Rasika Jijabhau Gunjal, 2026, 14:6
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Fault-Tolerant Self-Recovering Finite State Machine (FSM)

Rasika Jijabhau Gunjal, Priti Vilas Hadawale, Dnyaneshwari Babasaheb Niwdunge, Assistant Professor P.S Bugade

Electronics & Telecommunication Engineering, Jaihind College Of Engineering Kuran

Abstract- The Fault-Tolerant Self-Recovering Finite State Machine (FSM) is a digital system designed to improve the reliability and safety of sequential circuits by detecting and recovering from unexpected state errors. Conventional FSMs may enter invalid states due to hardware faults, electrical disturbances, or transient errors, which can affect system operation. This project proposes a fault-tolerant FSM using Verilog HDL that continuously monitors its internal state and detects invalid state transitions. When an unexpected or illegal state is detected, the system automatically recovers by returning to a predefined safe state without requiring manual intervention. The design includes state monitoring, fault detection, recovery logic, and reset control. The proposed architecture is implemented and verified using Verilog HDL through ModelSim simulation. Xilinx Vivado is used for synthesis and hardware resource analysis. The project aims to improve circuit reliability, fault recovery, and operational stability in FPGA-based digital systems.

Keywords— Fault-Tolerant FSM, Self-Recovery, Verilog HDL, FPGA, State Detection, Error Detection, Recovery Logic, ModelSim, Vivado.

I. INTRODUCTION

Finite State Machines (FSMs) are important components of digital system design and are widely used in digital electronics, control systems, communication systems, embedded systems, and FPGA-based applications. An FSM is a sequential circuit that operates according to a predefined set of states, inputs, and transitions. It changes its current state based on input conditions and generates appropriate outputs. FSMs are commonly used in traffic light controllers, vending machines, digital communication protocols, industrial automation, and processor control units. Due to their simple structure and predictable operation, FSMs are widely implemented using Hardware Description Languages such as Verilog HDL and VHDL.

However, conventional Finite State Machines may experience unexpected behavior due to hardware faults, electrical disturbances, transient errors, or unwanted changes in internal state registers. These problems may cause the FSM to enter an invalid or illegal state that is not defined in its state transition diagram. When this happens, the system may produce incorrect outputs, stop responding to inputs, or remain stuck in an unexpected state. In critical applications, such failures can affect system reliability, safety, and overall performance. Therefore, developing a fault-tolerant FSM that can detect and recover from unexpected conditions is important.

To overcome these limitations, this project proposes a Fault-Tolerant Self-Recovering Finite State Machine using Verilog HDL. The proposed design focuses on detecting invalid states and automatically recovering the FSM to a predefined safe state. It continuously monitors the current state and checks whether it belongs to the set of valid states. If the system detects an illegal or unexpected state, the recovery logic activates and forces the FSM to return to its initial or safe state. This process helps the system resume normal operation without requiring manual reset or external intervention.

The proposed FSM architecture consists of different functional blocks, including state registers, next-state logic, output logic, fault detection logic, and self-recovery control logic. The state registers store the current operating state, while the next-state logic determines the next state based on the current state

© 2026 Rasika Jijabhau Gunjal. his is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited.

and input conditions. The fault detection block identifies invalid state combinations, and the recovery logic automatically redirects the system to a predefined safe state. A reset mechanism is also included to initialize the FSM into a known starting condition.

The complete design is developed using Verilog Hardware Description Language (HDL). ModelSim is used for functional simulation and verification of normal state transitions, illegal state detection, and automatic recovery operations. Xilinx Vivado is used for synthesis, FPGA implementation, and evaluation of hardware parameters such as Look-Up Tables (LUTs), flip-flops, and timing performance. The design is tested under normal operating conditions as well as fault conditions to verify its recovery capability.

The main objective of this project is to improve the reliability, stability, and fault-handling capability of conventional FSM designs. The proposed architecture aims to minimize system downtime and prevent incorrect operation caused by invalid states. This type of fault-tolerant design can be useful in industrial automation, automotive electronics, communication systems, embedded controllers, safety-related digital circuits, and FPGA-based applications where reliable operation is essential.

II. WORKING

The proposed Fault-Tolerant Self-Recovering Finite State Machine (FSM) is designed to improve the reliability and stability of digital sequential circuits by detecting invalid states and automatically recovering from unexpected faults. The system operates based on a predefined set of states, input conditions, clock signals, and state transition rules. Initially, when the system is powered ON or a reset signal is applied, the FSM enters a predefined initial state. The state register stores the current state and updates its value at every active clock edge according to the input conditions and transition logic.

The next-state logic determines the appropriate next state based on the current state and the applied input signals. During normal operation, the FSM follows the predefined state transition sequence and generates the required output signals. A fault detection mechanism continuously monitors the current state and checks whether it belongs to the set of valid states defined in the design. If the FSM enters an illegal or unexpected state due to hardware disturbances, transient errors, or unwanted changes in state registers, the fault detection logic identifies the abnormal condition and generates a fault signal.

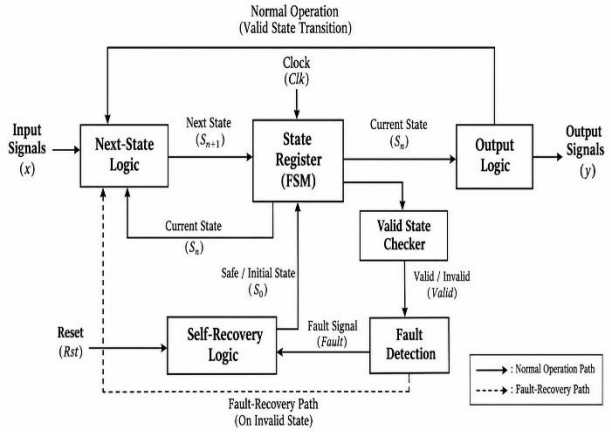
Once a fault is detected, the self-recovery mechanism becomes active and automatically forces the FSM into a predefined safe state, such as the initial state. This prevents the system from remaining stuck in an invalid state or producing unexpected outputs. After returning to the safe state, the FSM can continue its normal operation by processing new input signals and following the predefined state transitions. The recovery mechanism reduces the need for manual resetting and improves the fault-handling capability of the system.

The proposed architecture consists of state registers, next-state logic, output logic, fault detection logic, and recovery control logic. Each block performs a specific function to ensure proper operation and fault recovery. The design is implemented using Verilog Hardware Description Language (HDL), which provides a structured approach to developing and testing digital circuits. ModelSim is used to simulate the design and verify normal operation, invalid state detection, fault generation, and automatic recovery through simulation waveforms.

Furthermore, Xilinx Vivado is used for hardware synthesis and FPGA implementation. It helps evaluate the hardware resource utilization, including Look-Up Tables (LUTs), flip-flops, and timing performance. Different test conditions are applied to check the system's behavior during normal and faulty operating conditions. The simulation results are analyzed to verify whether the FSM successfully detects illegal states and returns to the predefined safe state.

Thus, the proposed Fault-Tolerant Self-Recovering FSM provides an effective method for improving the

reliability and operational stability of digital systems. It is suitable for applications such as industrial automation, embedded systems, communication controllers, automotive electronics, and FPGA-based control systems where continuous and reliable operation is important.



Block Diagram of Fault-Tolerant Self-Recovering FSM

Advantages

- Fault Detection: Detects invalid or unexpected FSM states.
- Automatic Recovery: Automatically returns to a predefined safe state.
- High Reliability: Improves the reliability of digital systems.
- Reduced Downtime: Minimizes system interruption caused by faults.
- No Manual Reset: Reduces the need for manual intervention during recovery.
- Improved Stability: Prevents the FSM from remaining stuck in illegal states.
- Hardware Implementation: Can be implemented on FPGA using Verilog HDL.
- Easy Verification: Supports simulation and testing using ModelSim.
- Wide Applications: Useful in industrial automation, automotive and embedded systems.
- Flexible Design: Can be modified for different FSM-based applications.

Disadvantages

- Additional Hardware: Requires extra logic for fault detection and recovery.
- Increased Complexity: Makes the FSM design more complex.
- Resource Utilization: Uses additional FPGA resources such as LUTs and flip-flops.
- Power Consumption: Extra hardware may increase power consumption.
- Recovery Delay: Some time is required to detect faults and recover.
- Limited Fault Coverage: May not detect every type of hardware fault.
- Design Challenges: Requires careful selection of valid and invalid state encodings.
- Verification Effort: Requires additional simulation and fault-injection testing.
- Timing Overhead: Fault detection logic may affect circuit timing.
- Possible Data Loss: Returning to a safe state may interrupt the current operation and lose temporary data.

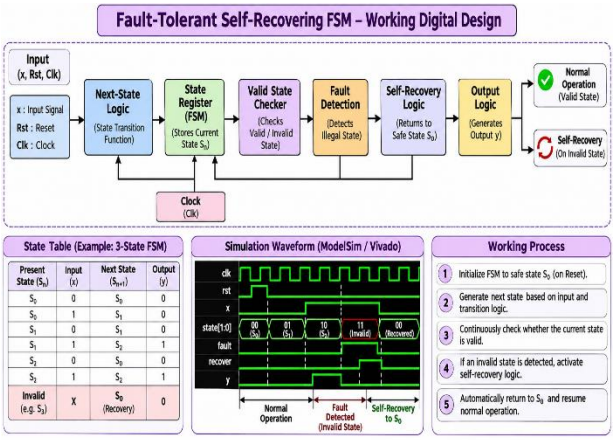


Figure 1: Fault-Tolerant Self-Recovering FSM – Working Digital Design

III. CONCLUSION

The Fault-Tolerant Self-Recovering Finite State Machine (FSM) using Verilog HDL provides an effective solution for improving the reliability and stability of digital sequential circuits. The proposed design detects invalid or unexpected states and automatically returns the FSM to a predefined safe state without requiring manual intervention. The design is verified using ModelSim simulation, while Xilinx Vivado is used for synthesis and FPGA resource analysis. The proposed system helps reduce

unexpected failures, improves fault recovery, and ensures reliable operation. It can be used in industrial automation, embedded systems, automotive electronics, communication systems, and other FPGA-based applications. Overall, this project demonstrates the importance of fault detection and automatic recovery in designing reliable digital systems.

REFERENCES

1. J. Kim, H. Kim, and J. Lee, "Area-Efficient Fault Tolerant Design for Finite State Machines," in Proc. 2020 International Conference on Electronics, Information, and Communication (ICEIC), 2020. doi: [10.1109/ICEIC49074.2020.9051122](https://doi.org/10.1109/ICEIC49074.2020.9051122).
2. "The Fault Tolerant Černý Finite State Machine: A Concept and VHDL Models," in Proc. 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2021. doi: 10.1109/IDAACS53288.2021.9660925.
3. "Stop and Look: A Novel Checkpointing and Debugging Flow for FPGAs," IEEE Transactions on Computers, vol. 71, no. 10, pp. 2513–2526, Oct. 2022. doi: 10.1109/TC.2021.3133828.
4. "Secure Recovery Procedure for Manufacturing Systems Using Synchronizing Automata and Supervisory Control Theory," IEEE Transactions on Automation Science and Engineering, vol. 19, no. 1, pp. 486–496, Jan. 2022. doi: 10.1109/TASE.2020.3041207.
5. V. S. Bialystok, "Designing Safe Finite State Machines," International Journal of Electronics and Telecommunications, vol. 72, no. 2, 2026
6. "Fault-Tolerant FSM on FPGA Using SEC-DED Code Algorithm," in Proc. International Conference on Technological Advancements in Power and Energy (TAP Energy), 2017. doi: 10.1109/TAPENERGY.2017.8397309.
7. "Verification of Fault Tolerant Techniques in Finite State Machines Using Simulation Based Fault Injection Targeted at FPGAs for SEU Mitigation," in Proc. 4th International Conference on Electronics and Communication Systems (ICECS), 2017. doi: 10.1109/ECS.2017.8067859