Project Concept Draft:

Traffic Light Controller

for a Street

with Two Lanes and a Pedestrian Walkway using FPGAs

Team Valnesta
HW Engineering SoSe 24

Onyesi John Abiagam
Nnaemeka Valentine Eze
Manoj Luitel

1. **Introduction**

1.1 Motivation

FPGA and VHDL Motivation:

Field Programmable Gate Arrays (FPGAs) offer a high level of flexibility, performance, and parallelism that is unmatched by traditional microcontrollers. By using VHDL (VHSIC Hardware Description Language), designers can describe the hardware behavior at a very detailed level, which allows for precise control over timing and resource utilization. This makes FPGAs an excellent choice for implementing a traffic light controller where processing and reliability are critical.

Traffic Light Controller Motivation:

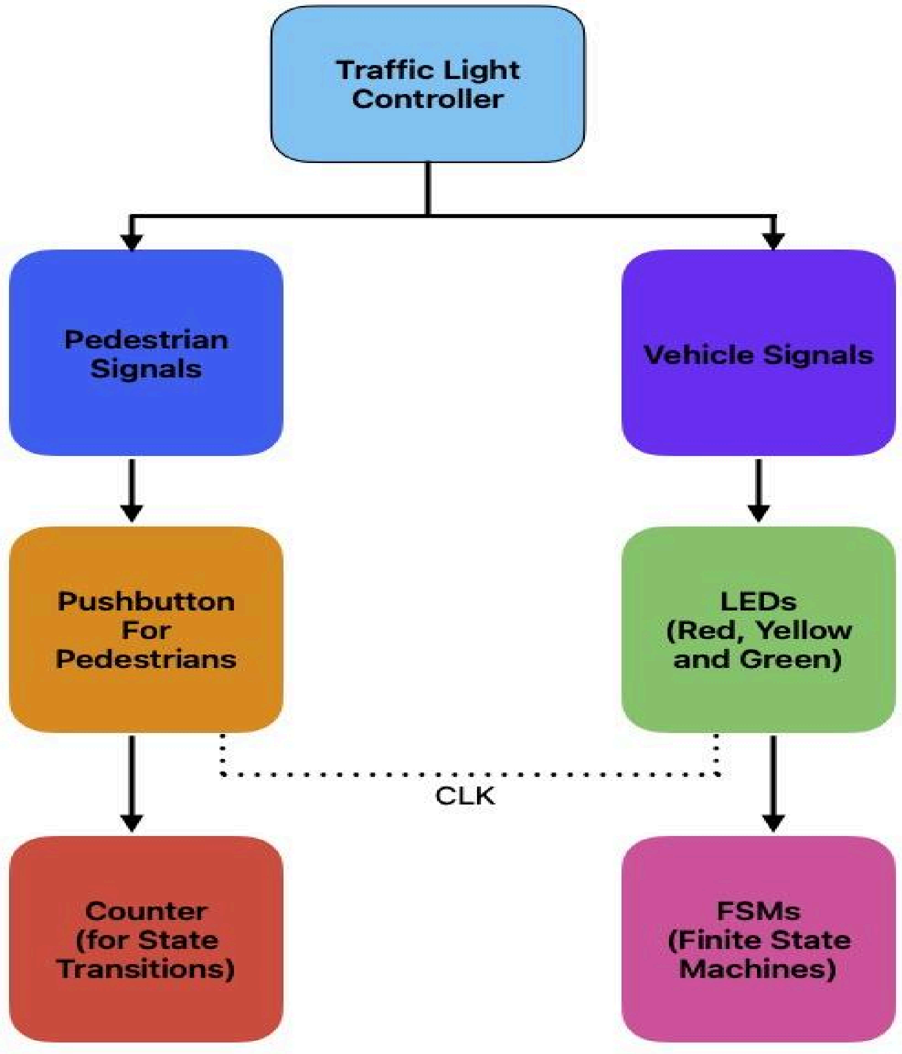
A traffic light controller is essential for managing traffic flow and ensuring pedestrian safety on the streets. By implementing the controller on an FPGA, we can leverage the parallel processing capabilities to handle multiple traffic lights and pedestrian signals simultaneously, ensuring timely and efficient transitions between states. This approach also allows for easy updates and scalability, making it a robust solution for modern traffic management systems.

2. Concept Description

The project involves designing a traffic light controller for a street with two lanes and a pedestrian walkway. The system will manage traffic lights for vehicles and pedestrian signals. The controller will have three main states: Initial, Waiting, and Green (for pedestrian). The system will include a push-button for pedestrians, a clock signal (clk), LEDs to represent traffic lights, and counters to manage timing for each state.

3. Block Diagram

Below is the block diagram representing the components and their interactions within the traffic light controller system:


Traffic Light Controller
Pedestrian Signals
Vehicle Signals
Pushbutton For Pedestrians
LEDs (Red, Yellow and Green)
CLK
Counter (for State Transitions)
FSMs (Finite State Machines)

Detailed Description

1. **Pushbutton**: A button for pedestrians to request crossing. When pressed, it triggers a transition in the FSM (Finite State Machine).
2. **Clock (clk)**: Provides the timing reference for the system. All state transitions and timing operations are synchronized to this clock.
3. **LEDs**: Represent the traffic signals. There will be separate LEDs for vehicle signals (Red, Yellow, Green) and pedestrian signals (Walk, Don't Walk).
4. **Counter**: Manages the duration of each state. It counts clock pulses to determine how long the system should stay in each state before transitioning.
5. **Finite State Machine (FSM)**: Controls the state transitions. It has three states:
    - **Initial State**: Default state where the vehicle light is green and pedestrian light is "Don't Walk" (Red).
    - **Waiting State**: Activated when the pedestrian pushbutton is pressed. The vehicle light changes to yellow, preparing to stop.
    - **Green State (for pedestrians)**: Pedestrian light changes to "Walk", allowing pedestrians to cross. The vehicle light remains red.
6. **Pedestrian Signals**: Indicate to pedestrians when it is safe to cross (Walk) and when they should wait (Don't Walk).
7. **Vehicle Signals**: Control the flow of vehicular traffic with Red, Yellow, and Green lights.

State Transitions

- **Initial State**: Vehicle lights are green, pedestrian lights are "Don't Walk".
    - **Transition to Waiting State**: Occurs when the pedestrian pushbutton is pressed.
- **Waiting State**: Vehicle lights change from green to yellow.
    - **Transition to Green State (for pedestrians)**: After a fixed duration, vehicle lights change to red, and pedestrian lights change to "Walk".
- **Green State (for pedestrians)**: Pedestrian lights are "Walk".
    - **Transition to Initial State**: After a fixed duration, pedestrian lights change to "Don't Walk", and vehicle lights change back to green.

Using VHDL to describe this behavior, we can implement a highly efficient and reliable traffic light controller on an FPGA, capable of handling traffic management and ensuring pedestrian safety.