# Project 3

Dual Traffic Light System with Pedestrian Crossing

CptS 466

Kevin Evans

## Requirements

### Overview

In this project, we will implement two traffic lights at an intersection with a pedestrian crossing. Two one-way streets will meet at an intersection and each traffic light will relate to the other, as well as the pedestrian stop/go light. This system will run on the LaunchPad board, connecting to PORT\_E, PORT\_F, and PORT\_B.

### Functional Description

The system will include two pairs of three LEDs: red (stop), yellow (slow down), and green (go). It will also use the on-board RGB led for pedestrian control: flashing red (stop/hurry) and green (walk). Three buttons will be used to represent the two car sensors and the pedestrian sensor.

### Deliverables

The deliverable will be the LaunchPad connected to a breadboard, populated with the necessary LEDs and switches, as defined in the functional description. The program will be written in C and compiled for the LaunchPad.

At the end, we will have a system with 8 LEDs (3 for two traffic signals, two for the go/stop pedestrian LED), 3 switched inputs (Main traffic, Spring traffic, Pedestrian).

## Design Document

The two stoplight LEDs will be connected to PB5:0, and the pedestrian LEDs will be connected to the on-board RGB LED PB2:0. The 3-bit (8 total) inputs will be given by PE2:0. This is shown in the data flow diagram below in Figure 1. We will use the 80 MHz clock for the timing of the system (for the delays, etc).

Diagram

Description automatically generated

Figure 1: The dataflow diagram of the traffic light system.

The system will implement the finite state machine, as described in Table 1 below. Appropriate delays were added to ensure the states to not change too quickly.

Table 1: the finite state machine in a state table format.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # | State Name | *Outputs* | | *Inputs ([ped][spring][main])* | | | | | | | |
| **Main/Spring** | **Ped** | **000** | **001** | **010** | **011** | **100** | **101** | **110** | **111** |
| 1 | goMain | 001100 | Red | goMain | goMain | waitMain | waitMain | waitMain | waitMain | waitMain | waitMain |
| 2 | waitMain | 010100 | Red | goSpring | goSpring | goSpring | goSpring | goPed | goPed | goPed | goPed |
| 3 | goSpring | 100001 | Red | goSpring | waitSpring | goSpring | waitSpring | waitSpring | waitSpring | waitSpring | waitSpring |
| 4 | waitSpring | 100010 | Red | goMain | goMain | goMain | goMain | goPed | goPed | goPed | goPed |
| 5 | goPed | 100100 | Green | hurryPed | hurryPed | hurryPed | hurryPed | goPed | goPed | goPed | goPed |
| 6 | hurryPed | 100100 | Flashing Red | goMain | goMain | goSpring | goSpring | goMain | goMain | goSpring | goSpring |

## Discussion

There are a few limitations to this design. First, the timing of the traffic lights are not ideal, as it may not give enough time for cars to pass through. There is currently a 5 second delay for the yellow light, which can be adjusted later on. Additionally, there should be a longer delay between the red light and the pedestrian go light, as some cars may have trouble stopping in time. This could lead to a dangerous situation where pedestrians can be struck by cars.