Design and Implementation of a Finite State Machine for Robotic Arm Control

Zubair Ibn Kabir
Electrical and Electronic Engineering
Ahsanullah University of Science & Technology
141,142 ,Love Road,Tejgaon Industrial Area, Dhaka-1208
zubair.eee.20200205148@aust.edu

Angshuman Aumi
Electrical and Electronic Engineering
Ahsanullah University of Science & Technology
141,142 ,Love Road,Tejgaon Industrial Area, Dhaka-1208
angshuman.eee.20200205134@aust.edu

Md. Raihan Mahmud
Electrical and Electronic Engineering
Ahsanullah University of Science & Technology
141,142 ,Love Road,Tejgaon Industrial Area, Dhaka-1208
raihan.eee.20200205157@aust.edu

Navid Aziz
Electrical and Electronic Engineering
Ahsanullah University of Science & Technology
141,142 ,Love Road,Tejgaon Industrial Area, Dhaka-1208
navid.eee.20200205160@aust.edu

Abstract*— *This project presents the design, simulation, and implementation of a Finite State Machine (FSM) to control a robotic arm using a hardware description language (HDL). The FSM is designed to handle object detection, gripping, sorting based on weight, and emergency stops. The FSM was implemented in Verilog and verified through simulation using ModelSim. The synthesized netlist was generated using Cadence Genus, and the full chip layout was completed using Cadence Innovus. The system was optimized for power, area, and timing constraints to ensure efficient operation. The results confirm the successful implementation of a reliable FSM-based robotic arm controller suitable for industrial automation.

Keywords: *Finite State Machine (FSM), Robotic Arm, Verilog, ModelSim, RTL, Gate-level Netlist, Cadence Genus, Cadence Innovus, Place and Route (PnR), Synthesis, Testbench Simulation.*

.

I. Introduction

Automated robotic systems are widely used in manufacturing and industrial applications to enhance efficiency and precision. A well-structured control mechanism is crucial for robotic arms to perform tasks such as object detection, gripping, sorting, and placing. The Finite State Machine (FSM) approach provides a systematic method to define the sequential operations of a robotic arm, ensuring that each movement is accurately controlled based on sensor inputs.

This project presents the design and implementation of an FSM-based robotic arm controller. The FSM receives input signals from sensors and generates corresponding output commands to control the arm's movement and gripping function. The project aims to create a robust control system that efficiently handles different scenarios, including emergency stop functionality.

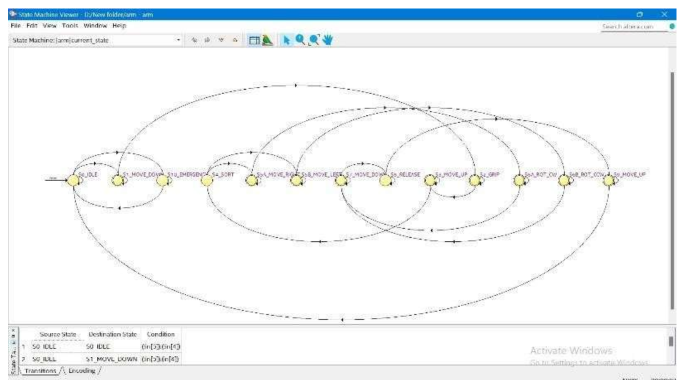


Fig: FSM State Diagram for Robotic Arm Control

II. Methodology

The FSM design for robotic arm control follows three major phases:

***1. RTL Simulation**:* A. The FSM was designed in Verilog and simulated using ModelSim.
B. A testbench was created to verify correct state transitions and output signals.
2. *Synthesis using Cadence Genus*: A. The Verilog design was synthesized to generate a gate-level netlist.
B.Design constraints were applied to optimize power and area.
*3. **Physical Design using Cadence Innovus:** A.* The synthesized design was placed and routed using Cadence Innovus.
B. Major steps included:
- Floor planning
- Power planning
- Standard cell placement
- Clock tree synthesis (CTS)
- Routing and Design Rule Check (DRC) validation
- Resolving connectivity violation

III. FSM Design and State Transitions

The FSM controls the robotic arm based on sensor inputs. The key states include:

1. S0_IDLE: The system waits for an object detection signal.
2. S1_MOVE_DOWN: The arm moves down to pick up an object.
3. S2_GRIP: The gripper closes around the object.
4. S3_MOVE_UP: The arm lifts the object.
5. S4_SORT: The FSM decides the sorting path based on the weight sensor.
6. S5A_MOVE_RIGHT (Light Object) / S5B_MOVE_LEFT (Heavy Object): The arm moves to the sorting area.
7. S6A_ROT_CW / S6B_ROT_CCW: The arm rotates accordingly.
8. S7_MOVE_DOWN: The arm lowers the object to its designated place.
9. S8_RELEASE: The gripper opens to release the object.
10. S9_MOVE_UP: The arm returns to its initial position.
11. S10_EMERGENCY: All movements stop if the emergency signal is triggered

IV. Result Analysis

Parameters	Value
DESIGN DENSITY	100%
AREA	1.0260 mm²
INTERNAL POWER	28.301%
SWITCHING POWER	71.53%
LEAKAGE POWER	7.28%
DRC VIOLATIONS	0
CONNECTIVITY VIOLATIONS	0
OTHER VIOLATIONS	0

V. Conclusion

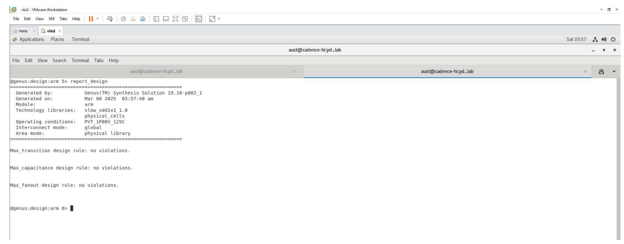
This project successfully designed and implemented an FSM-based robotic arm controller. The FSM efficiently handled object detection, gripping, sorting, and emergency stops through defined state transitions. The design was optimized for power, area, and timing using Cadence tools, ensuring reliable and efficient operation. Future enhancements may include optimizing sorting speed and integrating machine learning for adaptive decision-making.

VI. Pictorial Representation

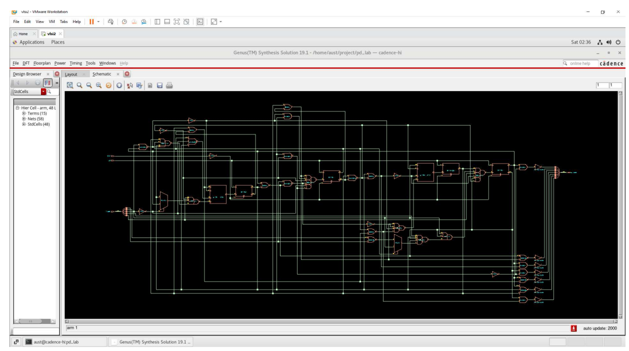
The following images provide a visual representation of the design process:

Modelsim Simulation:

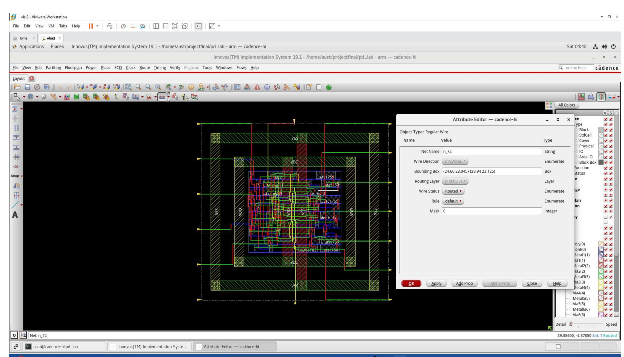
Synthesis Result:



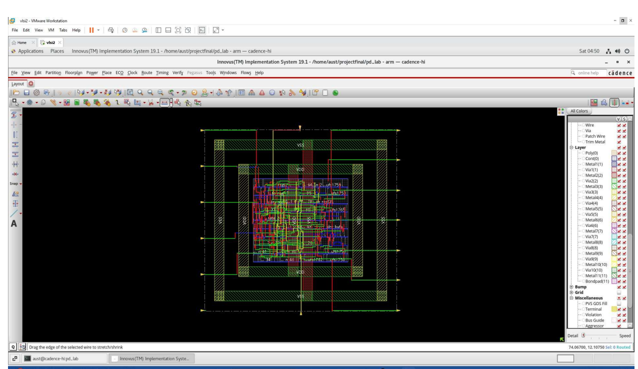
Schematic:



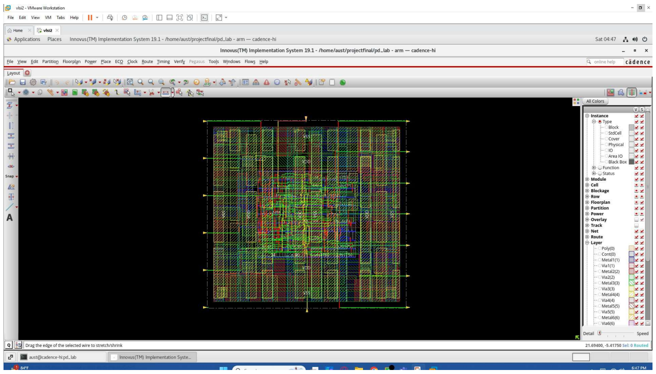
Route Confirmation:



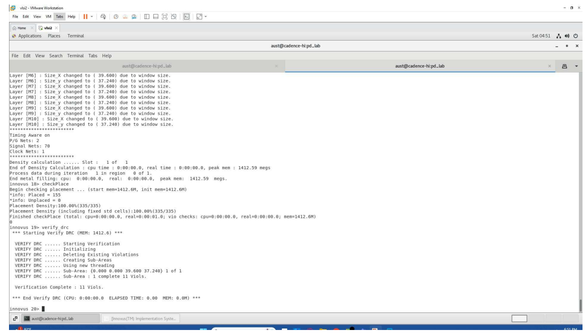
Before Metal Fill:



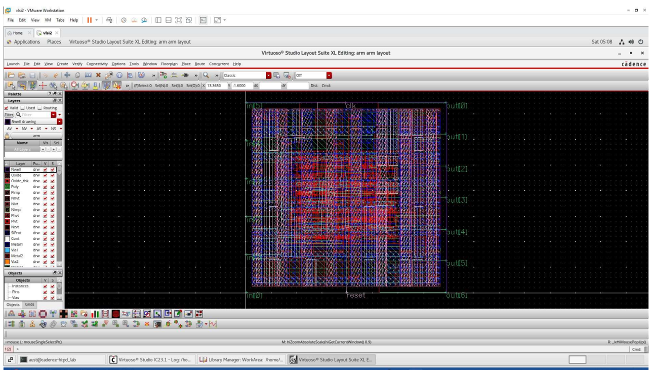
After Metal Fill & Filler Cell:



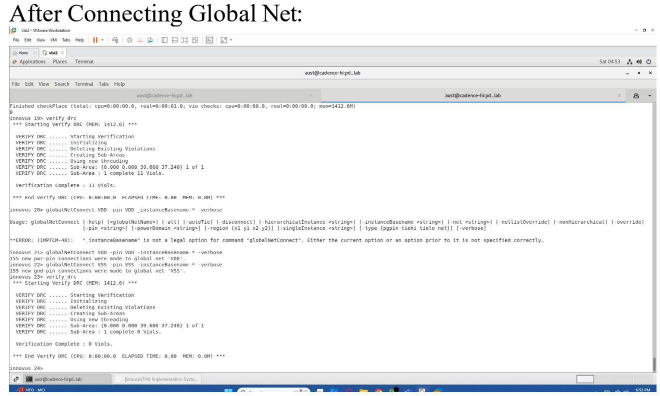
Initial DRC Check:



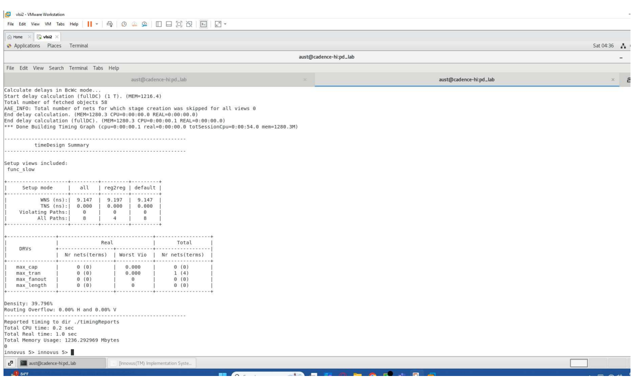
Layout:



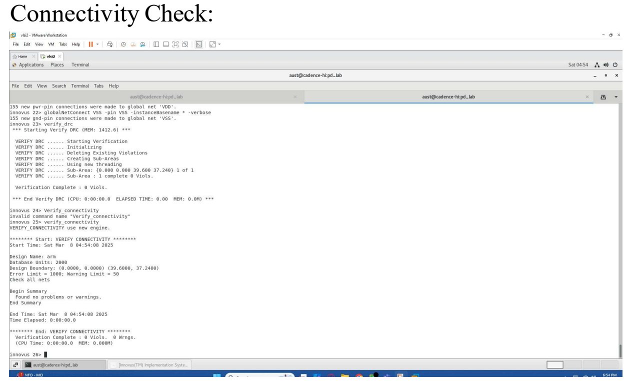
After Connecting Global Net:



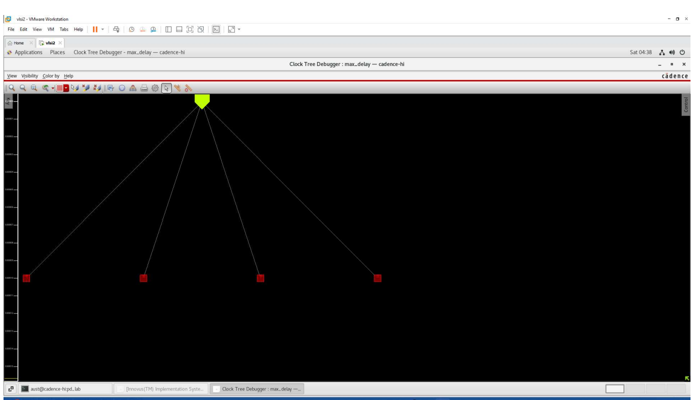
Pre-CTS Setup Analysis:



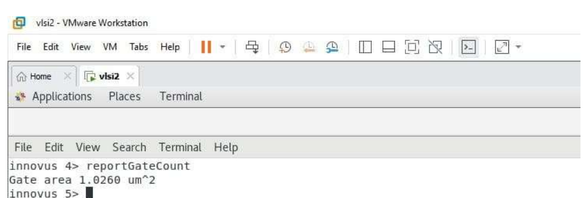
Connectivity Check:



CTS:



Gate Count And Area



Total Power