Project Report

on

Automatic Railway Gate Controller with High Speed Alerting System

Submitted by

Sk. Md. Zubayer-Al-Mahmud

Student ID: 1017311014

Course Name: Embedded System Design

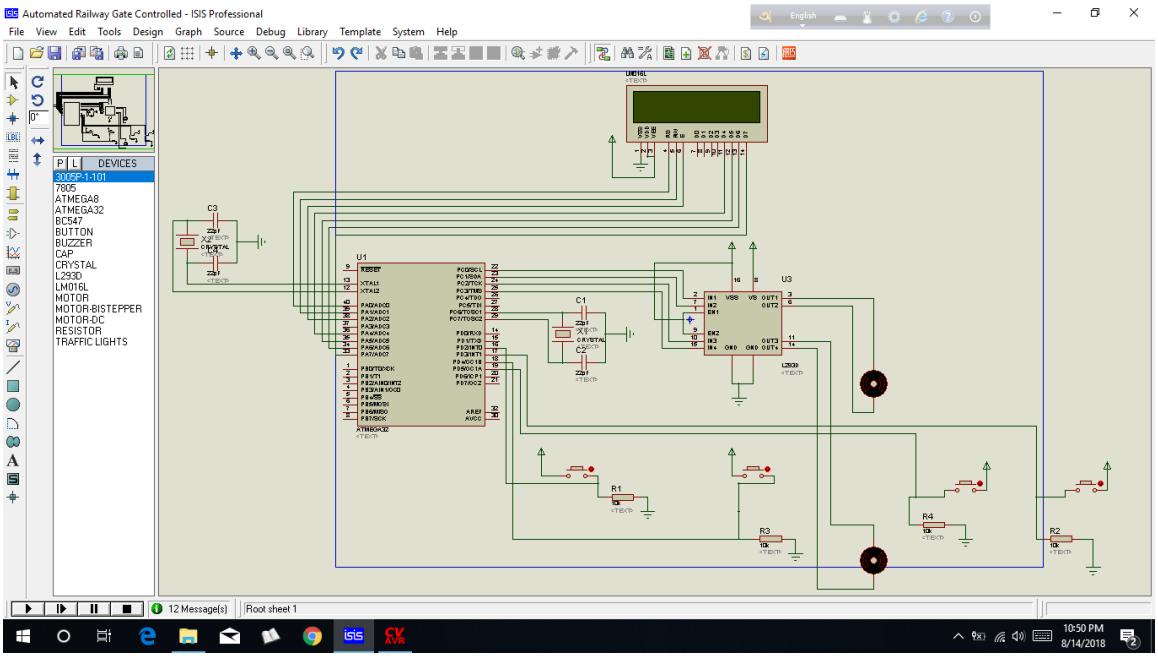
Course Code: 5307

Institute of Information & Communication Technology

Bangladesh University of Engineering & Communication Technology

(IICT, BUET)

Introduction: The main aim of this project is to operate and control the unmanned railway gate in the proper manner in order to avoid the accidents in the unmanned railway crossing. In a country like ours where there are many unmanned railway crossings, accidents are increasing day by day. These train accidents are due to the absence of human power in the railway. In order to overcome the accidents due to the above problem we have planned to design the project.Automatic Railway Gate Control System with High Speed Alerting System is an innovative circuit which automatically controls the operation of railway gates detecting the arrival and departure of trains at the gate. It has detectors at the far away distance on the railway track which allows us to know the arrival and departure of the train. These detectors are given to microcontroller which activates the motors which open/close the railway gate correspondingly.

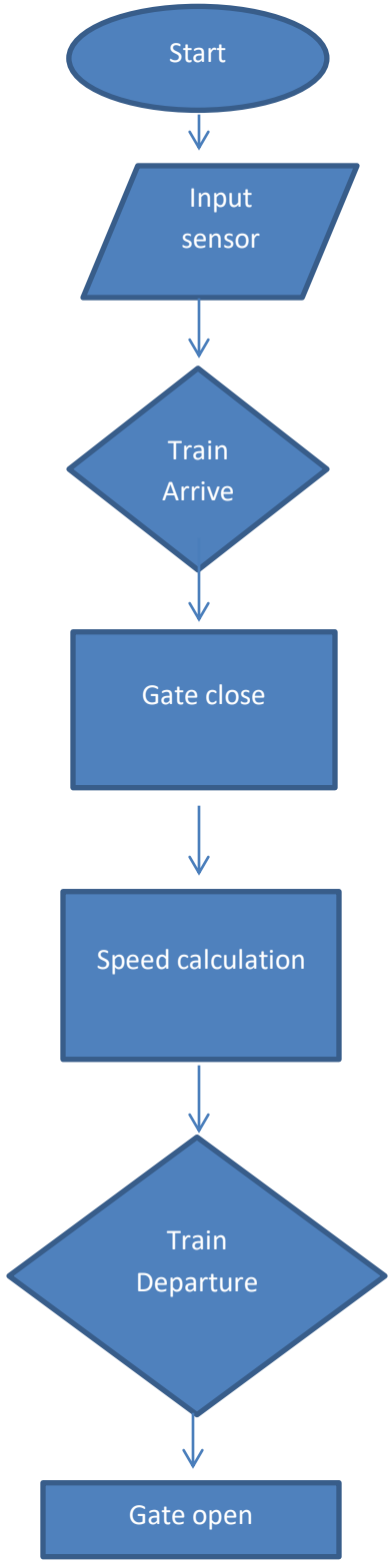


Picture 1: Design of the project in Proteus

Required Components:

1. DC Geared Motors
2. pressure sensor - 4
3. ATmega32 - 1
4. crystal – 2/1
5. IC-L293D - 1
6. Registers
7. Capacitors
8. Battery
9. LCD display
10. some wires, burg strips etc.

Flow chart of the program developed :



Program code :

```
#include <mega32.h>

#include <alcd.h>

#include <delay.h>

#include <stdlib.h>

#define push1   PIND.2

#define push2   PIND.3

#define push3   PIND.4

#define push4   PIND.5

#define buzz_ddr  DDRB

#define buzz PORTB


// Declare your global variables here

float speed=0;

char disp[16];

int second=0, time=0;

int distance=10; // distance= 10 meter


// Timer1 overflow interrupt service routine

interrupt [TIM1_OVF] void timer1_ovf_isr(void)

{

TCNT1H=0xC2;

TCNT1L=0xF7;
```

```
  second++;
 if(second==2){
   PORTC = 0b00000000;
 } }


void main(void)
{    buzz_ddr=0xFF;
TCCR1A=(0<<COM1A1) | (0<<COM1A0) | (0<<COM1B1) | (0<<COM1B0) | (0<<WGM11) | (0<<WGM10);
TCCR1B=(0<<ICNC1) | (0<<ICES1) | (0<<WGM13) | (0<<WGM12) | (1<<CS12) | (0<<CS11) | (1<<CS10);
TCNT1H=0xC2;
TCNT1L=0xF7;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=(0<<OCIE2) | (0<<TOIE2) | (0<<TICIE1) | (0<<OCIE1A) | (0<<OCIE1B) | (1<<TOIE1) | (0<<OCIE0) |
(0<<TOIE0);
      lcd_init(16);
      lcd_gotoxy(0,0);
      DDRC=0xFF;
      DDRD=0x00;
// Global enable interrupts
#asm("sei")
```

```
while (1)
    {
    if(push1==1)
    {
    while (1)
    {
     if(push1==1){
     lcd_clear();
     lcd_gotoxy(0,0);
     lcd_putsf("Train Comming");
     lcd_gotoxy(0,1);
     lcd_putsf("Gate Close");
     delay_ms(10);
      //clockwise rotation
      PORTC = 0b00000101;
      second=0;
      buzz = 0xFF;
      }
    if(push3==1){
      lcd_clear();
      lcd_gotoxy(0,0);
      lcd_putsf("Train speed:");
      delay_ms(5);
      lcd_gotoxy(0,1);
      time=second;
```

```
 second=0;
 delay_ms(500);
 speed=distance*3.6;    //speed= km/hr
 speed=speed/time;
 ftoa(speed,2,disp);
 lcd_puts(disp);
 delay_ms(5);
 lcd_gotoxy(8,1);
 lcd_putsf("km/hr");
 delay_ms(10);
 second=0;
 }
if(push2==1){
buzz = 0x00;
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Train departure");
lcd_gotoxy(0,1);
lcd_putsf("Gate Open");
// counter-clockwise rotation
  PORTC = 0b00001010;
  delay_ms(2000);        // wait 2s
  PORTC = 0b00000000;
   break;
} } }
```

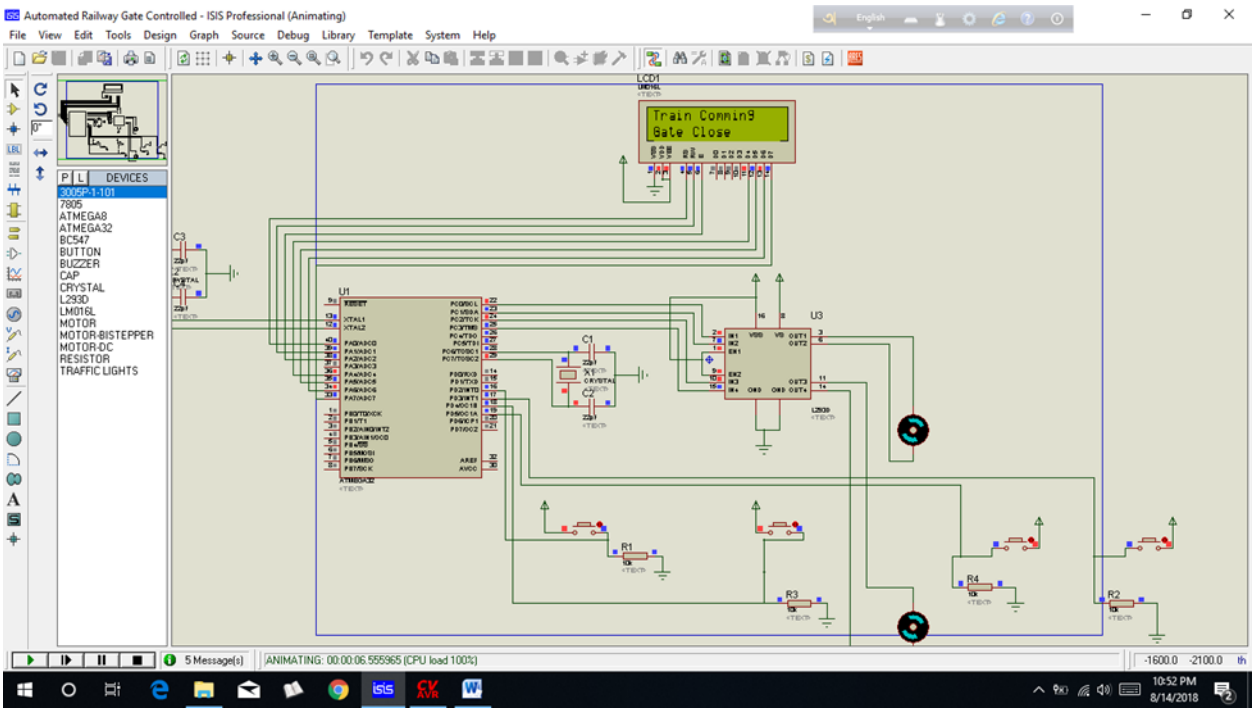
```
//////////////////////////////////////////////////////

if(push2==1)
{
while (1)
{
  if(push2==1){
  lcd_clear();
  lcd_gotoxy(0,0);
  lcd_putsf("Train Comming");
  lcd_gotoxy(0,1);
  lcd_putsf("Gate Close");
  delay_ms(10);
   //clockwise rotation
   PORTC = 0b00000101;
   second=0;
   //delay_ms(2000);        // wait 2s
   buzz = 0xFF;
   }
 if(push4==1){
   lcd_clear();
   lcd_gotoxy(0,0);
   lcd_putsf("Train speed:");
   delay_ms(5);
```

```
 lcd_gotoxy(0,1);
 time=second;
 second=0;
 delay_ms(500);
 speed=distance*3.6/time;    //speed= km/hr
 //speed=speed/time;
 ftoa(speed,2,disp);
 lcd_puts(disp);
 delay_ms(5);
 lcd_gotoxy(8,1);
 lcd_putsf("km/hr");
 delay_ms(10);
 second=0;
 }
if(push1==1){
buzz = 0x00;
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Train departure");
lcd_gotoxy(0,1);
lcd_putsf("Gate Open");
// counter-clockwise rotation
  PORTC = 0b00001010;
  delay_ms(2000);        // wait 2s
  PORTC = 0b00000000;
```

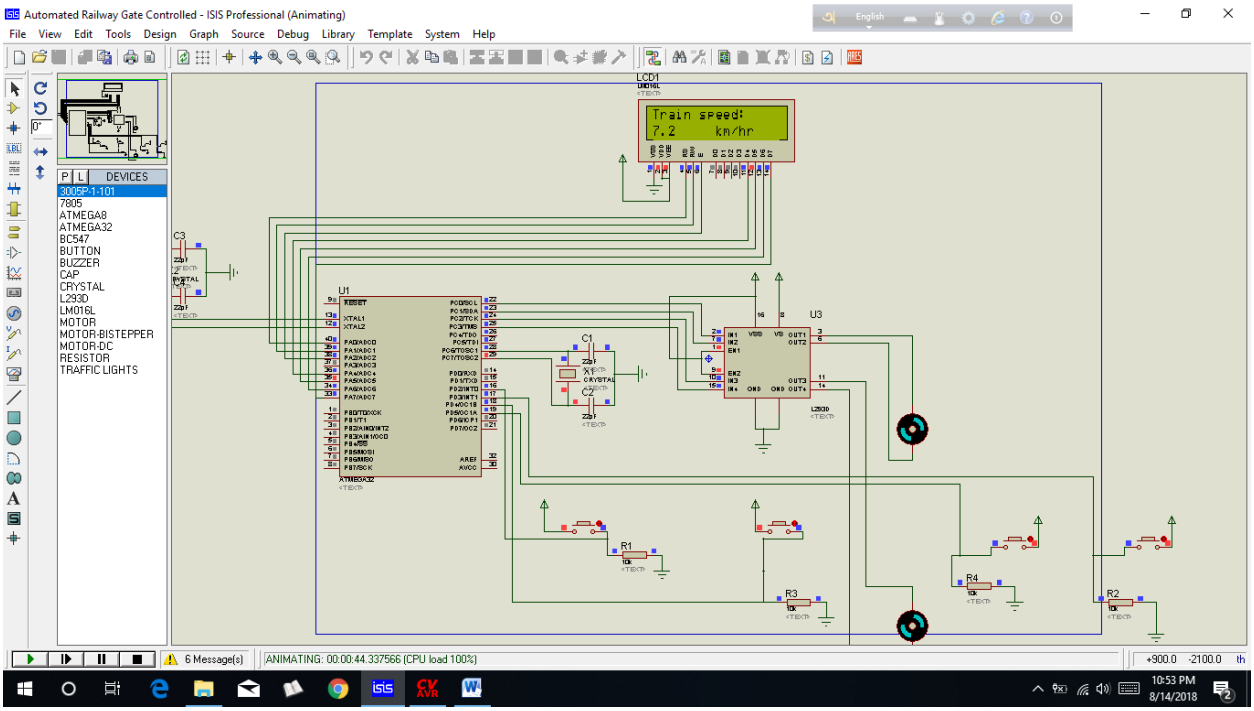
```
        break;
    }}}
}}
```

Output of the project in Proteus:

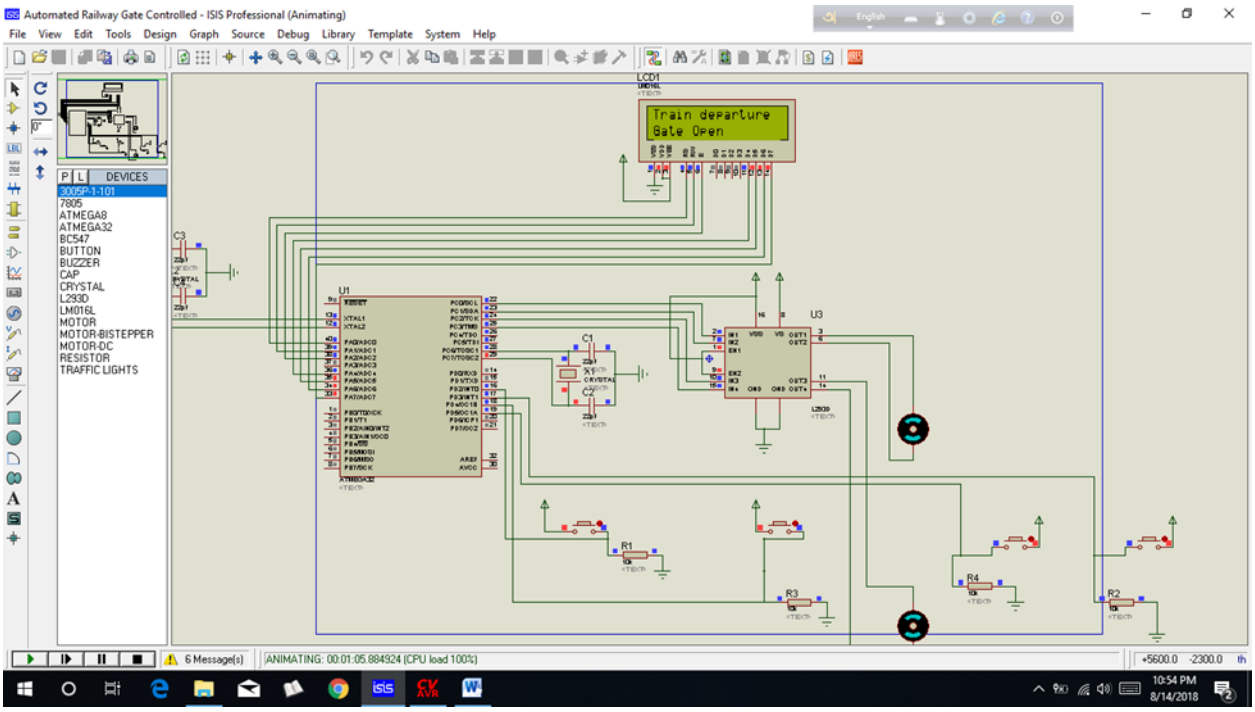
1st pressure sensor detect by pressing 1st button:



2st pressure sensor detect to speed calculation by pressing 2st button:



last pressure sensor detect to gate open by pressing last button:



Advantages and Applications:

- An Automatic Railway Gate Control is implemented with very simple hardware and easy control.
- Human intervention at level crossings can be removed with the help of this project and many railway level crossing accidents can be prevented.

Limitations:

- The system can be implemented more efficiently by incorporating more efficient sensor network.
- A combination manual wireless control and sensors based control can be used for better operation.

Scope of Future works :

This project can be more developed in future.

Conclusion: Automatic gate control system offer an effective way to reduce the occurrence of railway accidents. This system can contribute a lot of benefit either to the road users or to the railway management. Since the design is completely automated it can be used in remote villages where no station master or line man is present. Railway sensors are placed at two sides of gate. It is used to sense the arrival and departure of the train. This system uses the DC motor to open and close the gates automatically when it is rotated clockwise or anticlockwise direction. The LCD display shows the speed of The train. In this system, this is controlled by using ATmega32 microcontroller. Now a day's automatic system occupies each and every sector of applications as it is reliable and accurate.