### American International University- Bangladesh

**Department of Electrical and Electronic Engineering**

EEE4103: Microprocessor and Embedded Systems Laboratory

**Title:** Implementation of traffic runway lights using timer functions.

**Introduction:** The objective of this experiment is to get familiarized with Timers and use them for the implementation of a traffic control system.

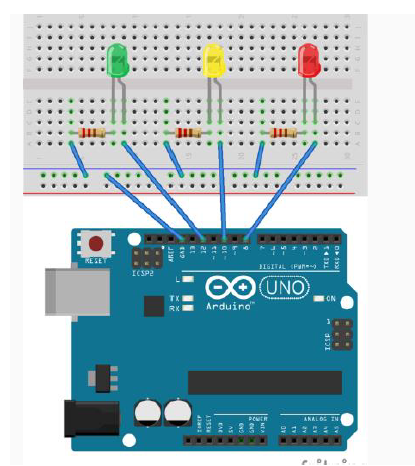
**Theory and Methodology:**

***Timer:*** Every electronic component of a sequential logic circuit works on a time base. This time 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 28=256 steps from 0 to 255. Similarly, a 16-bit timer is capable of counting 216=65536 steps from 0 to 65535.

**Experimental setup:**



**Apparatus:**

* Arduino Uno/ Arduino Mega
* LED lights (YELLOW, RED, and GREEN)
* Resistors (220 ohms)

**Code implementation of a traffic system with Timer:**

|  |
| --- |
| #define RED\_PIN 8 //define name of pins used  #define YELLOW\_PIN 10  #define GREEN\_PIN 12  //define the delays for each traffic light color  int red\_on = 3000; //3s delay  int red\_yellow\_on = 1000; //1s delay  int green\_on = 3000; //3s delay  int green\_blink = 500; //.5s delay  int yellow\_on = 1000; //1s delay  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 make red LED on  digitalWrite(RED\_PIN, HIGH);  delay\_timer(red\_on);    //to turn yellow LED on  digitalWrite(YELLOW\_PIN, HIGH);  delay\_timer(red\_yellow\_on);    //turning off RED\_PIN and YELLOW\_PIN, and turning on greenLED  digitalWrite(RED\_PIN, LOW);  digitalWrite(YELLOW\_PIN, LOW);  digitalWrite(GREEN\_PIN, HIGH);  delay\_timer(green\_on);  digitalWrite(GREEN\_PIN, LOW);    //for turning green Led on and off for 3 times  for(int i = 0; i < 3; i = i+1)  {  delay\_timer(green\_blink);  digitalWrite(GREEN\_PIN, HIGH);  delay\_timer(green\_blink);  digitalWrite(GREEN\_PIN, LOW);  }    //for turning on yellow LED  digitalWrite(YELLOW\_PIN, HIGH);  delay\_timer(yellow\_on);  digitalWrite(YELLOW\_PIN, LOW);  } |

**Questions for report writing:**

1. Include all codes and scripts into the lab report following the writing template mentioned in appendix A of Laboratory Sheet Experiment 3.
2. Implement this system using an online simulation platform [*www.tinkercad.com*](http://www.tinkercad.com)*.*
3. Configure the system to have delays for outputs according to your ID. Consider the last three digits from your ID (if your ID is XX-XXABC-X then consider A for the RED light, B for the YELLOW LED, and C for GREEN LED). Include the program and results within your lab report.

**Reference(s):**

1) <https://www.arduino.cc/>.

2)ATMega328 manual

3) https://www.avrfreaks.net/forum/tut-c-newbies-guide-avr-timers

4) http://maxembedded.com/2011/06/avr-timers-timer0/