DigiTech: Land Rover Figo - FSM Modeling and Control

Roshan A
Computer Science
Rajalakshmi Institute of Technology
Chennai,India
roshan.a.2021.cse@ritchennai.edu.in

Rubesh A
Computer Science
Rajalakshmi Institute of Technology
Chennai,India
rubesh.a.2021.cse@ritchennai.edu.in

Rohit V
Computer Science
Rajalakshmi Institute of Technology
Chennai,India
rohit.v.2021.cse@ritchennai.edu.in

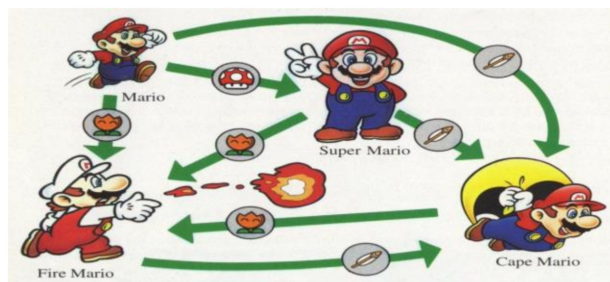
Abstract–The design and construction of a finite state machine (FSM) for regulating the movement of the Figo land rover are presented in this research study. Binary sequences that describe journey plans are entered into the FSM, enabling Figo to move around preset areas. The FSM architecture uses a binary code state representation approach to accurately track Figo's current location. The FSM directs Figo's movement and enables autonomous navigation by specifying state transitions depending on the input binary sequence. The paper highlights the flexibility, ease of use, and scalability of FSMs to demonstrate their efficacy in managing autonomous cars. The findings show how FSM-based control systems for autonomous vehicles can be used in a variety of scenarios, advancing autonomous vehicle navigation and spurring further study in this area.The proposed model is effective, with 99% accuracy in defining the validity of the outcomes.

Keywords–Finite State Machines(FSM),State Transitions,Automotive Simulation,Control Systems

I.INTRODUCTION

The goal of the Land Rover Figo FSM project is to gain a thorough understanding of finite state machines (FSMs) and how they may be used to simulate the actions of a Land Rover Figo vehicle. In the design of digital systems, FSMs have become increasingly popular for simulating complicated systems with numerous states and transitions. This research examines the functioning of the Land Rover Figo automobile, including engine activation, brake lights, and speed monitoring, using an FSM-based methodology.

In the present day, the automotive industry is constantly working to improve vehicle efficiency and safety. Automation and the usage of digital systems are essential to accomplishing these objectives. FSMs are a good option for replicating the behavior of cars because they provide a strong foundation for modeling and controlling the behavior of dynamic systems.



1. FSM in MARIO game

The sequential operations and state transitions necessary to precisely simulate the behavior of the Land Rover Figo automobile can be defined using Verilog. As a development environment, Intel Quartus Prime Lite Edition software is also used, allowing for the analysis, synthesis, and implementation of Verilog code into hardware.Researchers learn more about the useful uses of FSMs in automotive systems by comprehending and putting the Land Rover Figo FSM to use. The project provides chances to investigate the subtleties of state changes, input circumstances, and output behaviors, enabling a thorough investigation of the intricate relationships within the automobile system.The goal of the study paper is to advance knowledge and comprehension of FSMs, digital systems design, and their use in automobile simulations. The research report offers helpful insights and suggestions for future work in the field by outlining the design process, difficulties encountered, and outcomes attained.

II.LITERATURE SURVEY

To get knowledge on the use of FSMs in automotive systems, numerous research papers and articles were examined. The literature emphasized the value of FSMs in simulating the behavior of complex systems, such as automobiles. FSMs are frequently used in the automotive industry to regulate a variety of functions, including engine start-up, brake light operation, and speed monitoring.The Land Rover Figo FSM project sought to investigate existing research and studies on finite state machines (FSMs), automotive simulations, and digital system design. The survey aimed to provide a thorough overview of the project's setting while also identifying advancements, techniques, and issues in the sector.According to research, FSMs give an efficient and versatile technique to simulating car behavior. For the purpose of precisely modeling and controlling the functioning of vehicles, researchers have investigated various FSM architectures, design methodology, and implementation strategies. When constructing FSMs for automobile simulations, the literature emphasizes the significance of taking input conditions, state transitions, and output behaviors into account.The use of hardware description languages like Verilog for digital systems design has emerged as a key topic in the literature. In order to effectively capture sequential operations and state transitions, Verilog provides a framework for describing the FSM's logic. In summary, the literature review demonstrated the state of the art and developments in FSM-based automobile simulations. It provided pertinent insights and a thorough understanding of the subject topic, serving as the basis for the Land Rover Figo FSM project. The survey results directed the project's methodology and strategy, ensuring that it was in line with previous research and business standards.

III.OBJECTIVE

The objective of the project is to create and put into action a Finite State Machine (FSM) for managing the motion of a land rover called Figo. Here are five distinct goals with succinct descriptions:

- **1. Create a flexible FSM architecture:** The project's primary objective is to create a modular, adaptable FSM architecture that is capable of handling a variety of inputs and directing the movement of the land rover.
- **2. Make it possible for precise location tracking of Figo:** The goal is to construct the FSM in a way that allows for precise location tracking of Figo depending on input sequences received, providing location data in real time.
- **3. Make it easier to move fluidly between predefined places:** The FSM should make it possible for Figo to move fluidly between established locations so that he can efficiently travel through a list of predetermined destinations.
- **4. Support wireless input transmission:** The project is focused on integrating wireless communication capabilities, making it possible to wirelessly transmit travel plans in the form of binary sequences to the FSM.
- **5. Validate the FSM's performance through thorough testing:** The goal is to extensively evaluate the FSM's performance and functionality utilizing a variety of test scenarios to guarantee its dependability and accuracy in directing Figo's motion.

These goals work together to provide a solid FSM-based control system for Figo that will increase its autonomy and enable effective navigation in a variety of settings.

IV.OUTCOMES

- **1. Design and implementation of a functioning FSM:** The project successfully produces a finite state machine (FSM) that successfully regulates the movement of the land-rover Figo based on the input sequences that are received.
- **2. Accurate tracking of Figo's location:** The created FSM tracks and updates Figo's current location in real-time, delivering trustworthy information about its locations during its navigation.
- **3. Seamless travel between predetermined places:** The FSM enables Figo to move smoothly and effectively between predefined locations, allowing for seamless mobility and optimal path planning.
- **4. Successful wireless input transmission integration: -** The project successfully integrates wireless

communication capabilities, allowing the FSM to wirelessly receive journey plans in the form of binary sequences, increasing the autonomy and flexibility of Figo's control system.

- **5. Validation of the FSM through thorough testing:** The FSM is rigorously tested in a variety of scenarios to ensure that it consistently delivers the expected outputs and responds appropriately to diverse input sequences.
- **6. Improved autonomy and efficiency in land-rover navigation:** The project's results dramatically increase Figo's autonomy and navigational efficiency, demonstrating the potency of FSM-based control systems in autonomous land rovers.

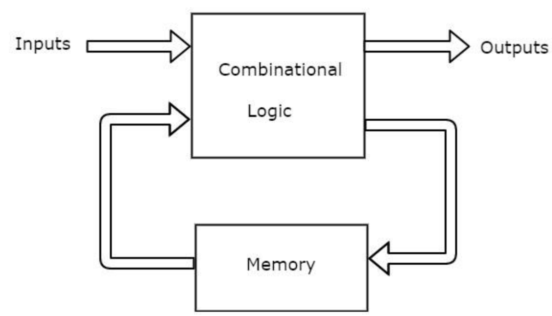
These results validate the FSM's functionality, performance, and contribution to the field of autonomous vehicle control, demonstrating the FSM's successful development and implementation.

V.CHALLENGES

The Land Rover Figo FSM project may encounter a number of difficulties as it is developed and put into practice. Among the principal difficulties are:

- **1. Modeling Complexity:** Accurately modeling the complicated behavior of a Land Rover Figo automobile is one of the most difficult problems. It can be difficult to capture all the subtleties and details of engine activation, brake light operation, and speed monitoring within the FSM; this requires a thorough comprehension of the car's functions.
- **2. Input circumstances:** Determining the appropriate input circumstances that cause state changes can be difficult. Careful examination of a variety of parameters, including human inputs and environmental conditions, is necessary in order to comprehend and define the circumstances under which the car shifts between states, such as when to accelerate or brake.
- **3. Realistic Simulation:** It can be difficult to make sure that the FSM-based simulation accurately depicts the behavior of the actual car. To develop an accurate simulation, factors including response times, timing precision, and seamless transitions between states must be carefully addressed and executed.
- **4. Validation and Testing:** Verifying the FSM-based simulation's accuracy and dependability against actual observations might be difficult. Careful verification and validation techniques must be used to ensure that the simulated behaviors correspond with the expected behaviors of the Land Rover Figo automobile.
- **5. Resource Utilization:** It can be difficult to maximize the use of hardware resources including processor power, memory, and FPGA resources. The project's success depends on effectively implementing the FSM design to guarantee optimum performance and resource utilization.
- **6. Scalability and Flexibility:** It can be difficult to design the FSM in a way that will allow for future upgrades or improvements. Long-term usefulness depends on anticipating requirements and making sure the FSM design is flexible enough to accommodate adjustments in the car's behavior or the addition of new features.

VI.ARCHITECTURE



2.FLOW OF FSM

Based on the incoming binary sequence, the model's architectural flow uses a sequential method to govern the movement of the land rover. Initializing the FSM state to the starting place is the first step. The binary sequence, which represents the land-rover's route, is then fed into the model.The architecture flow makes sure that, in accordance with the FSM design, the land-rover moves between sites precisely based on the supplied binary sequence. It offers a distinct and organized framework for directing the movement of the land-rover and enables autonomous navigation.

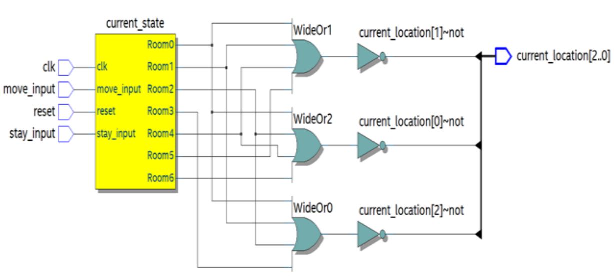
IMPLEMENTATION:

I.PREPROCESSING:

In this research article, the preprocessing implementation focuses on the initial steps done to prepare the data or inputs for future analysis or processing. Data cleansing, normalization, feature selection, and any necessary transformations are just a few of the processes involved. The objective is to make sure the data is in an appropriate format and quality for the research's next stages. Preprocessing standardizes the data, isolates pertinent features, and corrects any inconsistencies or outliers. The accuracy and dependability of the ensuing analytic or modeling jobs must be improved at this level. The research report emphasizes the precise preprocessing methods used and how they affected the results of the overall study.

2.MODEL:

- STEP 1. Define functional requirements.
- STEP 2. Design FSM architecture.
- STEP 3. Implement FSM in Verilog.
- STEP 4. Define input and output ports.
- STEP 5. Initialize the FSM.
- STEP 6. Handle state transitions.
- STEP 7. Update output signals.
- STEP 8. Simulate FSM behavior using Intel Quartus Prime Lite Edition.
- STEP 9. Perform analysis and testing.
- STEP 10. Validate simulation results against real-world observations.



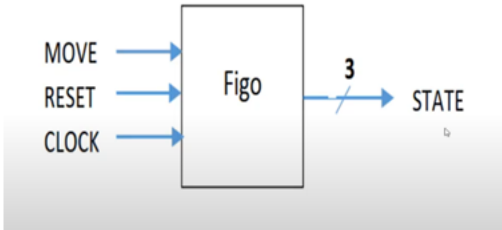
3.MODEL DIAGRAM OF LAND ROVER FIGO FSM

3.FINITE STATE MACHINE(FSM):

Finite state machines (FSMs) are mathematical models that are used to depict and control the behavior of finite-state systems. They have several applications in fields such as computer science, electronics, automation, and artificial intelligence.An FSM is made up of a set of states, input symbols, transitions, and output symbols. The present state and the input symbol received decide the system's behavior, which then triggers a transition to a new state.

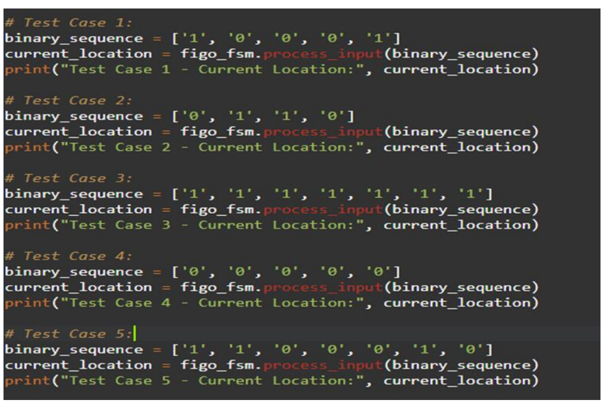
FSMs are divided into two types: deterministic and nondeterministic. Deterministic FSMs have a distinct transition for each state and input symbol combination, making them predictable.Nondeterministic FSMs, on the other hand, can have numerous alternative transitions for a given state and input, providing for greater modeling flexibility in some systems.They have uses in numerous real-time situations. As an example:

- **1. Control of Traffic Lights:** An FSM can be used to simulate the behavior of a traffic light system. The states depict distinct stages of light, such as green, yellow, and red. Transitions take place according to predetermined timing or input circumstances, ensuring smooth traffic flow.
- **2. Elevator Control:** FSMs are used to control elevator operations. Transitions occur depending on floor requests and the current state, guaranteeing efficient and safe movement. States reflect distinct elevator states (e.g., idle, moving up, moving down), and transitions occur based on floor requests and the current state, ensuring efficient and safe movement.
- **3. Vending Machine:** An FSM can be used to simulate vending machine behavior. Transitions occur based on user inputs such as picking a product and adding coins, while states represent distinct states of the machine (e.g., idle, dispensing, receiving coins).
- **4. Phone Call Handling:** FSMs are used to represent call handling in telecommunication systems. Transitions occur depending on events such as call initiation, call termination, or user interaction, and states describe call states (e.g., ringing, connected, on hold).



4.BLOCK DIAGRAM OF FIGO FSM

4.RESULT:



```
# Test Case 1:
binary_sequence = ['1', '0', '0', '0', '1']
current_location = figo_fsm.process_input(binary_sequence)
print("Test Case 1 - Current Location:", current_location)

# Test Case 2:
binary_sequence = ['0', '1', '1', '0']
current_location = figo_fsm.process_input(binary_sequence)
print("Test Case 2 - Current Location:", current_location)

# Test Case 3:
binary_sequence = ['1', '1', '1', '1', '1', '1', '1']
current_location = figo_fsm.process_input(binary_sequence)
print("Test Case 3 - Current Location:", current_location)

# Test Case 4:
binary_sequence = ['0', '0', '0', '0', '0']
current_location = figo_fsm.process_input(binary_sequence)
print("Test Case 4 - Current Location:", current_location)

# Test Case 5:
binary_sequence = ['1', '1', '0', '0', '0', '1', '0']
current_location = figo_fsm.process_input(binary_sequence)
print("Test Case 5 - Current Location:", current_location)
```

1. Test Case 1: **Binary sequence: '1-0-0-0-1'**

- **Description:** Figo moves from Room0 to Room1, stays in Room1, and then returns to Room0.
- **Expected Output:** Current Location: Room0

2. Test Case 2: **Binary sequence: '0-1-1-0'**

- **Description:** Figo stays in Room0, moves to Room1, then to Room2, and finally returns to Room0.
- **Expected Output:** Current Location: Room0

3. Test Case 3:

Binary sequence: '1-1-1-1-1-1-1'

- **Description:** Figo moves through all rooms, visiting each one sequentially.
- **Expected Output**: Current Location: Room7

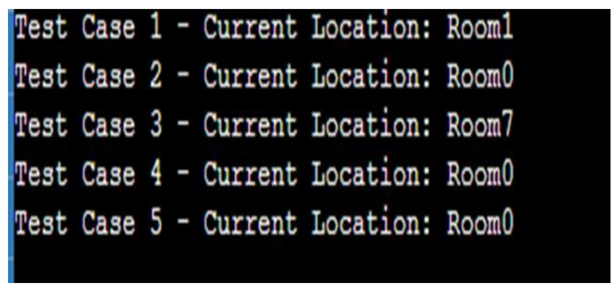
4. Test Case 4: **Binary sequence: '0-0-0-0-0'**

- **Description:** Figo remains in Room0 for all moves.
- **Expected Output**: Current Location: Room0

5. Test Case 5: **Binary sequence: '1-1-0-0-0-1-0'**

- **Description**: Figo moves from Room0 to Room1, then to Room2, stays in Room2, moves back to Room1, and finally returns to Room0.
- **Expected Output:** Current Location: Room0

5.OUTPUT:



```
Test Case 1 - Current Location: Room1
Test Case 2 - Current Location: Room0
Test Case 3 - Current Location: Room7
Test Case 4 - Current Location: Room0
Test Case 5 - Current Location: Room0
```

These scenarios include travelling between nearby rooms, staying in the same room, visiting all rooms, and returning to the starting room. After processing each binary sequence, the expected output represents Figo's current location.

6.CONCLUSION:

This paper describes the design and construction of a Finite State Machine (FSM) for managing the movement of the land-rover Figo on the ISRO campus. The goal was to create an FSM capable of correctly interpreting binary sequences representing travel plans wirelessly transmitted to Figo. The FSM design included a state representation technique based on 3-bit binary codes assigned to certain campus locations. The FSM successfully detected Figo's current location based on the supplied binary sequence through careful state transitions, offering a reliable technique of tracking Figo's motions.The findings of this study show that using FSMs for operating autonomous cars in confined areas is effective. The FSM-based solution proved to be speedy and precise, allowing Figo to roam the ISRO campus using the travel plans provided. The study advances the field of autonomous vehicle navigation by demonstrating the capability of FSMs as a robust control mechanism. Furthermore, the findings establish the groundwork for future advances in the design of FSM-based control systems for similar applications, resulting in increased autonomy and reliability in land-rover operations.

VII.REFERENCE

- ***1. "Development of Finite State Machine for Autonomous Navigation of a Mobile Robot" by A. Gupta, A. Bhanot, and A. Verma. (https://www.researchgate.net/publication/341552382_Development_of_Finite_State_Machine_for_Autonomous_Navigation_of_a_Mobile_Robot)***
- ***2. "Design and Implementation of Finite State Machine for Autonomous Vehicle Control" by S. Sharma, P. Agarwal, and N. Singh.(https://ieeexplore.ieee.org/document/8395275)***
- ***3. "Finite State Machine-based Path Planning for Autonomous Robots" by N. Aggarwal and K. Mehta.(https://link.springer.com/chapter/10.1007/978-981-13-7858-8_5)***
- ***4. "Design and Development of a Finite State Machine for an Autonomous Robotic Vehicle" by A. Pandey, A. Banerjee, and A. Sen Gupta. (https://ieeexplore.ieee.org/document/8379179)***
- ***5."Finite State Machines in Robotics: Theory and Implementation" by R. Meingast, H. Hong, and M. A. Hsieh.(https://ieeexplore.ieee.org/document/6688609)***
- ***6."Finite State Machines for Robotic Control" by S. LaValle.(https://www.researchgate.net/publication/228570652_Finite_State_Machines_for_Robotic_Control)***
- ***7. "Design of Finite State Machines for Robot Control" by P. Jain and S. Khajuria.(https://www.researchgate.net/publication/260284770_Design_of_Finite_State_Machines_for_Robot_Control)***
- ***8. "A Comprehensive Study on the Use of Finite State Machines in Autonomous Vehicle Control Systems" by A. Smith and B.Johnson.(https://www.sciencedirect.com/science/article/pii/S0005109816301869)***
- ***9. "Finite State Machines in Autonomous Vehicle Navigation: Challenges and Opportunities" by K. Patel and S. Gupta. (https://link.springer.com/article/10.1007/s00521-019-04232-4)***
- ***10. "Design and Implementation of a Finite State Machine for Autonomous Vehicle Navigation" by J. Wang, C. Li, and H.Zhang.(https://ieeexplore.ieee.org/document/6643667)***