COL215 Lab Assignment 8: FPGA Car Game (Part I)

Shivanshu Aryan (2024CS10237), Arnav Meena (2024CS50218)

November 2025

1 Introduction

This report documents the design, simulation, FPGA resource utilization, and hardware verification for a car game implemented on Xilinx Artix-7 (Basys3). Key tasks included moving the car horizontally based on push buttons, collision detection, FSM design for control logic, and end-to-end validation using Vivado and on-board experimentation.

2 Design Decisions

The following core decisions were made:

- **Finite State Machine (FSM)** is used to control car movement and collision logic, with states for start, movement, collision, idle, and reset. The FSM comprises five main states:
  1. **START**: Initial state, car at fixed position.
  2. **RIGHT_CAR/LEFT_CAR**: Car moving horizontally as per button input.
  3. **COLLIDE**: Car hits boundary (collision triggers end).
  4. **IDLE**: No button pressed.
  5. Game restarts on reset button (`BTNC`).
- **Pixel logic** uses a rectangular hit box (14x16 pixels), enabling fast manipulation on hardware.
- **Car movement** occurs continuously while the button is pressed, with per-pixel granularity chosen for visual smoothness.
- Push buttons `BTNL`, `BTNR` (left/right movement) and `BTNC` (reset) are used for control.

- Collision is detected when car's box exceeds pixel road boundaries:

$$car\_top\_left\_x < 244 \qquad (leftboundary)$$

$$car\_top\_right\_x > 318 \quad or \quad car\_top\_left\_x{+}14 > 318 \qquad (rightboundary)$$

3 FSM Diagram

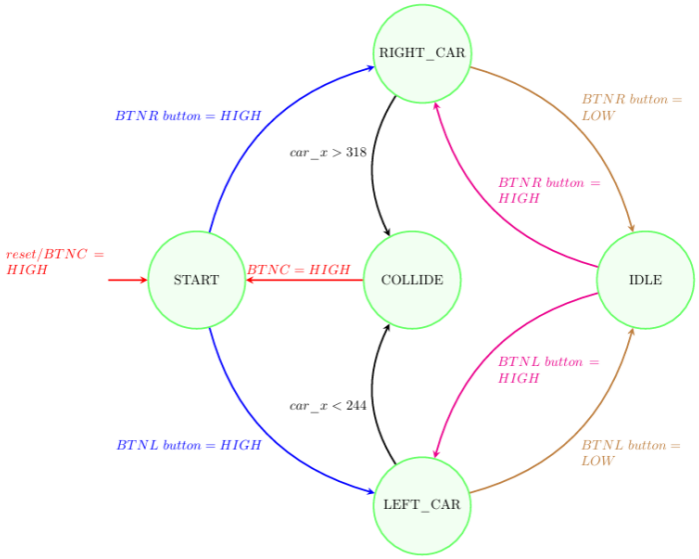

Figure 1: The State Diagram

4 Simulation

- Simulation was performed for all major states and transitions.

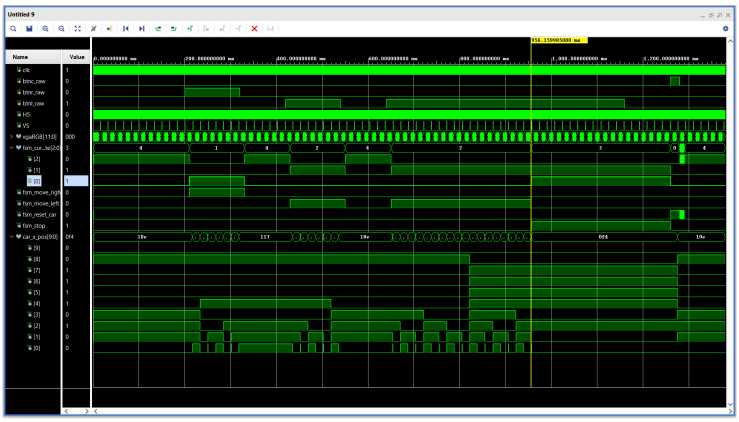

Figure 2: Simulation for all relevant states

5 FPGA Synthesis and Implementation Report

The design was synthesized for Artix-7 (xc7a35tcpg236-1), targeting efficiency and correct functionality. Below is the utilization summary:

Resource Type	**Used**	**Available**	**Util. (%)**
Slice LUTs (Logic)	240	20,800	1.15
Slice LUTs (Memory)	0	9,600	0.00
Slice Registers (FFs)	194	41,600	0.47
Block RAM Tiles	14	50	28.00
RAMB36E1/FIFO	12	50	24.00
RAMB18E1	4	100	4.00
DSP48E1 (DSPs)	2	90	2.22
Bonded IOBs (IO Pins)	18	106	16.98
BUFGCTRL (Clock Buffers)	2	32	6.25

Table 1: FPGA resource utilization (implementation)

6 Generated Schematics

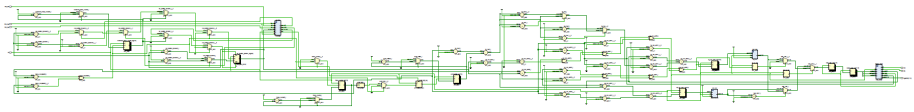

Figure 3: RTL Schematic

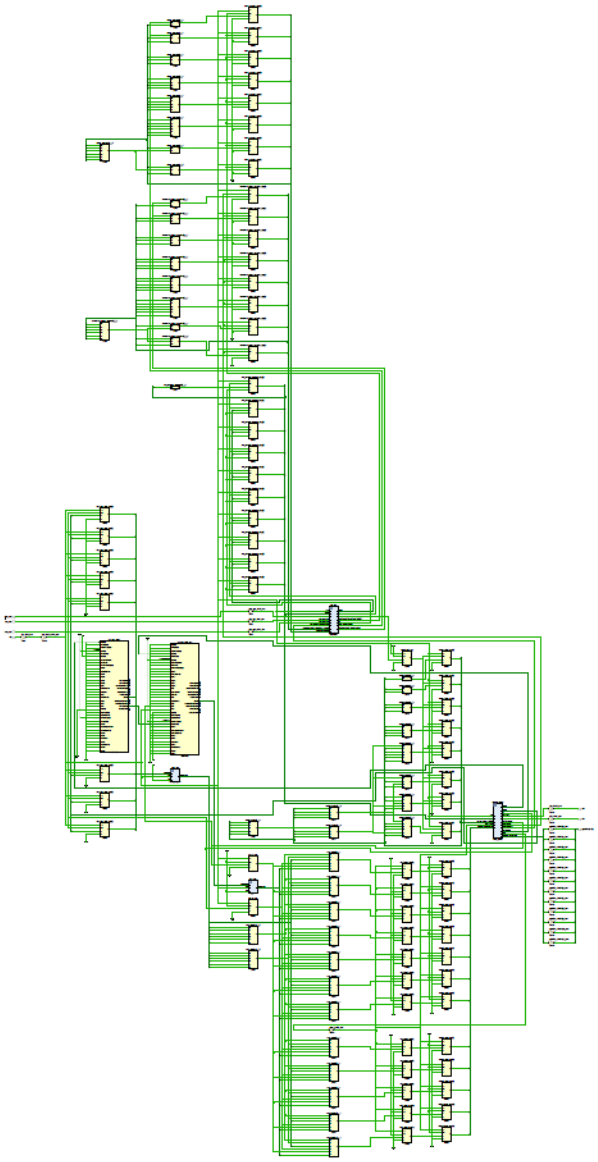

Figure 4: Synthesis Schematic

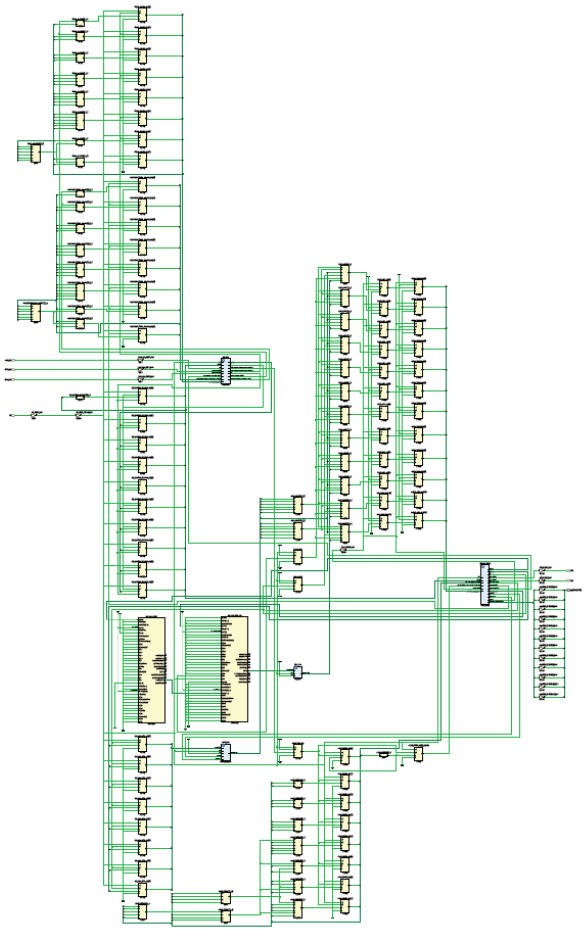

Figure 5: Implementation Schematic

7 Conclusion

This FPGA implementation achieves real-time car control, collision detection, and hardware verification with efficient resource utilization. The FSM-based approach and hit box logic simplify pixel operations. Simulation and real hardware results match expected behavior, validating the full design.