American International University- Bangladesh
Department of Electrical and Electronic Engineering
EEE4103: Microprocessor and Embedded Systems Laboratory

Title: Timers: Implementation of a traffic control system

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

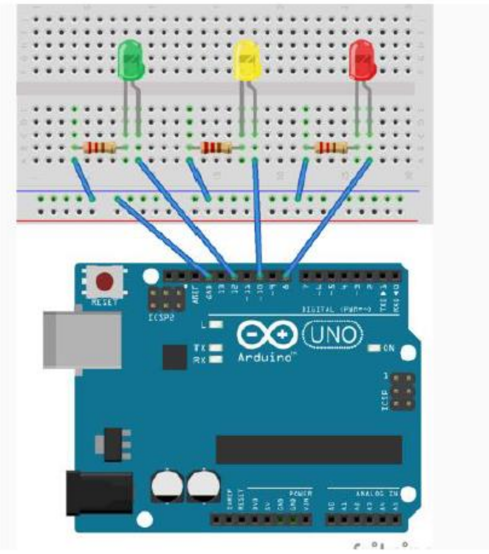
Theory and Methodology:

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 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.

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 = _______; //3s delay
int red_yellow_on = _____; //1s delay
int green_on = _____; //3s delay
int green_blink = ___; //.5s delay
int yellow_on = ______; //1s delay

int delay_timer (int milliseconds)
{
 int count = 0;
 while(1)
 {
  if(TCNT0 >= __)  // 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 = ___________;
  TCCR0B = ___________; //setting pre-scaler for timer clock
  TCNT0=____;
}
```

```
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*.
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/

