American International University- Bangladesh (AIUB)
Faculty of Engineering (EEE)

Course Name :	Microprocessor and Embedded Systems Laboratory	**Course Code :**	4103
Semester :	Summer 2023	**Sec :**	F
Lab Instructor :	Md Sajid Hossain		

Experiment No :	02
Experiment Name :	Timers: Implementation of a traffic control System

Submitted by (NAME):	**Srabone Raxit**	**Student ID:**	**21-45038-2**

Group Members	**ID**	**Name**
1.	20-42461-1	Rifat Hossain
2.	20-42831-1	Ishrat Jahan
3.	21-45019-2	Ahnaf Abdullah Zayad
4.	21-45038-2	Srabone Raxit
5.	21-45206-2	Kazi Ramisa Samiha
6.	21-45263-2	Shakibul Hasan
7.		

Performance Date :	**13.06.2023**	**Due Date :**	**20.06.2023**

Marking Rubrics (to be filled by Lab Instructor)

Category	Proficient [6]	Good [4]	Acceptable [2]	Unacceptable [1]	Secured Marks
Theoretical Background, Methods & procedures sections	All information, measures and variables are provided and explained.	All Information provided that is sufficient, but more explanation is needed.	Most information correct, but some information may be missing or inaccurate.	Much information missing and/or inaccurate.	
Results	All of the criteria are met; results are described clearly and accurately;	Most criteria are met, but there may be some lack of clarity and/or incorrect information.	Experimental results don't match exactly with the theoretical values and/or analysis is unclear.	Experimental results are missing or incorrect;	
Discussion	Demonstrates thorough and sophisticated understanding. Conclusions drawn are appropriate for analyses;	Hypotheses are clearly stated, but some concluding statements not supported by data or data not well integrated.	Some hypotheses missing or misstated; conclusions not supported by data.	Conclusions don't match hypotheses, not supported by data; no integration of data from different sources.	
General formatting	Title page, placement of figures and figure captions, and other formatting issues all correct.	Minor errors in formatting.	Major errors and/or missing information.	Not proper style in text.	
Writing & organization	Writing is strong and easy to understand; ideas are fully elaborated and connected; effective transitions between sentences; no typographic, spelling, or grammatical errors.	Writing is clear and easy to understand; ideas are connected; effective transitions between sentences; minor typographic, spelling, or grammatical errors.	Most of the required criteria are met, but some lack of clarity, typographic, spelling, or grammatical errors are present.	Very unclear, many errors.	
Comments:				Total Marks (Out of):	

"Implementation of a Traffic Control System using Arduino Uno microcontroller, to get familiar with timer and microcontroller."

I. Abstract:

The experiment describes the implementation of a Traffic Control System using an Arduino Uno microcontroller. The experiment also highlights the utilization of timers within the microcontroller to regulate traffic lights using red, green, and yellow LEDs, showcasing the practical application of the Arduino UNO in real-time systems. This experiment serves as an introductory exploration of timers and microcontrollers, providing a foundation for further understanding and experimentation in the field of embedded systems.

II. Introduction:

The main objective of this experiment was:

1) To get familiar with Timers
2) To get familiar with Arduino Uno microcontroller
3) To use Timers for the implementation of a traffic control system.

III. Theory and Methodology:

Timer: Every electronic component of a sequential logic circuit works on a time base. This timer base helps to keep all the work synchronized. Without a time base, devices would have no idea as to when to perform particular actions. Thus, the timer is an important concept in the field of electronics.

A timer/counter is a piece of hardware built into the Arduino controller. It is like a clock and can be used to measure time events. A timer is a register whose value increases/decreases automatically.

In AVR, timers are of two types: 8-bit and 16-bit timers. In an 8-bit timer, the register used is 8-bit wide whereas, in a 16-bit timer, the register width is 16 bits. This means that the 8-bit timer is capable of counting $2^8 = 256$ steps from 0 to 255. Similarly, a 16-bit timer is capable of counting $2^{16} = 65536$ steps from 0 to 65535.

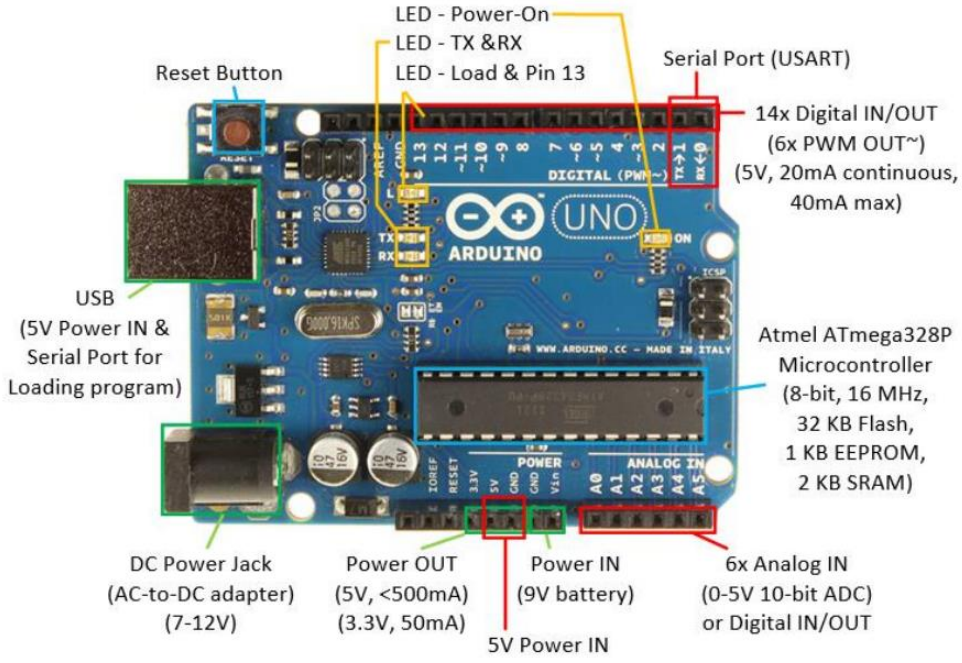

Figure 3.1: Overview of an Arduino Uno Board

IV. Apparatus:

1. Arduino Uno R3 Board (ATmega328P)
2. Arduino IDE
3. Resistors (200 ohms, 3)
4. LED Lights (Red, Green, and Yellow)
5. Connecting Wires

V. Precautions:

1. The computers' connection with the Arduino board was made sure.
2. The ports' connection on the board that matched with the code implementation were checked carefully.

VI. Experimental Procedure:

1. The Arduino Uno IDE 1.8.2 was opened and then, a blank sketch was opened.

2. A window was come up on the PC.

3. The following program was written for Traffic Light Control System:

Code for Traffic System Control:

```
#define RED_PIN 8 //define name of pins used
#define YELLOW_PIN 9
#define GREEN_PIN 10

//define the delays for each traffic light color
int red_on = 3000; //3s delay
int yellow_on = 1000; //1s delay
int green_on = 3000; //3s delay


//user define
int delay_timer (int milliseconds)
{
 int count = 0;
 while(1)
  {
 if(TCNT0 >= 16) // Checking if 1 millisecond has passed
    {
 TCNT0=0;
 count++;
 if (count == milliseconds) //checking if required milliseconds delay has passed
       {
  count=0;
  break; // exits the loop
       }
    }
  }
 return 0;
```

```
}

void setup() {
 //define pins connected to LEDs as outputs
   pinMode(RED_PIN, OUTPUT);
   pinMode(YELLOW_PIN, OUTPUT);
   pinMode(GREEN_PIN, OUTPUT);

 //set up timer
 TCCR0A = 0b00000000;
 TCCR0B = 0b00000101; //setting pre-scaler for timer clock
 TCNT0=0;
}

void loop() {
 //to turn red LED on
 digitalWrite(RED_PIN, HIGH);
 delay_timer(red_on);
 digitalWrite(RED_PIN, LOW);

 //to turn yellow LED on
 digitalWrite(YELLOW_PIN, HIGH);
 delay_timer(yellow_on);
 digitalWrite(YELLOW_PIN, LOW);

 // to turn green LED on
 digitalWrite(GREEN_PIN, HIGH);
 delay_timer(green_on);
 digitalWrite(GREEN_PIN, LOW);

//to turn yellow LED on
 digitalWrite(YELLOW_PIN, HIGH);
 delay_timer(yellow_on);
 digitalWrite(YELLOW_PIN, LOW);

}
```

4. After writing the program the sketch was saved, went to File->Save As->give a File name-> Select Save.

5. Then the code was compiled to find out and correct the errors, go to Sketch- >Verify/Compile.

6. After compiling, the code was uploaded onto the Arduino Uno board. To upload the program, the Arduino UNO R3 board was connected to the PC with a USB cable. Before uploading the code, the board type and port were selected at Arduino IDE, go to

   - Tools-> Board:"Arduino/Genuino Uno" -> Arduino/Genuino Uno.
   - Tools->Ports-> COMx

   After selecting the board and port the upload option was clicked at the Arduino IDE to upload the code.

VII. Simulation and Measurement:

Simulation was done after implementation of the hardware part to match with the result.

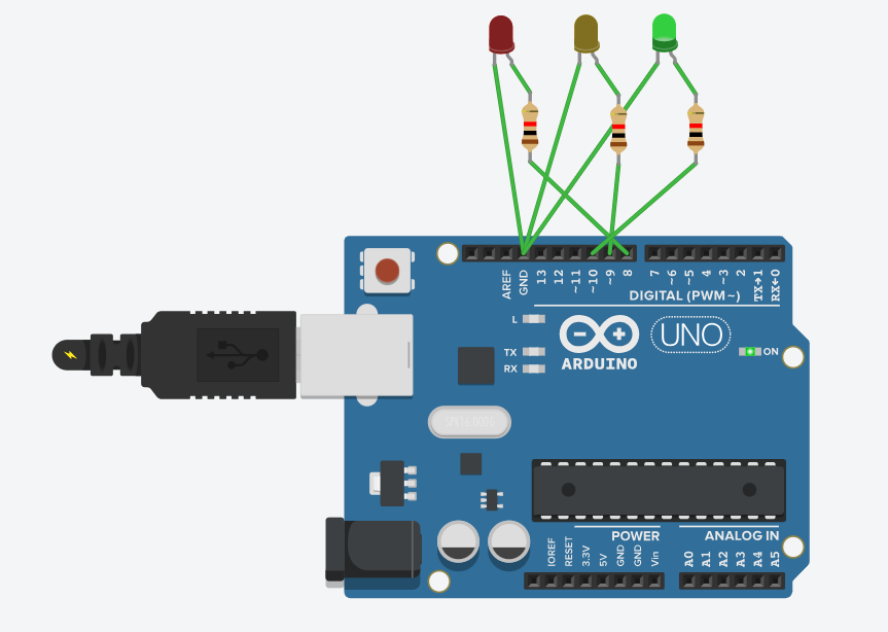

Figure 7.1: Traffic control System (Green LED on)

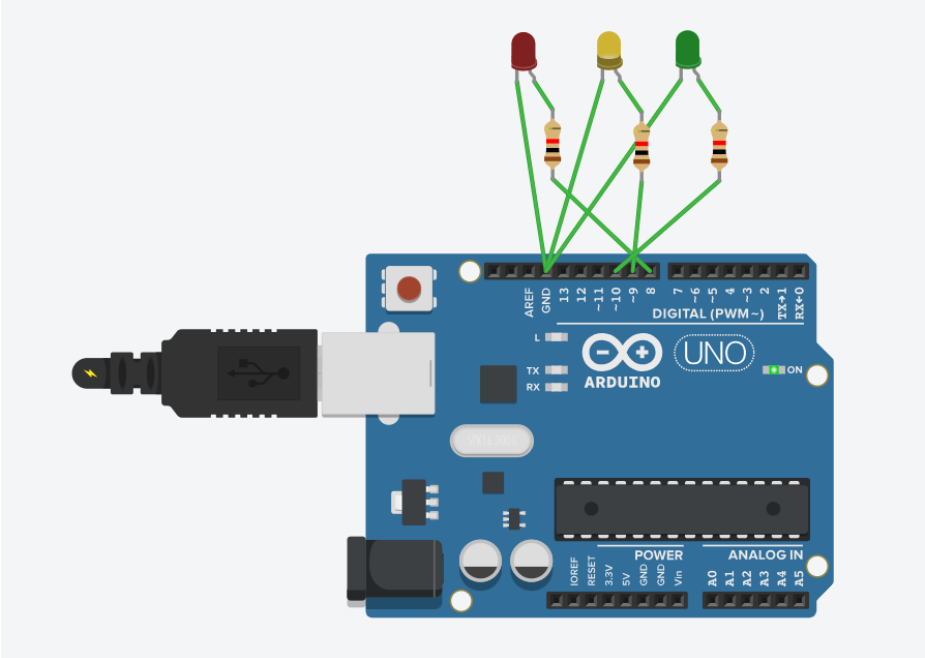

Figure 7.2: Traffic control System (Yellow LED on)

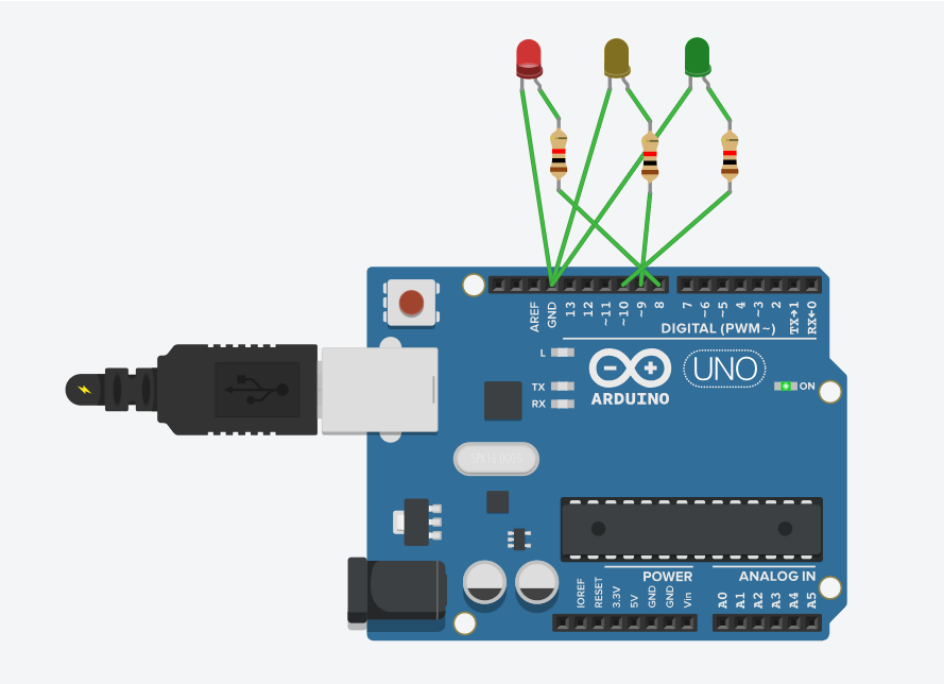

Figure 7.3: Traffic control System (Red LED on)

VIII. Results:

In the experiment, a traffic control system was designed using Timer in the code implementation and the desired Outputs were:

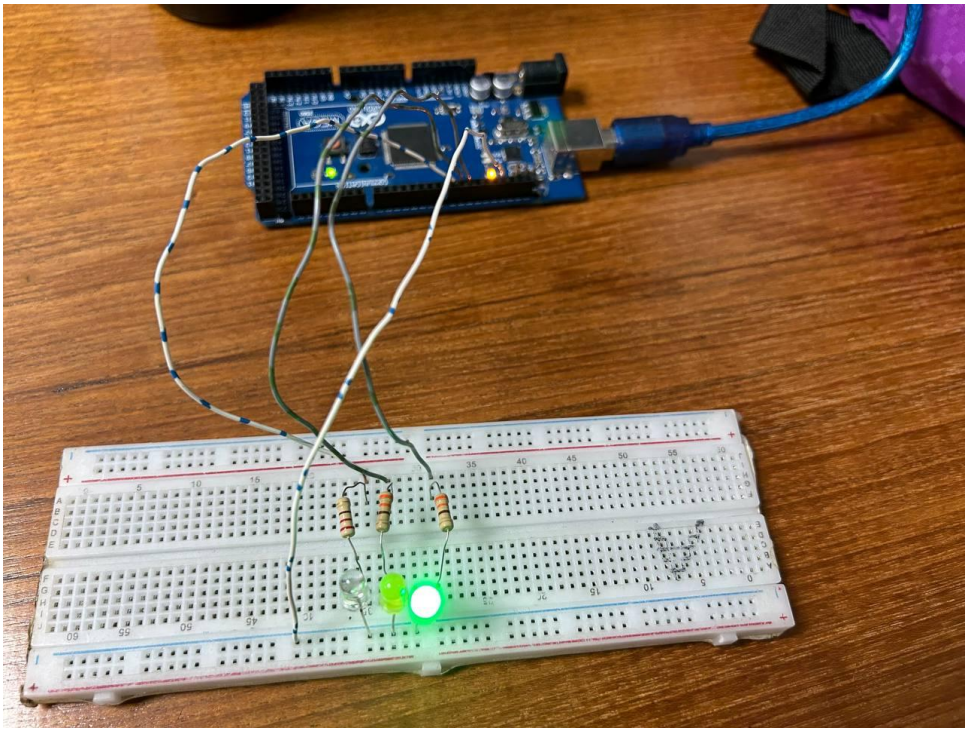

Figure 8.1: Traffic control System (Green LED on)

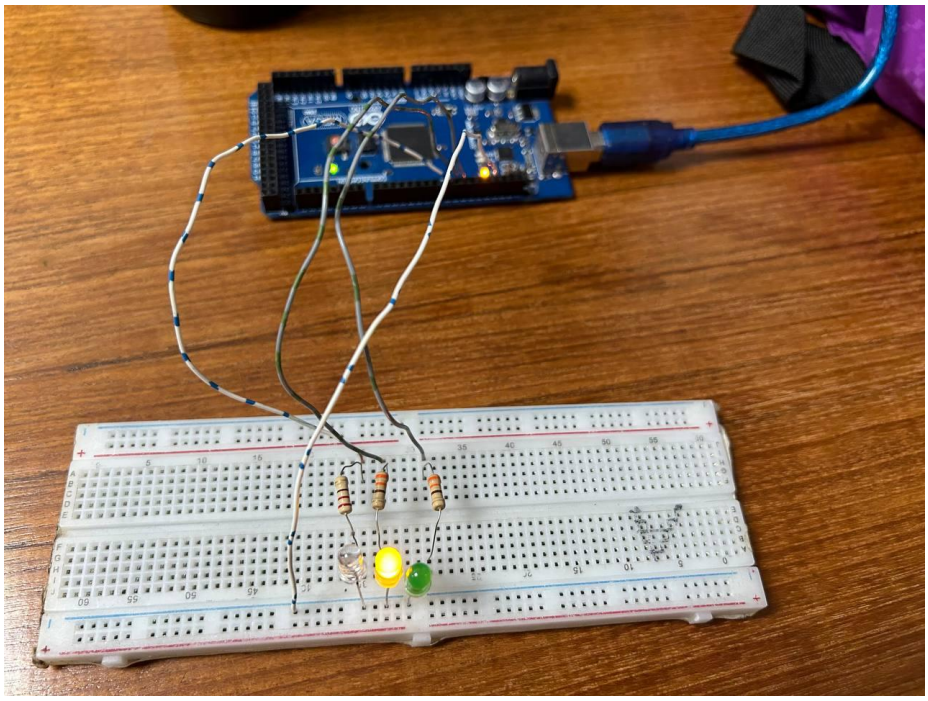

Figure 8.2: Traffic control System (Yellow LED on)

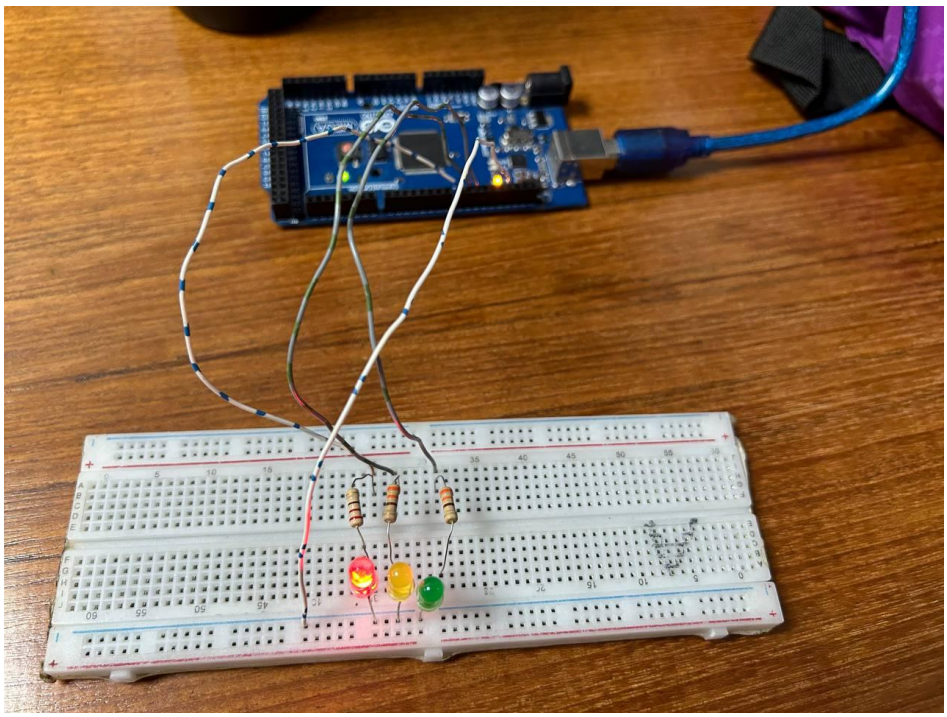

Figure 8.3: Traffic control System (Red LED on)

IX. Discussion:

In this experiment, a Traffic Control System was carried out successfully, where the green, yellow, and red LEDs were operated using Arduino UNO microcontroller. This successful outcome demonstrates the feasibility of using microcontrollers in real-world applications.

X. Conclusion:

The Traffic Control System was successfully implemented, and the simulation was matched with the experimental results. Thus, it can be concluded that the experiment was successful.

XI. Reference (s):

[1] ATmega328 Arduino Uno Board Working and Its Applications-Elprocus. Available at: https://www.elprocus.com/atmega328-arduino-uno-board-working-and-its-applications/?fbclid=IwAR31Al9NcljIcVp0TRvwr7XA7Pu6Vq9yaaJXyTs7kpuPHvqNjOXkS2dFUAk (Accessed: 19 June 2023).