FPGA-Based Bidirectional Motor Control System Using Verilog

1. Project Overview

This project demonstrates a bidirectional motor control system implemented using Verilog HDL and deployed on an FPGA platform.

The system controls the direction of a DC motor - upward or downward - based on two input signals.

An H-bridge motor driver is used to control the motor, with the logic signals generated by the FPGA.

2. Objectives

- Design a Verilog-based system for motor direction control.
- Simulate the design using standard tools and verify its correctness.
- Implement the system on an FPGA development board.
- Test and validate the control logic with real hardware.
- Document the process and share the results through GitHub and a video demonstration.

3. System Design

Functional Description:

- up = 1, down = 0 -> Upward rotation
- up = 0, down = 1 -> Downward rotation
- Other cases -> Stop motor

4. Verilog Code

```
module motor_control (
    input clk,
    input up,
```

```
    input down,
    output reg motor_a,
    output reg motor_b
);
always @(posedge clk) begin
    if (up && !down) begin
        motor_a <= 1;
        motor_b <= 0;
    end
    else if (!up && down) begin
        motor_a <= 0;
        motor_b <= 1;
    end
    else begin
        motor_a <= 0;
        motor_b <= 0;
    end
end
endmodule
```

5. Simulation and Testing

Testbench (motor_control_tb.v) simulates all motor states.

Simulation verified successful transitions and output responses.

6. Hardware Implementation

Components:

- FPGA Board (e.g., Artix-7)

- L293D Motor Driver

- DC Motor

- Push buttons

- Power supply

7. Documentation and GitHub

Repository includes code, simulation results, block diagrams, and demo video.

8. Demonstration Video

Demo shows motor control using push buttons connected to FPGA.

9. Deliverables Summary

- Verilog Code v

- Simulation v

- Hardware Test v

- GitHub Documentation v

- Demo Video v

10. Conclusion

A simple and effective demonstration of motor control using Verilog and FPGA, applicable in automation and embedded systems.