Design Lab Experiment No. 1:

*Traffic Light Controller Implementation on Arduino UNO*

By: Alan Palayil

Lab Partner: Individual

Instructor: Prof. Jafar Saniie, Prof. Won-Jae Yi

TA: Mikhail Gromov

ECE 442 Internet of Things and Cyber Physical Systems

Date: 06-03-2022

**Acknowledgment**: I acknowledge all works including figures, codes and writings belong to me and/or persons who are referenced. I understand if any similarity in the code, comments, customized program behavior, report writings and/or figures are found, both the helper (original work) and the requestor (duplicated/modified work) will be called for academic disciplinary action.

Signature: \_\_\_\_\_\_Alan Biju Palayil\_\_\_\_\_\_\_\_

**I. Introduction**

**A. Purpose**

The task of this experiment is to get familiar with the working of the Arduino UNO microcontroller board and programming it is using with the IDE. The experiment is to implement the traffic signal lights using Arduino UNO R3 board. It will be implemented on a breadboard using components like LEDs, jumper wires, push buttons, etc. The programming in Arduino involves having a knowledge of C programming the commands like digitalRead, digitalWrite, inputs, and outputs.

**B. Background**

The scope of this experiment is to understand the working and getting familiar with the Arduino board, its IO ports, interface, and have the microcontroller be programmed to control and simulates the lights using C programming. The experiment is divided into two parts:

1. Simple Two-way Intersection: To consider N/S and E/W as the two directions, for the simple traffic signal, each traffic signal has three lamps- red for stop, yellow for changing, and green for proceed. A simple timer without the use of sensors. The experiment states that N/S is busier than the E/W road, so the N/S green light will stay on 10sec compared to 5sec for the E/W green light. The traffic lights will cycle from red, yellow, and green with the transition time of 1 unit.
2. Two-way Intersection with traffic sensors: Taking the same pattern from the first circuit and adding 2 push buttons which will function as the automobile sensors for traffic from E/W road. If there are no cars waiting on the E/W road, the N/S traffic light remains green. When the sensor is activated, the corresponding traffic light turns green and allows the flow of traffic. This will allow the N/S road to have an unrestricted traffic flow if no cars are waiting at either of the E/W traffic lights.

**II. Lab Procedure and Equipment List**

1. **Equipment**
   * 1 x Arduino UNO board and USB cable
   * 2 x Red LED
   * 2 x Yellow LED
   * 2 x Green LED
   * 2 x Pushbutton
   * 6 x 220 Ohm & 2 x 10k Ohm Resistors
   * 1 x Breadboard
2. **Procedure**

\*Refer to Lab Manual for detailed Procedure\*

**III. Results and Analysis**

1. **Results**

The experiment was modified a little because of the lack of push buttons. The experiment was successfully with the following images:

A circuit board with wires

Description automatically generated with low confidenceA picture containing text, electronics

Description automatically generated

A close-up of a circuit board

Description automatically generated with medium confidence

1. **Answers to Discussion Questions**
   1. Refer to Appendix 1 and 2 for the fully commented program.

*TinkerCAD schematics for Simple Traffic Light and Traffic Lights with Automobile sensor:*

Diagram

Description automatically generatedDiagram

Description automatically generated

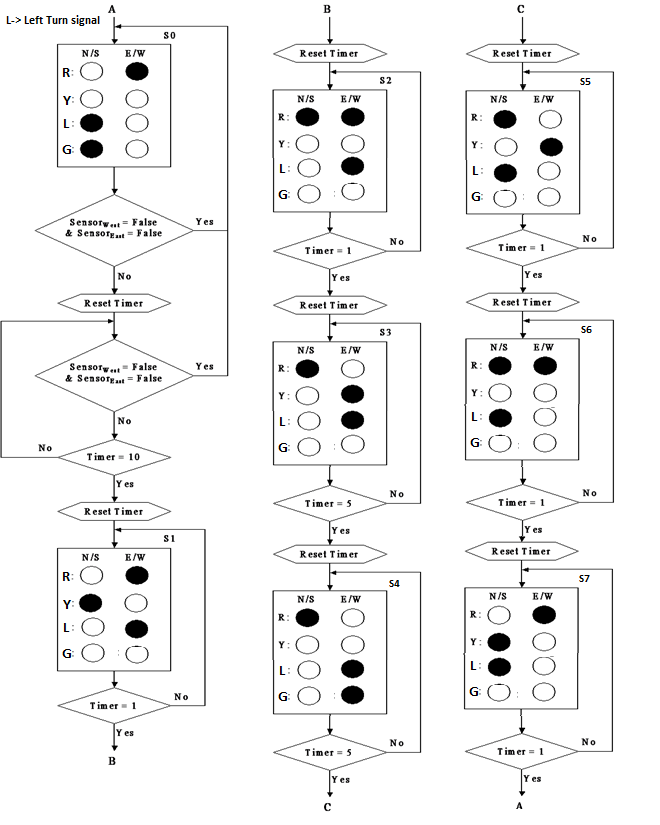
* + 1. Arduino is an open-source computer hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and kits for building digital devices. The board consists of a single integrated chip which has a CPU, Control Unit, Memory, Clock, and I/O. Other circuits and expansion boards with digital, analog, input and output pins can be designed. It can support for SPI and I2C interfaces along with WiFi, Ethernet, and USB connections. It is programmed using C and C++. The Arduino UNO R3 that we are using for the labs is for general purpose development with electronics and coding. It has a ATmega328, 5V, 20 digital I/Os with 6 PWM channels.
    2. Arduino has a lot of different applications in various fields such as homes, medical, industrial, IoT, and farming. Combabilities with a wide variety of sensors like IR, temperature, humidity, and light sensors the Arduino can be used to create smart devices such as automated home appliances, mobile devices, health temperature controller, etc. The only drawback for the Arduino is that it can only be used for small projects (max 20mA of power usage) and projects above that require the addition of a shield which amplifies the power.
  1. Refer to Appendix 3 for the fully commented program.

*TinkerCAD schematics for Ultrasonic sensor:*

Diagram

Description automatically generated

* 1. Refer to Appendix 4 for the fully commented program.

*Flowchart for Traffic Lights with Automobile sensor with left turn:*

*TinkerCAD schematics for Traffic Lights with Automobile sensor with left turn:*Diagram

Description automatically generated

**IV. Conclusion**

It can be concluded that using LEDs and other basic components like push buttons, a reliable code can be developed for the traffic lights. The code can be implemented using the Arduino UNO board and the several types of traffic lights systems can be obtained. The Arduino board is a handy and reliable component with an enormous potential in different projects within IoT.

**References**

[1] Experiment 1 Lab Manual

[2] Lecture Note 3 of ECE 442/510

**Appendix**

1. *Program Code for Simple Traffic Lights*

// North\_South LEDs

#define r\_ns 2

#define y\_ns 3

#define g\_ns 4

// East\_West LEDs

#define r\_ew 5

#define y\_ew 6

#define g\_ew 7

//time\_base define the running speed

//Set it to 1000 is in real time

#define time\_base 500

void setup () {

// initialize all LEDs as output

pinMode (r\_ns, OUTPUT);

pinMode (y\_ns, OUTPUT);

pinMode (g\_ns, OUTPUT);

pinMode (r\_ew, OUTPUT);

pinMode (y\_ew, OUTPUT);

pinMode (g\_ew, OUTPUT);

}

void ChangeLedValue (byte number)

{

digitalWrite (r\_ns, bitRead(number,5));

digitalWrite (y\_ns, bitRead(number,4));

digitalWrite (g\_ns, bitRead(number,3));

digitalWrite (r\_ew, bitRead(number,2));

digitalWrite (y\_ew, bitRead(number,1));

digitalWrite (g\_ew, bitRead(number,0));

}

void LightSequence () {

beginning:

ChangeLedValue(B001100);

delay (time\_base \*10);

ChangeLedValue(B010100);

delay (time\_base \*1);

ChangeLedValue(B100100);

delay (time\_base \*1);

ChangeLedValue(B100001);

delay (time\_base \*5);

ChangeLedValue(B100010);

delay (time\_base \*1);

ChangeLedValue(B100100);

delay (time\_base \*1);

}

void loop ()

{

LightSequence ();

}

1. *Program Code for Traffic Lights with Sensors:*

// North\_South LEDs

#define r\_ns 2

#define y\_ns 3

#define g\_ns 4

// East\_West LEDs

#define r\_ew 5

#define y\_ew 6

#define g\_ew 7

//time\_base define the running speed

//Set it to 1000 is in real time

//Set it to a smaller value for debugging

#define time\_base 500

//automobile sensor

#define e\_sensor 12

#define w\_sensor 13

void setup () {

// initialize all LEDs as output

pinMode (r\_ns, OUTPUT);

pinMode (y\_ns, OUTPUT);

pinMode (g\_ns, OUTPUT);

pinMode (r\_ew, OUTPUT);

pinMode (y\_ew, OUTPUT);

pinMode (g\_ew, OUTPUT);

pinMode (e\_sensor, INPUT);

pinMode (w\_sensor, INPUT);

}

void ChangeLedValue (byte number)

{

digitalWrite (r\_ns, bitRead(number,5));

digitalWrite (y\_ns, bitRead(number,4));

digitalWrite (g\_ns, bitRead(number,3));

digitalWrite (r\_ew, bitRead(number,2));

digitalWrite (y\_ew, bitRead(number,1));

digitalWrite (g\_ew, bitRead(number,0));

}

void LightSequence\_ns ()

{

ChangeLedValue(B001100);

}

void LightSequence\_ew ()

{

//sequence for traffic light pattern using the flowchart

delay (time\_base \*10);

ChangeLedValue(B010100);

delay (time\_base \*1);

ChangeLedValue(B100100);

delay (time\_base \*1);

ChangeLedValue(B100001);

delay (time\_base \*5);

ChangeLedValue(B100010);

delay (time\_base \*1);

ChangeLedValue(B100100);

delay (time\_base \*1);

}

void carDetection ()

{

//To detect if there is a car in the E/W road

if (digitalRead (12) || digitalRead (13))

{

LightSequence\_ew ();

}

else

{

LightSequence\_ns ();

}

}

void loop ()

{

carDetection ();

}

1. *Program Code for Ultrasonic Sensor:*

#include <LiquidCrystal.h> // includes the LiquidCrystal Library

LiquidCrystal lcd (1, 2, 4, 5, 6, 7); // Creates an LCD object. Parameters: (rs, enable, d4, d5, d6, d7)

const int trigPin = 9;

const int echoPin = 10;

const int LED = 8;

long duration;

int cm, inches;

int threshold= 0;

void setup () {

lcd.begin (16,2); // Initializes the interface to the LCD screen, and specifies the dimensions (width and height) of the display

pinMode (trigPin, OUTPUT);

pinMode (echoPin, INPUT);

pinMode (LED, OUTPUT);

}

void loop () {

threshold= 310\*0.034/2;

digitalWrite (trigPin, LOW);

delayMicroseconds (2);

digitalWrite (trigPin, HIGH);

delayMicroseconds (10);

digitalWrite (trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

cm= duration\*0.034/2;

inches = duration\*0.0133/2;

lcd.setCursor (0,0); // Sets the location at which subsequent text written to the LCD will be displayed

lcd.print ("Distance: "); // Prints string "Distance" on the LCD

lcd.print (cm); // Prints the distance value from the sensor

lcd.print (" cm");

delay (10);

lcd.setCursor (0,1);

lcd.print ("Distance: ");

lcd.print(inches);

lcd.print (" inch");

if (cm <= threshold) {

digitalWrite (LED, HIGH);

}

delay (10);

}

1. *Program Code for Traffic Lights with Sensors with left turn:*

// North\_South LEDs

#define r\_ns 2

#define y\_ns 3

#define g\_ns 4

// East\_West LEDs

#define r\_ew 5

#define y\_ew 6

#define g\_ew 7

//Left turn green lights

#define gl\_ns 8

#define gl\_ew 9

//time\_base define the running speed

//Set it to 1000 is in real time

//Set it to a smaller value for debugging

#define time\_base 500

//automobile sensor

#define e\_sensor 12

#define w\_sensor 13

void setup () {

// initialize all LEDs as output

pinMode (r\_ns, OUTPUT);

pinMode (y\_ns, OUTPUT);

pinMode (g\_ns, OUTPUT);

pinMode (r\_ew, OUTPUT);

pinMode (y\_ew, OUTPUT);

pinMode (g\_ew, OUTPUT);

pinMode (gl\_ns, OUTPUT);

pinMode (gl\_ew, OUTPUT);

pinMode (e\_sensor, INPUT);

pinMode (w\_sensor, INPUT);

}

void ChangeLedValue (byte number)

{

digitalWrite (r\_ns, bitRead(number,5));

digitalWrite (y\_ns, bitRead(number,4));

digitalWrite (g\_ns, bitRead(number,3));

digitalWrite (r\_ew, bitRead(number,2));

digitalWrite (y\_ew, bitRead(number,1));

digitalWrite (g\_ew, bitRead(number,0));

}

void LightSequence\_ns ()

{

ChangeLedValue(B001100);

digitalWrite (gl\_ns, HIGH);

}

void LightSequence\_ew ()

{

//sequence for traffic light pattern using the modified flowchart

delay (time\_base \*10);

ChangeLedValue(B010100);

digitalWrite (gl\_ns, LOW);

digitalWrite (gl\_ew, HIGH);

delay (time\_base \*1);

ChangeLedValue(B100100);

delay (time\_base \*1);

ChangeLedValue(B100001);

delay (time\_base \*5);

ChangeLedValue(B100010);

digitalWrite (gl\_ns, HIGH);

digitalWrite (gl\_ew, LOW);

delay (time\_base \*1);

ChangeLedValue(B100100);

delay (time\_base \*1);

}

void carDetection ()

{

//To detect if there is a car in the E/W road

if (digitalRead (12) || digitalRead (13))

{

LightSequence\_ew ();

}

else

{

LightSequence\_ns ();

}

}

void loop ()

{

carDetection ();

}