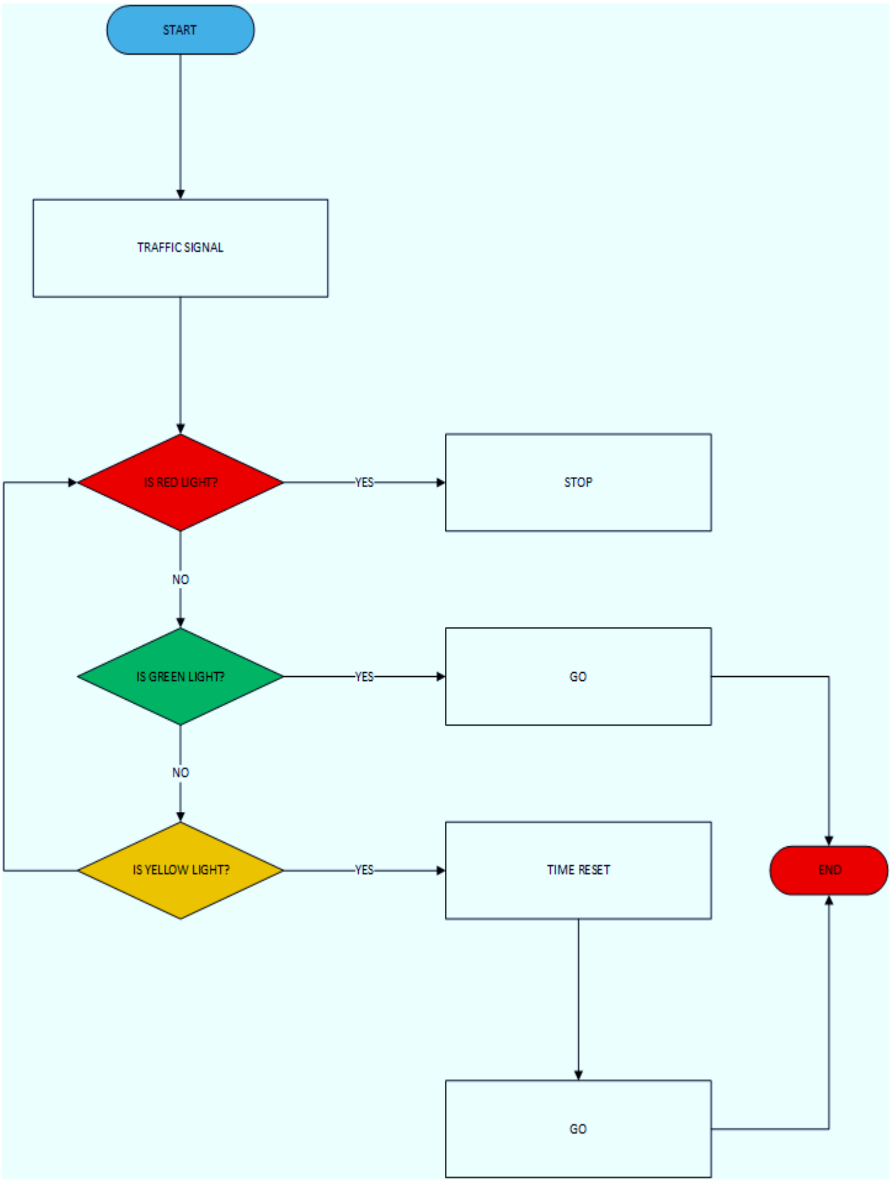
Hubei University of Technology

Experiment report

Grade	

Course name	Embedded SYSTEMS AND DESIGNS			
Experimental name	EXPERIMENT 3 – TRAFFIC LIGHT SYSTEM DESIGN			
Departments	COMPUTER SCIENCE	**Lecturer**	Dr. Liu Chun	
Name	**Rimon Mahmud**	**Student id**	**1811561124**	
Experimental purpose	The aim of this experiment is to design a four-way traffic light system using the 8051 microcontroller.			
Experimental preparation	**1.Experimental environment:** PROTEUS 8 PROFESSIONAL, WINDOWS 10 **2. Knowledge preparation**：8051 Architecture, C language, 8051 Instruction set.			
Experimental content	A traffic light system is a system for indicating the status of a variable using the red, yellow and green traffic lights. It is usually placed at a pedestrian crossing, cross-road or intersection to manage and control the flow of traffic. The green light allows traffic to proceed in the noted direction. The amber/yellow light warns the drivers that the signal is about to change to red. Finally, the red signal prohibits any traffic from proceeding and that drivers should stop and wait for the lights to change back to red for them to move.			

Experimental analysis	Traffic light system is essential in an urban and populated environments such as big cities across the world as it ensures smooth flow of traffic at roundabouts and crossroads. They also help in preventing accidents at intersections.
Experimental flowchart	 START TRAFFIC SIGNAL IS RED LIGHT? YES STOP NO IS GREEN LIGHT? YES GO NO IS YELLOW LIGHT? YES TIME RESET END GO

Code

```
#include <reg51.h>
#include <stdio.h>

/**************************
Cross Road Traffic Control C Program

*****************************************/

#define uchar unsigned char
# define uint unsigned int

/******Define control bits**************/
/******EW:EAST-WEST ,SN:SOUTH-NORTH****************/
sbit      EW_LED2=P2^3;       //EW_LED2 control bit
sbit      EW_LED1=P2^2;       //EW_LED1 control bit
sbit      SN_LED2=P2^1;       //SN_LED2 control bit
sbit      SN_LED1=P2^0;       //SN_LED1 control bit
sbit      SN_Yellow=P1^6;      //SN yellow light
sbit      EW_Yellow=P1^2;      //EWyellow light
  sbit      EW_Red= P1^3;      //EW red light
  sbit SN_Red=P1^7;         //SN red light
  bit     Flag_SN_Yellow; //SN yellow light flag
  bit     Flag_EW_Yellow;     //EW yellow light flag
char Time_EW;//EW countdown unit
char Time_SN;//SN countdown unit
uchar EW=60,SN=40, EWL=19,SNL=19; //default value,   normal mode
uchar EW1=60,SN1=40,EWL1=19,SNL1=19; //modified value
uchar code
table[10]={0X3F,0X06,0X5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
//1~~~~~9display code
uchar code S[8]={0x28,0x48,0x18,0x48,0x82,0x84,0x81,0x84};
/*************************delay*******************/
void Delay (uchar   a)
  {
     uchar     i;
     i=a;
     while (i--){;}
  }
  /**************diaplay*************/
  void Display(void)
  {
  char h,l;
  h=Time_EW/10;
```

```
l=Time_EW%10;
P0=table[l];
EW_LED2=1;
Delay(2);
EW_LED2=0;
P0=table[h];
EW_LED1=1;
Delay(2);
EW_LED1=0;

h=Time_SN/10;
l=Time_SN%10;
P0=table[l];
SN_LED2=1;
Delay(2);
SN_LED2=0;
P0=table[h];
SN_LED1=1;
Delay(2);
SN_LED1=0;

}

/********To interrupt service program*********/
void timer0(void)interrupt 1 using 1
{
static uchar count;
TH0=65530;
TL0=(65536-50000)%256;
count++;

if(count==10)
{
if(Flag_SN_Yellow==1)
{SN_Yellow=~SN_Yellow;}
if(Flag_EW_Yellow==1)
{EW_Yellow=~EW_Yellow;}
}
if(count==20)
{
Time_EW--;
Time_SN--;
if(Flag_SN_Yellow==1)
{SN_Yellow=~SN_Yellow;}
```

```
 if(Flag_EW_Yellow==1)
 {EW_Yellow=~EW_Yellow;}
 count=0;
 }

 }

 void main(void)

 {
IT0=1;
 TMOD=0x01;
 TH0=(65536-50000)/256;
 TL0=(65536-50000)%265;
 EA=1;
 ET0=1;
 EX0=1;
 TR0=1;

 while(1)

 {
 Flag_EW_Yellow=0;
 Time_EW=EW;
 Time_SN=SN;
 while(Time_SN>=5)
 {P1=S[0];
 Display();}
 /**S1*******/
 P1=0x00;
 while(Time_SN>=0)
 {Flag_SN_Yellow=1;

 EW_Red=1;


 Display();
 }
 /****s2******/
 Flag_SN_Yellow=0;
 Time_SN=SNL;
 while(Time_SN>=5)
 {P1=S[2];
     Display();}
```

```
    /***s3***/
P1=0x00;
while(Time_SN>=0)
{Flag_SN_Yellow=1;
EW_Red=1;

Display();}


/*assignment**********/
EW=EW1;
SN=SN1;
EWL=EWL1;
SNL=SNL1;


/**************s4***********/
Flag_SN_Yellow=0;
Time_EW=SN;
Time_SN=EW;
while(Time_EW>=5)
{P1=S[4];
Display();}
/*****s5********/
P1=0X00;
while(Time_EW>=0)
{Flag_EW_Yellow=1;
SN_Red=1;
Display();}
Display();}
/******s6**********/
Flag_EW_Yellow=0;
Time_EW=EWL;
while(Time_EW>=5)
{P1=S[6];
Display();}

/************s7******/
P1=0X00;
while(Time_EW>=0)
{Flag_EW_Yellow=1;
SN_Red=1;
Display();}
```

```
/*************assignment**********/
EW=EW1;
SN=SN1;
EWL=EWL1;
SNL=SNL1;
}
```