**ISP - The Traffic Light System**

By: Braulio Andres Carrion Corveira, Kirill Erkovich, Devin Lau, Carlo Luis V. Domingo

TEJ3M1

Mr. Fourmanovskis

June 11, 2018

**PART 1: PLAN**

**INTRODUCTION:**

This project is about a three-way traffic light system in a T-intersection which includes a pedestrian light, street light, and a parking lot with its gate. This traffic light system was created in order to meet the requirements of the project given and to apply our knowledge of what was learned in this computer engineering course. The three-way traffic light system was created by Garrett A. Morgan, who patented a T-shaped electrical system with an interim warning sign, much like today’s yellow light. He did this to ensure safe passage across the streets, as he observed too many collisions involving instant “stop” and “go” signals with no warnings for the drivers to react when the signals changed.

To create this system, a shoe box, coloured construction paper, hot glue and tape was used to design the T-intersection and parking lot of this project. Red, yellow, green and white LED lights were used to construct the traffic, street, and pedestrian lights. A gate is placed at the entrance of the parking lot. To control the traffic light system and parking lot gate, an arduino with a corresponding program, breadboard, wires, IR sensor, servo motor, and a pedestrian button are used. The LDR is used to turn on the street lights when it gets too dark.

**EXPLANATION:**

The three-way traffic light system has each LED stay on for three seconds unless the pedestrian button is pressed. The pedestrian button changes the corresponding perpendicular traffic lights to yellow then red and its pedestrian light to green if it was previously red. If the pedestrian light is green and the pedestrian button is pressed, the time limit for the current set of lights extends. The street lamp works using a light dependent resistor (LDR) that turns on the light when it is dark and turns off when the room around it is bright. The parking lot gate is closed by default, but when something is in-between its IR sensors, the servo motor opens up the gate for one second and then closes. All of these systems are made possible from the arduino and its programmed code connected to its breadboard and computer program used to run the code.

**SKETCHUP EXERCISE:**

**PART 4: FINAL REPORT**

**CHALLENGES IN GETTING SYSTEM TO WORK:**

To begin this project, we first had to wrap our heads around how the arduino worked. It was a challenge learning the new language, but with knowledge of other programming languages, such as C and Java, it helped us use related code to understand how the system worked. Once we wrapped our heads around the concept with input from fellow classmates, we were able to slowly work our way to the final product. Setting up the code and testing it with the LEDs and resistors was no easy feat. Next, being able to understand how each component works was not difficult, but it was indeed a challenge. Testing materials was a big part of our trials to get the system to work, but we were limited on how much materials we could use, so we were careful to not damage any of the materials we were given, as doing so means we would have to purchase our own materials. (Some materials were provided after damaged, materials such as the arduino and sensors had to be repurchased if broken.)

**PROCESS OF MAKING PROJECT:**

As previously stated in the Challenges Faced segment, understanding how each material functioned was a big part of the project. We first started off by looking at a live demonstration by a student who already completed the project. Once we had a basic knowledge on how the project was going to look like, we began creating a demo design in the program SketchUp. Using SketchUp, we were able to create a rendition of what could be our final project design. Although we swayed a bit away from our original design, our SketchUp project influenced our decisions on our final design. Once we had a rough idea of what the box was going to look like, we moved on to the breadboard. During this segment we were given materials and tools to aid us during the breadboard testing phase. As mentioned previously, this was one of the hardest parts of the project. We had to ask multiple classmates and receive input from other students to better understand how to control our arduino code in order to manipulate our lights to the desired timing. In a natural street light configuration, some lights would turn green while others would either be in a yellow or red phase. It was a challenge to get the timing right for the lights but after multiple tests, we were able to properly manipulate the light system to mimic a real traffic system.

In this project, we are required to manipulate the traffic lights based on an input from a pedestrian button placed at the pole of a crossing light. When this button is pressed, it allows a pedestrian to cross the road without worrying about a car hitting them. We first had to understand how to get data from a button. Afterwards, we would take the data from the button and pass it through our arduino code to override the current traffic light configuration. If the lights are green for incoming traffic and the button is pressed to cross the street, the arduino code will recognise this change and proceed to put the incoming traffic light to yellow, then to red. This allowed us to mimic the function of a crossing light in our project. Next we worked on controlling the light of a generic white street light. This light would turn on when the sensor detects the absence of light within the area. If light is detected, the street light will be turned off. This was not very difficult to understand, as it carried the same idea of the other LED lights, except it only swaps from on to off using the data from a sensor detecting a lot of light or barely any light. After we had sorted the function of the street light, next was to set up the parking arm based off of data from an infrared sensor. The infrared sensor continuously emits light and returns data once the infrared light pointed to another area is interrupted. Using this data, we can manipulate a small arm lift, known as a servo, to mimic the use of a parking lot arm used to close off the entrance and exit of a parking lot. The process of setting the sensor up was straightforward, as it uses the same idea as the daylight sensor. Based on the data that is returned by the infrared sensor, the arduino will lift a small arm upwards or downwards.

To finish off the project, we laid out all the materials based on our SketchUp model. As stated previously, our final design is inspired from our SketchUp, but not all features are used, some improved, scrapped, or new additions have been added. Once we had found a suitable size for our final design, we began cutting construction paper and laying it on our box to resemble and create a T intersection scene. Lastly, we took our LEDs, cables, sensor, and other materials to start transitioning to our box design. In order to be able to extend our lights far enough to reach the exterior of the box from the inside, we had to connect all of our components using a solder. Each LED requires two ends to be soldered. This means instead of using one cable to connect lights as we did in our breadboard testing phase, we instead connected the positive and negative ends to essentially achieve the same connection, although it would be later laid on the board differently. Once we had done so for all of our LEDs, we had to create a makeshift traffic light post. To do this, Braulio designed a 3D printed model traffic light posts fitting the LED lights and cables.

**WHAT COULD HAVE BEEN DONE DIFFERENTLY:**

Aside from working on a small time frame, being able to manage our time better could have had a major impact on our final project. We could have spent more time to perfect our finished product. Another thing we could of utilized in advance was learning the arduino language beforehand. We spent a considerable time mastering the arduino code, all of which could have been used to set up the breadboard and the model to show off the design. More time to work on these segments of the project would of made our project eye catching, instead of a bland and copy-paste of what another team has created. Lastly, we could have managed better group parts, who is responsible for who. It started as a mess to begin, difficult to determine what each member was working on. Had we sorted this out in the beginning of the project, we would of had more coordination on what is happening.