was expanded by integrating edge computing with FPGA-based controllers to reduce latency in traffic signal adjustments. In 2019, researchers explored the integration of 5G communication technologies with traffic light controllers, enhancing realtime data exchange between vehicles and traffic management systems [8]. The year 2022 saw the implementation of deep learning models directly on FPGA hardware, improving the predictive capabilities of traffic signal systems by analyzing large volumes of traffic data in real-time. In 2023, studies demonstrated the effectiveness of hybrid systems combining traditional rule-based traffic control with machine learning algorithms, achieving better traffic flow optimization in complex urban environments. Recent developments in 2024 have focused on energy-efficient traffic light controllers using low power FPGAs, aiming to reduce the environmental impact of smart city infrastructure [9].

III. SYSTEM DESCRIPTION

A. Junction Layout

The four-lane junction consists of:

- M1 and M3: Opposite roads, with traffic lights controlling straight and right-turn movements.

- M2 and M4: Opposite roads, with similar traffic controlling straight and right-turn movements.

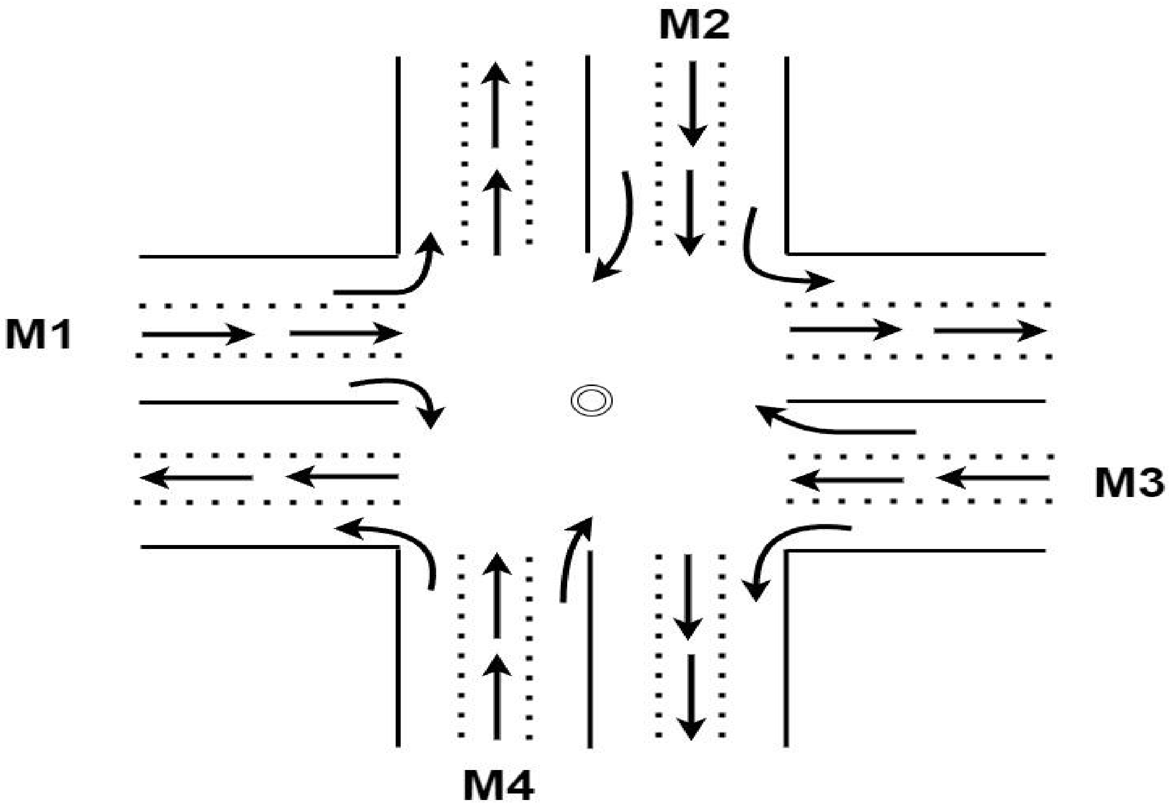

Fig. 2. 4-Lane junction

B. Traffic Light Notations

- M1SG: Main Road 1 Straight Traffic Green
- M1SY: Main Road 1 Straight Traffic Yellow
- M1SR: Main Road 1 Straight Traffic Red
- M1RTG: Main Road 1 Right Turn Green
- M1RTY: Main Road 1 Right Turn Yellow
- M1RTR: Main Road 1 Right Turn Red
- M2SG: Main Road 2 Straight Traffic Green
- M2SY: Main Road 2 Straight Traffic Yellow
- M2SR: Main Road 2 Straight Traffic Red
- M2RTG: Main Road 2 Right Turn Green
- M2RTY: Main Road 2 Right Turn Yellow
- M2RTR: Main Road 2 Right Turn Red
- M3SG: Main Road 3 Straight Traffic Green
- M3SY: Main Road 3 Straight Traffic Yellow
- M3SR: Main Road 3 Straight Traffic Red
- M3RTG: Main Road 3 Right Turn Green
- M3RTY: Main Road 3 Right Turn Yellow
- M3RTR: Main Road 3 Right Turn Red
- M4SG: Main Road 4 Straight Traffic Green

- M4SY: Main Road 4 Straight Traffic Yellow
- M4SR: Main Road 4 Straight Traffic Red
- M4RTG: Main Road 4 Right Turn Green
- M4RTY: Main Road 4 Right Turn Yellow
- M4RTR: Main Road 4 Right Turn Red

This notation system is similarly applied to M2, M3, and M4.

C. Traffic Management Strategy

The system is designed with three primary traffic parts:

- **Left Turn:** Always allowed, ensuring that vehicles can turn left regardless of the state of other traffic lights.
- **Straight Traffic:** Managed by the FSM, which controls the green, yellow, and red lights for straight movements.
- **Right Turn Traffic:** Also managed by the FSM, controlling the green, yellow, and red lights for right turns.

IV. DESIGN METHODOLOGY

A. Finite State Machine (FSM) States

The FSM includes the following states, with each state controlling specific traffic light configurations:

- **S0:** Main Roads 1 and 3 have green lights for straight traffic. Main Roads 2 and 4 have red lights for straight traffic. Right turns for all roads are red. Left turns are allowed continuously.
- **S1:** Main Roads 1 and 3 switch to yellow for straight traffic. Main Roads 2 and 4 remain red. Right turns remain red.
- **S2:** Main Roads 1 and 3 turn red for straight traffic. Main Roads 2 and 4 turn green for straight traffic. Right turns for all roads remain red.
- **S3:** Main Roads 2 and 4 switch to yellow for straight traffic. Main Roads 1 and 3 remain red. Right turns remain red.
- **S4:** Main Roads 2 and 4 are red. Right turns for Main Roads 1 and 2 are green. Right turns for Main Roads 3 and 4 remain red.
- **S5:** Right turns for Main Roads 1 and 2 turn yellow. Right turns for Main Roads 3 and 4 remain red.
- **S6:** Right turns for Main Roads 1 and 2 turn red. Right turns for Main Roads 3 and 4 turn green. Main Roads 1 and 3 remain red.
- **S7:** Right turns for Main Roads 3 and 4 turn yellow. Main Roads 1 and 3 turn green for straight traffic. Main Roads 2 and 4 remain red.
- **S8:** Main Roads 1 and 3 turn red for straight traffic. Right turns for Main Roads 3 and 4 turn red.

B. Timing Specifications

The timing for each traffic light phase is as follows:

- Green Light: 40 seconds
- Yellow Light: 5 seconds
- Right Turn Light: 30 seconds

Timers are used to manage the duration of each light phase, ensuring smooth transitions and consistent traffic management.

V. VERILOG HDL IMPLEMENTATION

A. Module Definition

The Traffic Light Controller is designed as a Verilog module, which is a fundamental building block in hardware description languages used for digital logic design. The module's primary purpose is to control traffic signals at a four-lane intersection, ensuring the safe and efficient flow of vehicles.

The module takes in two primary inputs:

- **Clock Signal:** This input provides the timing reference for all sequential logic operations within the controller. The clock signal is fundamental to synchronizing state transitions and timing operations across the system.

• **Reset Signal:** This input initializes the controller to a known state, typically the first state of the FSM (Finite State Machine). It ensures the controller starts correctly and recovers from any unexpected conditions or power cycles.

The outputs of the module are the control signals for the traffic lights, which include green, yellow, and red signals for both straight and right-turn lanes on all four roads (M1, M2, M3, and M4). The outputs are organized to reflect the traffic light configurations as defined in the FSM.

Example of module definition in Verilog:

```
module Traffic Light Controller(
         input wire clk,
        input wire reset,
        output reg [2:0] M1, M2, M3, M4
         );
```

Here, M1, M2, M3, and M4 represent the state of the traffic lights (green, yellow, red) for the corresponding main roads.

B. State Machine Logic

The core of the Traffic Light Controller is a Finite State Machine (FSM). The FSM logic in Verilog dictates how the controller transitions between different states based on timing and input conditions. Each state represents a unique configuration of traffic lights for the intersection, controlling both straight and right-turn movements.

• **State Definitions:** Each state (S0 to S8) corresponds to a specific configuration of green, yellow, and red lights for the main roads (M1, M2, M3, M4). For example:

– S0: Main Roads 1 and 3 (M1, M3) have green lights for straight traffic; Main Roads 2 and 4 (M2, M4) have red lights.

– S1: M1 and M3 switch to yellow, indicating a transition period before the red light.

– And so on, until S8, which cycles back to S0.

• **State Transitions:** The FSM moves from one state to the next based on a counter that represents elapsed time. The transitions ensure that each state is held for the appropriate duration (e.g., 40 seconds for green, 5 seconds for yellow). The logic is implemented using case statements in Verilog, with each case representing a state and specifying the conditions for transitioning to the next state.

C. Timing and Counters

Timing is crucial for the traffic light controller to function correctly. The FSM relies on counters to manage the duration of each state, ensuring each traffic light signal is displayed for the appropriate amount of time.

• **Counters:** Separate counters are used for green, yellow, and right-turn signals. When a counter reaches its preset value (e.g., 40 seconds for green lights), the FSM transitions to the next state. The counters reset after each state transition.

• **Clock Division:** To manage real-world timing (e.g., seconds), the high-frequency clock signal from the FPGA is divided down to a lower frequency. This lower frequency clock drives the counters, ensuring accurate timing for the traffic light signals.

VI. TESTBENCH

A. Testbench Overview

A comprehensive test bench is essential to verify the functionality of the Traffic Light Controller. The test bench is a separate Verilog module that simulates the environment around the Traffic Light Controller and monitors its outputs to ensure correctness.

• Test bench Inputs: The test bench generates clock and reset signals, applies them to the Traffic Light Controller module, and stimulates it with a sequence of test cases that cover all possible states and transitions.

• Expected Outputs: The test bench checks the outputs of the Traffic Light Controller against expected values. It verifies that the traffic lights transition correctly through green, yellow, and red phases as per the FSM specifications.

VII. RTL BLOCK DESIGN

A. RTL Block Overview

The Register Transfer Level (RTL) block diagram of the Traffic Light Controller represents the digital logic design implemented using Verilog HDL. The RTL abstraction describes the flow of data between registers and the logical operations performed on that data within each clock cycle. In the context of the Traffic Light Controller, the RTL block diagram illustrates the various components involved in managing traffic signals and the finite state machine (FSM) responsible for transitioning between states based on timing requirements.

• **Clock and Reset Signals:** The clock (clk) signal is the driving force for synchronous operations within the controller. All state transitions and timing calculations are based on the rising edge of the clock. The reset (reset) signal initializes the FSM to the starting state, ensuring a predictable and safe operation at system startup or in the event of an unexpected condition.

• **Finite State Machine (FSM):** The FSM is the core component of the RTL block, governing the state transitions of the traffic light controller. Each state (S0 to S8) represents a unique configuration of traffic lights, controlling both straight and right-turn signals across all

four main roads (M1, M2, M3, M4). The FSM logic determines the next state based on the current state and the values of timing counters, which are updated on each clock cycle.

• **Traffic Light Control Signals:** The output signals (M1,M2, M3, M4) represent the traffic light states for each road. These signals are encoded to indicate green, yellow, or red for straight and right-turn traffic. The RTL design ensures that these signals are updated in accordance with the FSM state transitions, providing real-time control of traffic lights at the junction.

• **Counters and Timing Logic:** Counters are used to keep track of the duration each traffic light remains in its current state (e.g., green, yellow, red). The timing logic increments these counters on each clock cycle and signals the FSM when it is time to transition to the next state. This mechanism ensures that each light phase is maintained for the correct duration, adhering to traffic management requirements.

VIII. SIMULATION

The simulation of the RTL design was conducted using ModelSim, a widely used simulator for verifying HDL designs. The results demonstrate the correct operation of the Traffic Light Controller, validating both the FSM state transitions and the timing of the traffic light signals.

• **State Transition Validation:** The simulation confirmed that the FSM transitions through all defined states (S0 to S8) in the correct sequence. Each state transition is triggered by the corresponding counter reaching its preset value, ensuring that traffic lights change as per the design specifications.

• **Timing Accuracy:** The simulation results showed that each traffic light remains in its designated state (green, yellow, red) for the correct duration. The counters effectively manage the timing, with no observed deviations from the expected behavior. For example, green lights are maintained for 40 seconds, yellow lights for 5 seconds, and right-turn signals for 30 seconds, matching the design requirements.

• **Continuous Left Turn Functionality:** One of the unique features of this Traffic Light Controller is the continuous allowance of left turns, regardless of the state of other traffic lights. The simulation results verified that the leftturn signals remain consistently enabled across all states, ensuring that vehicles can always make left turns, which is crucial for the traffic flow in the given junction layout.

• **Robustness and Reliability:** The RTL design was also tested for edge cases, such as power-on reset and quick successive resets, to ensure reliable operation. The results demonstrated that the controller resets to the initial state (S0) correctly and resumes normal operation, indicating robust handling of reset conditions.

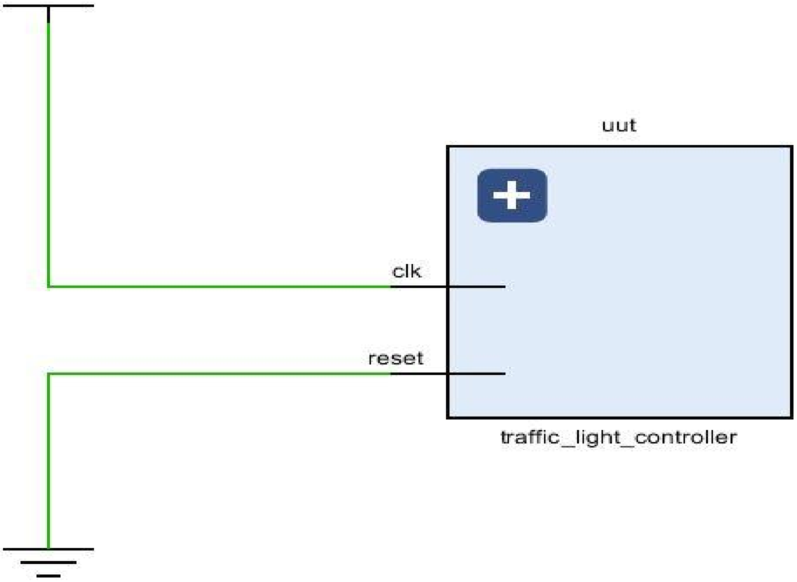

Fig. 3. RTL Diagram

The RTL block diagram (Figure 3) provides a visual representation of the design, illustrating the flow of signals and the interconnections between the FSM, counters, and traffic light control outputs. This diagram helps in understanding the overall architecture and functioning of the Traffic Light Controller.

A. Conclusion on RTL Design: The RTL design and simulation results affirm the effectiveness of the Traffic Light Controller in managing a four-lane junction with continuous left-turn allowance, optimized for Indian road conditions. The FSM-based approach, combined with accurate timing and robust state management, ensures smooth traffic flow and enhances road safety. Future enhancements could include adaptive signal control based on real-time traffic conditions, further improving the system's efficiency and responsiveness.

B. Simulation Results:Simulation results provide evidence of the Traffic Light Controller's functionality. The results are obtained using a simulation tool (e.g., ModelSim or Vivado Simulator), which executes the Verilog code and displays the timing and state transitions.

• Correct State Transitions: The results confirm that the FSM moves through all states (S0 to S8) in the correct order, with each state held for the specified duration.

• Timing Verification: The simulation verifies that the traffic light signals adhere to the timing requirements (e.g., green light for 40 seconds, yellow for 5 seconds).

• Continuous Left Turn Functionality: The testbench confirms that left turns are always allowed, as required by the design specifications.

C. Verification: Verification is the process of ensuring that the design meets all functional and timing requirements. In this project, verification involves checking that the Traffic Light Controller operates as expected under all conditions defined in the FSM.

• Functional Verification: Ensures that the traffic lights switch correctly between green, yellow, and red, and that right-turn signals activate according to the FSM.

• Timing Verification: Confirms that the counters accurately control the duration of each state, ensuring that no state is too short or too long.

• Edge Cases and Robustness: The testbench also tests edge cases, such as immediate reset after state changes, to ensure the controller is robust and handles unexpected inputs gracefully.

IX. CONCLUSION

A. Summary:This paper has presented the design and implementation of a Traffic Light Controller tailored for a four-lane junction, optimized for Indian driving conditions. The design uses a Finite State Machine (FSM) implemented in Verilog HDL to manage traffic light signals, ensuring efficient traffic flow and safety at the intersection. The FSM-based approach regulates straight and right-turn movements while continuously allowing left turns. The Verilog implementation has been thoroughly tested using simulation, confirming the controller's functionality and adherence to timing requirements.

B. Future Work: Future enhancements to this Traffic Light Controller could include:

• **Adaptive Signal Control**: Developing algorithms to adjust the duration of green and red signals dynamically based on real-time traffic conditions to further reduce congestion.

• **Integration with Sensor Data**: Enhancing the FSM to incorporate data from road sensors or cameras for smarter traffic management.

• **Hardware Optimization**: Exploring more efficient hardware implementations that reduce power consumption and increase reliability in real-world conditions.

REFERENCES

[1] Dunn Engineering Associates, Traffic Control Systems Handbook, prepared for the Federal Highway Administration, Office of Transportation Management, Washington, D.C., in association with Siemens Intelligent Transportation Systems, Oct. 2005.

[2] A. I. F. Awami and Z. Yusupov, "Microcontroller-based Traffic Light Control System," in Proc. 5th Int. Symp. Innovative Technol. Eng. Sci. (ISITES), Baku, Azerbaijan, Sep. 2017. [3] V. V. Dabahde and R. V. Kshirsagar, "FPGA-Based Intelligent Traffic Light Controller System Design," Int. J. Innovative Sci. Eng. Technol., vol. 2, no. 4, Apr. 2015.

[4] T. B. O. Reddy and V. Sowmya, "An Advanced Traffic Light Controller using Verilog HDL," International Journal of VLSI System Design and Communication Systems, vol. 5, no. 7, pp. 657-661, July 2017.

[5] S. Kaur, T. V. S, V. V, and P. K. H, "Adaptive Traffic Light Controller using FPGA," in Proceedings of the 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology (RTEICT), Bangalore, India, May 2018

[6] S. L. Qaddori and N. T. Gadawe, "Real-Time Traffic Light Controller System based on FPGA and Arduino," in Proceedings of the International Multi-Disciplinary Conference on Signal, Data and Processing (IMDCSDSP), Cyberspace, June 2020.

[7] J. Zhang, F. Zhang, M. Xie, X. Liu, and T. Feng, ”Title of the paper,” in Proceedings of the 2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT), Xi’an, China, Aug. 18-20, 2021.

[8] A. Li, X. Qiu, W. Yuan, X. Huang, X. Ban, and Y. Zhang, ”Traffic Control System Based on 5G Communication Network,” in Proceedings of the 2019 IEEE International Conference on Communications (ICC), Shanghai, China, May 20-24, 2019.

[9] A. Devipriya, H. Girish, V. Srinivas, N. A. Reddy, and D. Rajkiran, ”RTL Design and Logic Synthesis of Traffic Light Controller for 45nm Technology,” in Proceedings of the 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, May 27-29, 2022.