A PROJECT ON AUTOMATIC RAILWAY GATE CONTROL SYSTEM

A Project submitted in partial fulfillment of the requirements
for the Award of Degree of
Bachelor of Science in Electrical and Electronic Engineering

Submitted By

Md. Sayed Mortoza

ID: 171-33-3916

Md. Maruf Hasan

ID: 171-33-3917

Supervised by

Sutapa Debnath

Lecturer

Department of EEE

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

FACULTY OF ENGINEERING

DAFFODIL INTERNATIONAL UNIVERSITY

Certification

This is to certify that this project entitled "**A PROJECT ON AUTOMATIC RAILWAY GATE CONTROL SYSTEM**" is done by the following students under my direct supervision and this work has been carried out by them in the laboratories of the Department of Electrical and Electronic Engineering under the Faculty of Engineering of Daffodil International University in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering. The presentation of the work was held in May 2021.

Signature of the candidates

Mortoza

____

Name: Md. Sayed Mortoza

ID: 171-33-3916

maruf

____

Name: Md. Maruf Hasan

ID: 171-33-3917

Countersigned

Sutapa Debnath

____

Sutapa Debnath

Lecturer

Department of Electrical and Electronic Engineering

Faculty of Engineering

Daffodil International University.

The project entitled **“A PROJECT ON AUTOMATIC RAILWAY GATE CONTROL SYSTEM”** submitted by Md. Sayed Mortoza ID: 171-33-3916, Md. Maruf Hasan ID: 171-33-3917, Session: Fall 2020 has been accepted as satisfactory in partial fulfillment of the requirements for the degree of **Bachelor of Science in Electrical and Electronic Engineering** on January 2021.

BOARD OF EXAMINERS

____

Dr. Engr. Chairman
Professor
Department of EEE, DIU

____

Dr. Engr. Internal Member

Professor
Department of EEE, DIU

____

Dr. Engr. Internal Member

Professor
Department of EEE, DIU

Dedicated to

Our Parents

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENT

First of all, we give thanks to Allah. Then we would like to take this opportunity to express our appreciation and gratitude to our project supervisor **Sutapa Debnath, Lecturer,** of **Department of EEE** for being dedicated in supporting, motivating and guiding us through this project. This project can't be done without her useful advice and helps. Also thank you very much for giving us opportunity to choose this project.

Apart from that, we would like to thank our entire friends for sharing knowledge, information and helping us in making this project a success. Also thanks for lending us some tools and equipment.

To our beloved family, we want to give them our deepest love and gratitude for being very supportive and also for their inspiration and encouragement during our studies in this University.

ABSTRACT

In this project the gate will be controlled by a microcontroller which automatically opens, closes, and signaling. There will be a signal on the road for vehicles when the train arrives. Then the rail crossing will close. So, we used a weight sensor which increases the security of rail crossing a bit more. When, the train passes over the weight sensor it will send a signal to the road. So that the drivers of vehicles can understand the rail crossbar will be closed in a short while. But sometimes, we do not notice the road signals due to our unconsciousness. So that time, if a vehicle gets stuck at a rail crossing, a red light will flash on the rail line. So the train driver realizes that a vehicle is stuck at the rail crossing. At the same time, one more signal will go to the control room, so that this problem can be solved from the control room. After disconnecting the power from the control room and restarting it, the rail crossing will be open and the vehicle stuck in the rail crossing will be able to pass over the rail crossing. Then the train will be able to move easily to its own destination.

CHAPTER 1

INTRODUCTION

1.1 Introduction

This project is a digital railway crossing system made with an Arduino UNO, IR sensors, Relay and few other components. It works through some signals. The main goal of the project is to protect rail crossing from accidents.

Every year many people die in rail crossing due to unconsciousness. And many rail crossings are not safe because gates are not used there. That's why we can use digital rail crossing system.

So we have tried to build a rail crossing system in the primary way.

When the train comes in front of the IR sensor, the rail crossing is opened or closed which is controlled by Arduino.

1.2 Objectives

The purpose of the project must be clear and understandable in order to achieve it. So, the objectives of this project are:

i. To develop a program that will deliver signals through sensors.
ii. To create a program that can control rail crossbars.
iii. To create some IR sensors that give signals when a vehicle is stuck at a rail crossing.
iv. To create an easy and effective way to protect trains or vehicles from accidents.
v. To create a weight sensor to transmit signals from the train line before the train arrives.
vi. Digitalize all the rail crossings.

1.3 Scope of the Project

The purpose of the project is to create a system that can detect trains and using a signal can give a warning. The system analyzes the generated signals and detects trains and reflected waves, controlling the lights, signals and gates. Relays also have been used in this project to send another signal.

Every year many people die in train accidents due to lack of crossbars. Here are some pictures of train accident in our country.

Figure: 1.1 Pictures of Train Accident

This project implements by microcontroller, IR sensors and relays. It will work without any operator. Before the crossbar closes a signal will come to the road from the relay. And also, if a vehicle gets stuck at a rail crossing, another signal will come to the train line from the road.

As a result, there is no possibility of an accident.

1.4 Methodology

Basically, the main task of this project is controlling crossbars.

IR sensors and relay module have been used in this project to prevent accidents. Some IR sensors are controlled by an Arduino.

1.5 Project Outline

Chapter 1-Introduction:

This project is a digital railway crossing system made with an Arduino UNO, IR sensor, Relay, and few other components. The aim of the project is to detect the presence of a train on the railway line, control the crossbar, send a signal on the road before the train arrives, and also send a signal on the train line if any vehicle gets stuck on the road.

Chapter 2-System Reviews:

We were needed a lot of components to complete this project. These elements are discussed in detail in this chapter 2. And also discussed how those elements work in this project.

Chapter 3-Theoretical Model:

The process of making IR Sensor, Weight Sensor is discussed in this chapter. And the ingredients needed to make the Arduino part are also discussed.

Chapter 4-Software Analysis and Hardware Implementation:

This chapter discusses Arduino IDE software. And how to upload code from this software is explained in detail.

Chapter 5-Result and Discussions:

This chapter discusses how it works after connecting all the equipment. And a list of how much money we spent to buy these components is also shown in this chapter.

Chapter 6-Conclusions:

This Chapter discusses the conclusion, advantages, disadvantages, limitations, and future scopes of the work.

CHAPTER 2

SYSTEM REVIEWS

2.1 Introduction

This chapter will discuss all the components used in this project.

2.2 Component List

1. Arduino UNO
2. IR Sensor
3. Servo Motor
4. Buzzers
5. LED
6. Weight Sensor
7. Relay
8. Connector
9. Vero Board
10. Battery
11. Switch
12. Adapter

2.3 Arduino UNO

Arduino UNO is a microcontroller device based on hardware and software which is an open-source platform. Arduino devices can read inputs such as an object on the sensor, finger on the button and transfer them as an output to send a signal on another device.

Figure: 2.1 Arduino UNO

2.3.1 Working Principle

This project is microcontroller-based which is why Arduino is being used in this project. Two IR sensors, two servo motors and a buzzer have been used for this project which is connected with the Arduino. The IR sensors, Servo motors and the buzzer are controlled with specially developed code through the Arduino.

2.4 IR Sensor

The IR sensor or Infrared sensor is a device that measures and detects infrared radiation within its search area. The IR sensor works when an object is close to the sensor, the infrared light bounces off the object and is received by the receiver.

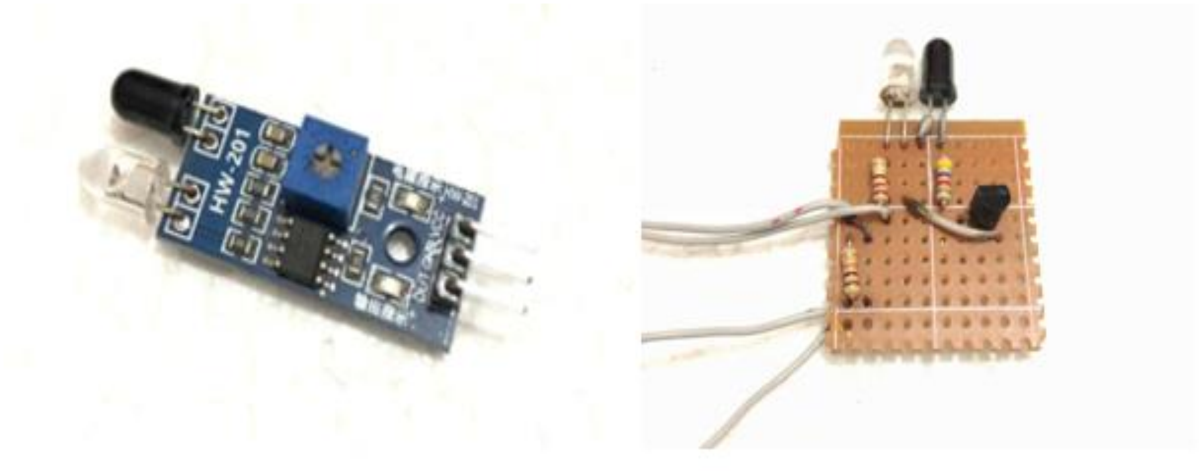


Figure: 2.2 Readymade and Handmade IR sensor

2.4.1 Working Principle:

There are Four IR sensors in this project. In the first IR sensor, there are three pins which are VCC, Ground, and output. The VCC pin is connected to a 5V pin of Arduino, the ground is connected to the Arduino ground and the output pin is connected to the 10th pin of the Arduino.

In the 2nd IR sensor, the VCC and ground are connected the same way but the output is connected to the 11th pin of the Arduino. Both the IR sensor is used to detect an object and send the signal to the Arduino.

The third and fourth IR sensor is homemade for this project. These are used to detect an object on the rail line and send a signal to two LED's.

2.5 Servo Motor:

The Servo motor is an electrical device that is used to move different parts of a machine with high precision and great efficiency. In this project, the servo motor is controlled by the Arduino with specially developed code. Two servo motor is being used for this project.

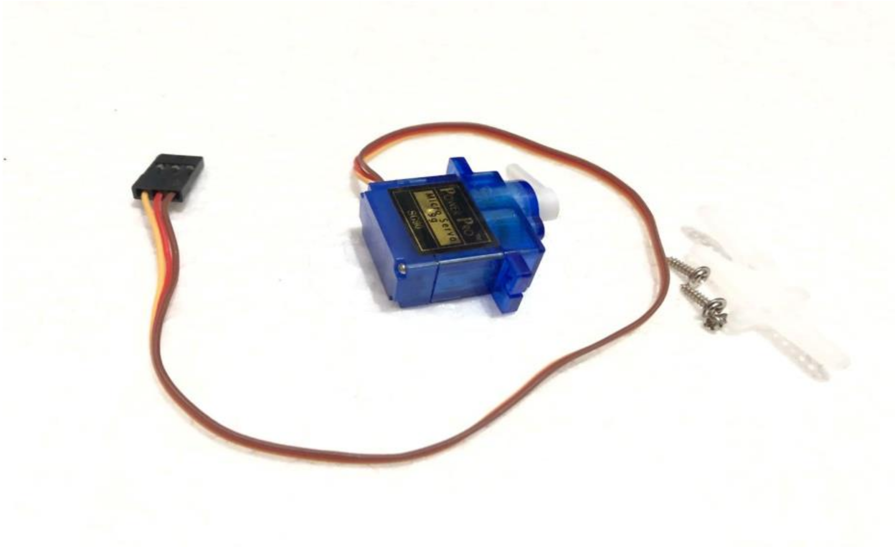


Figure: 2.3 Servo motor

2.5.1 Working Principle:

In this project, the servo motor is being used for opening and closing a railway crossing gate. This servo motor has three pins. In both the servo motors, the VCC and Ground pin is connected individually to the Arduino but the output is combinedly connected to the 9th pin of Arduino. When the first IR sensor detects an object it sends a signal to the Arduino to close the railway crossing gate and when the 2nd IR sensor detects an object it sends a signal to the Arduino to open the railway crossing gate.

2.6 Buzzer:

The buzzer is an electronic device that makes a sound when it receives a signal from the Arduino. A buzzer has two pins. One pin is connected to the 13th pin of the Arduino and the other is connected to the ground. When the IR sensor detects an object it sends the signal to the Arduino and the Arduino sends the signal to the buzzer.

Figure: 2.4 Buzzer

2.7 LED (Light Emitting Diode):

In this project 3 LED's have been used. One of the LEDs is placed at a safe distance for the train conductor to see and avoid any danger. This LED is activated when an IR sensor detects an object. The 2nd LED is placed in the control room and works in the same manner. The 3rd LED is activated by the weight sensor.

Figure: 2.5 LED

2.8 Weight Sensor:

In this project, we used a weight sensor which increases the security of rail crossing a bit more. We will put this sensor under the rail line which will turn on the LED light on the road with the help of relay, and this will helps the drivers to understand that anytime a train will pass this rail crossing. This LED light will help us with one more thing along with the first one, it will help the drivers to understand that the rail crossing bar will get down very soon. Because we will install the weight sensor machine at a safe distance from the IR sensor which will help close the gate. This will prevent the vehicles to come on the rail line.

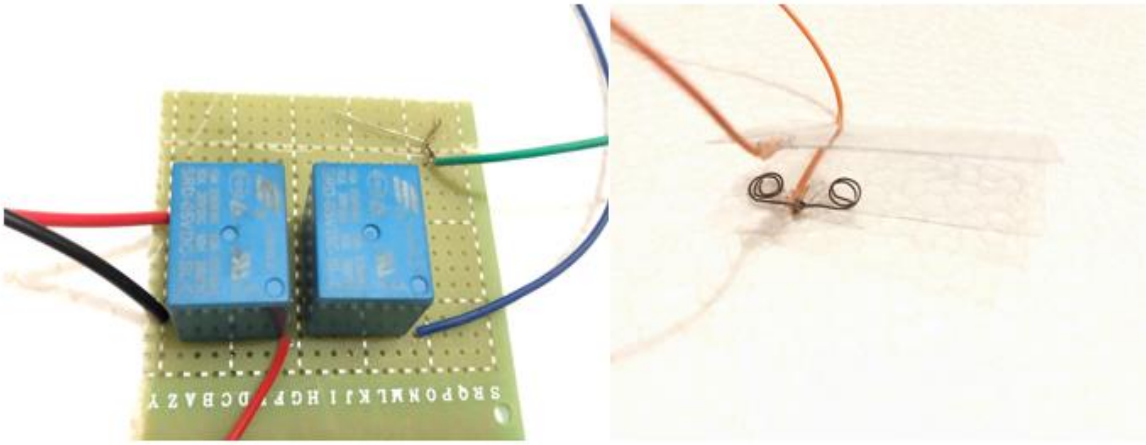


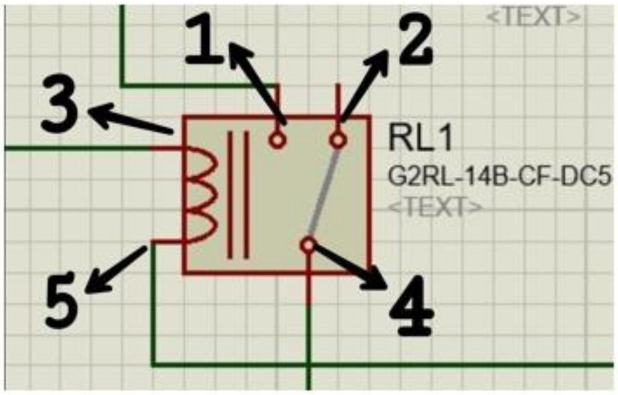
Figure: 2.6 Relay and Sensor

2.8.1 Relay Latching:

Meanwhile, we know that how weight sensor works. Now we will get informed about the weight sensor internal mechanism.

There are two relays in this weight sensor, Relay 1 and Relay 2.

Relay 1 and 2 both has 5 pins. Relay 2's 2nd pin is connected to the 4th pin and the 4th pin is connected to the battery's positive. As relay 2's 4th pin is connected to relay 1's 4th pin. So, it is obvious that relay 1's 4th pin is also connected to the battery's positive. Familiarly battery's negative is connected to relay 1's and 2's 5th pin.



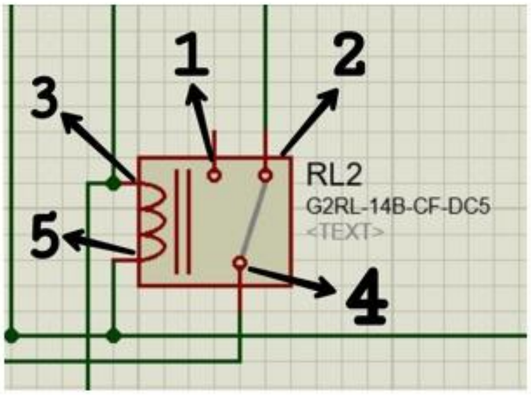


Figure: 2.7 Relay 1

Figure: 2.8 Relay 2

We used two sensors in this project.

In relay 1, pin number 1 and 3, 2 and 4 will remain connected until a train passes over sensor 1. When the train passes over sensor 1, this time pin number 1 and 4 will be connected and pin number 2 will be disconnected. Then the LED will turn on. This LED will take a very short time to turn on.

As relay 1 and relay 2 are connected so the LED will be turned on till relay 2 is disconnected. And relay 2 will only be disconnected when the train passes over sensor 2. After passing the train through sensor 2, this time in relay 1, pin number 2 and 4 will be disconnected. And again it will be connected within a very short time. In this short time Relay 1's pin number 1 and 4 will be disconnected and 2 and 4 will be connected.

So we can say, relay 1 and relay 2 will go back to their previous positions in this short time.

We need to connect the pins of relays 1 and relay 2 very carefully. Because, if we make a mistake in connecting the pins, it will not work.

2.9 Connector:

The connector is a set of wires that is used to connect or disconnect a circuit path. Every device in this project is connected with connectors.

Figure: 2.9 Connector

2.10 Vero Board:

In this project, Vero Board has been used for two reasons.

1. It is used for making home-made IR sensors.
2. It is used for Relay Latching.
3. It is used to connect the pins of Arduino to IR sensors, LED, Buzzer, and Servo Motor.

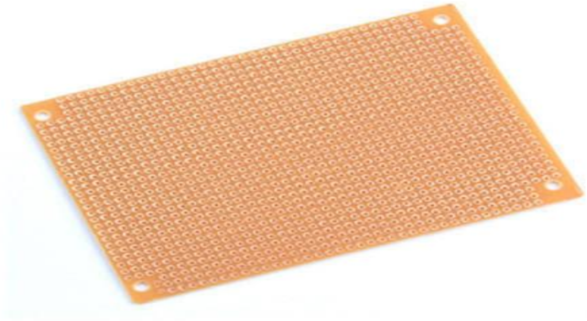


Figure: 2.10 Vero Board

2.11 Battery:

Two 9V battery has been used in this project to power the home made IR sensors and Weight Sensor.

Figure: 2.11 Battery

2.12 Switch & Adapter:

In this whole project a switch and a adapter has been used to control and supply DC power to the Arduino. The switch is used to control the adapter for the power supply of the arduino.

Figure: 2.12 Switch & Adapter

2.13 Summary

In this chapter, we have discussed all the components that have been used in this project. And also discussed the main functions of these components and how these are working in our project.

CHAPTER 3

THEORETICAL MODEL

3.1 Introduction

In this chapter we will discuss about Component List, And All Diagrams.

3.2 List

1. IR Sensor
2. Weight Sensor
3. Arduino UNO

3.3. IR Sensor

We need two IR Sensors in this project. These are not ready-made IR sensors.

Some components are needed to make these. These are listed in Table (3.1) below.

Component Name	Quantity
IR LED	2
IR Receiver	2
220 ohm Resistance	2
4.7 k Resistance	2
1 k Resistance	2
BC548 Transistor	2
Red LED	2
Vero Board	2
Battery 9v	2

Table 3.1 Component List For Home-made IR Sensor

3.3.1 Circuit Diagram

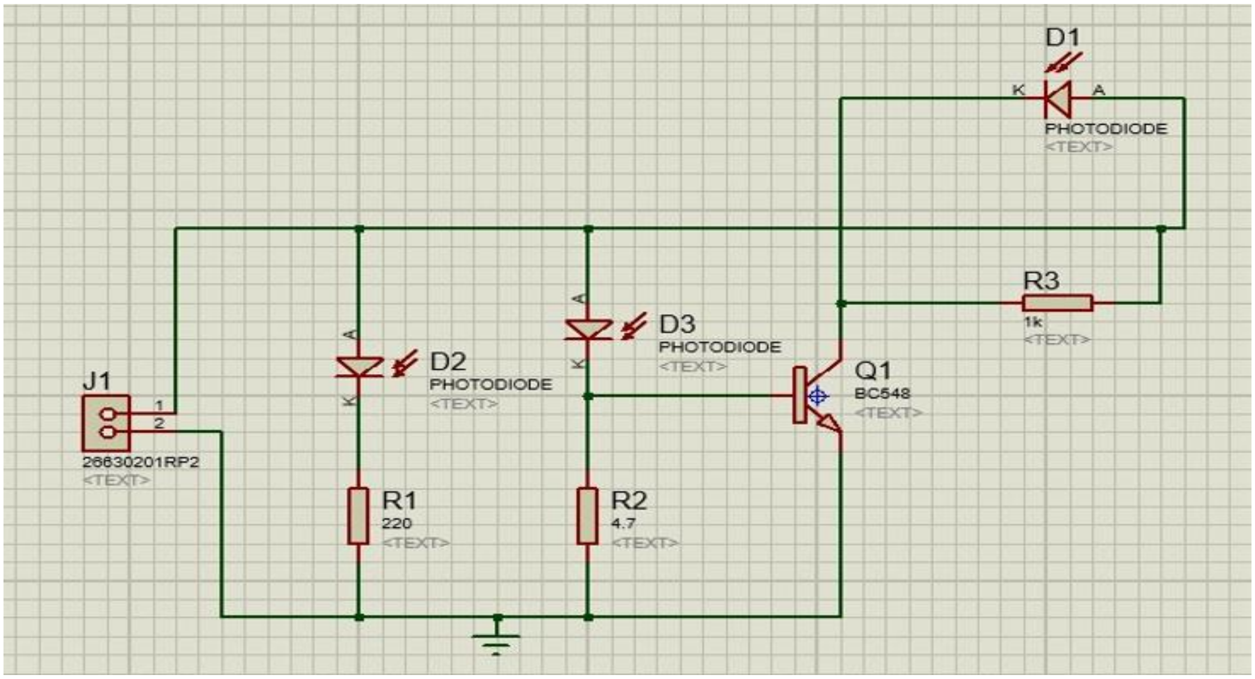


Figure: 3.1 Circuit diagram of Home-made IR sensor

From this circuit diagram, we can see that lot of components have been used there. Now it will be discussed in detail.

D1, D2, and D3 are Red led, IR led and IR receiver respectively. R1, R2 and R3 are 220 ohm resistance, 4.7k resistance and 1k resistance. J1 is a battery of 9v. And lastly, Q1 is a BC548 transistor.

This transistor has 3 pins. There are Emitter, Base and Collector.

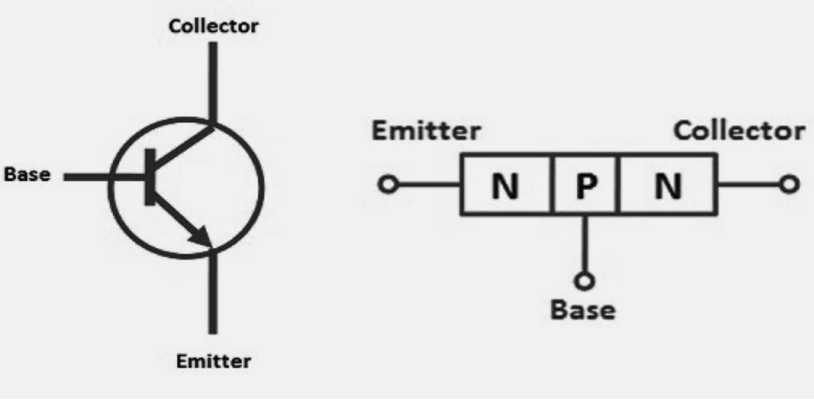


Figure: 3.2 NPN transistor symbol

The BC5748 transistor is connected to the IR receiver's negative and 4.7 k resistance on IR receiver’s negative to the emitter of the transistor. 220 ohm resistance on negative of IR led.

Another side, 1k resistance from positive of both IR led and IR receiver to the collector of the transistor. And red LEDs positive on VCC and negative on the collector.

This sensor will be used to detect street objects. As a result, if a vehicle gets stuck on the road, it can send a signal to the train line.

3.4. Weight Sensor

We need to make a weight sensor using two relays. Some components are needed to make this. These are listed in Table (3.2) below.

Component Name	Quantity
Relay	2
Wire	2 Meter
1 k Resistor	1
Red LED	1
Vero Board	1
Battery 9V	1

Table 3.2 Component List for Sensor using by Relay

3.4.1 Circuit Diagram

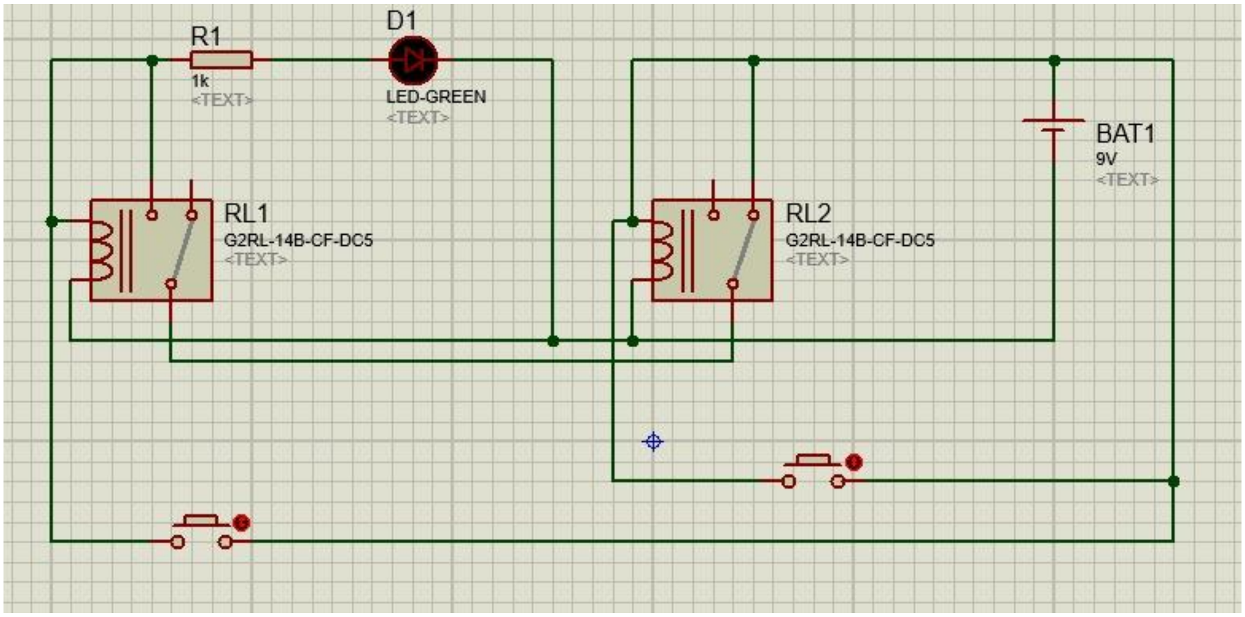


Figure: 3.3 Circuit diagram of Weight Sensor using the relays

The most important aspect of making this sensor is relay latching. And this is discussed in chapter 2 (2.8.1 Relay Latching).

3.4.2 Flowchart

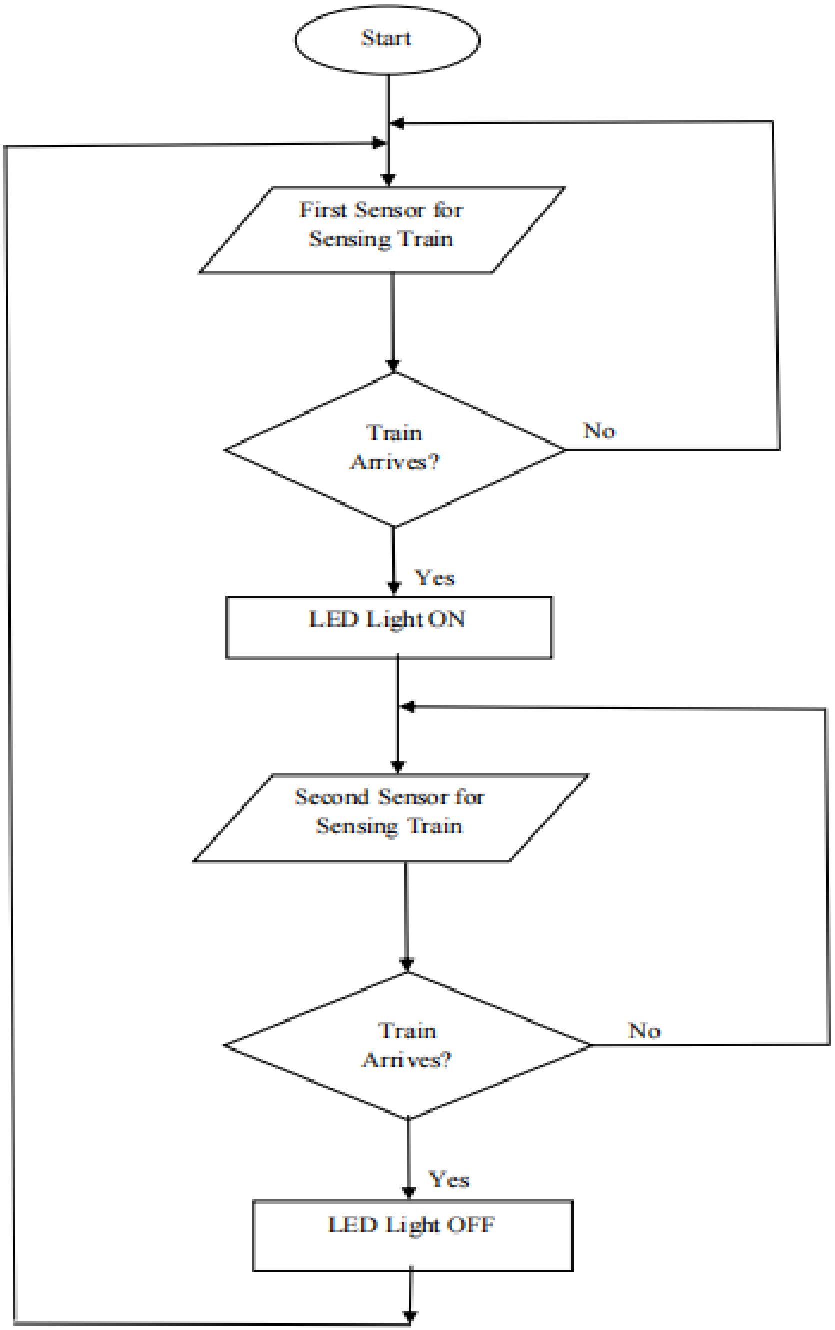


Figure: 3.4 Flowchart of Weight Sensor using the relays

At first, when the first sensor senses the train that means the train is coming, and that time led light will be on. After that, when the second sensor senses the train then the LED light will be off.

3.5 Arduino UNO

Arduino UNO is the main part of this project. Some components are needed to make this Arduino part.

Component Name	Quantity
Arduino UNO	1
IR Sensor	2
Servo Motor	2
Buzzer	1
Wire	5 Meter
Adapter (for power supply)	1

Table 3.3 Component List for Arduino Part

3.5.1 Block Diagram

The Arduino UNO will receive signals from the IR sensors. The Buzzer and Servo motors will act as the output.

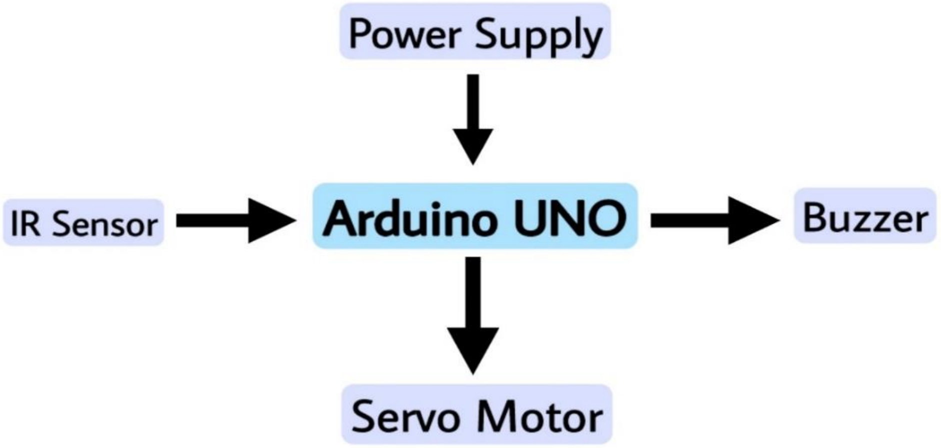


Figure: 3.5 Block Diagram

3.5.2 Circuit Diagram

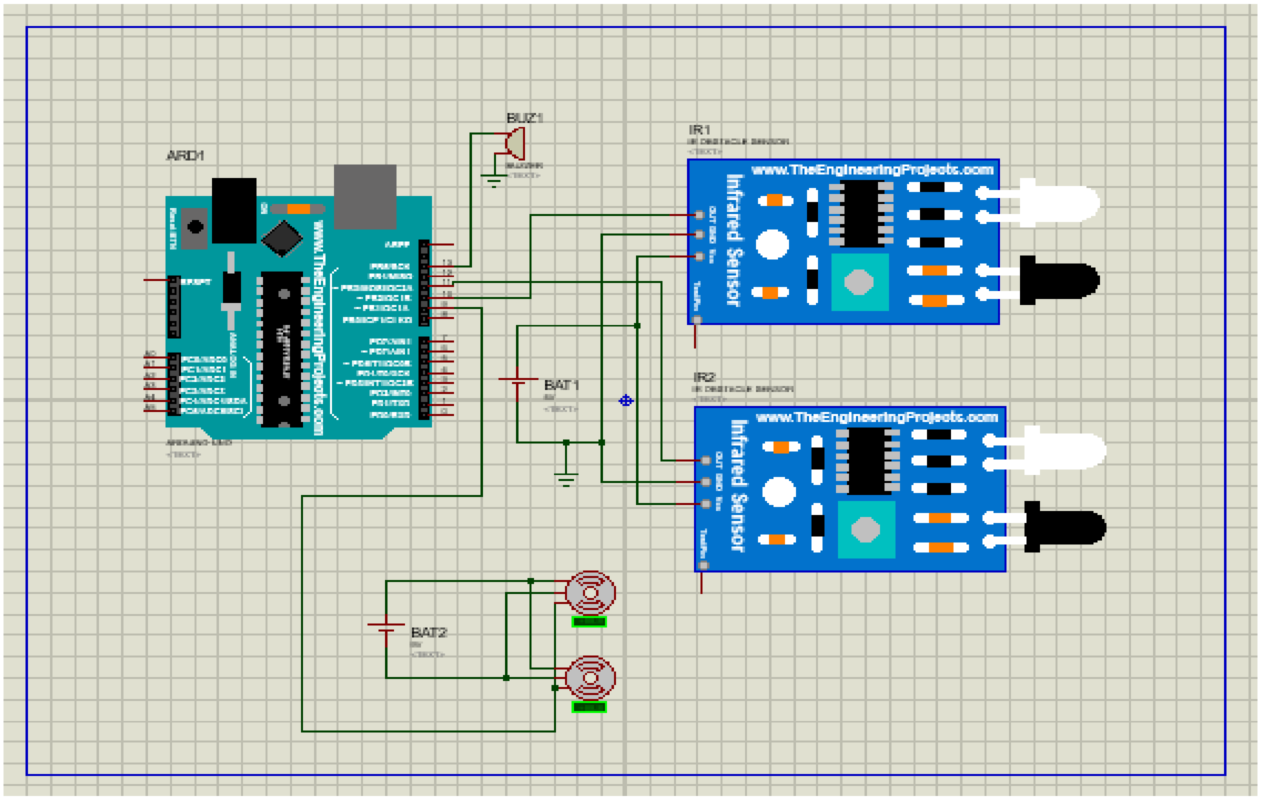

ARD1
BUZ1
IR1
IR2
BAT1
BAT2
www.TheEngineeringProjects.com
Infrared Sensor
www.TheEngineeringProjects.com
Infrared Sensor
www.TheEngineeringProjects.com

Figure: 3.6 Circuit Diagram of Arduino UNO

In this part, the pins are need to be connected carefully.

For example, Arduino digital pin number 13 must be connected to the Buzzer.

Pin number 10 and 11 must be connected to two separate IR sensors.

9 must be connected to servo motors.

An Adapter or a Battery must be connected to the Arduino for power supply. Otherwise, it doesn't work.

3.5.3 Flowchart

When IR sensor 1 detects the train, the IR sensor will send a signal to the Arduino and the Arduino will pass the signal to the servo motor and buzzer. Then the rail gate will be closed. When IR sensor 2 detects the train, the IR sensor will transmit the signal to the Arduino and again the Arduino will pass the signal to the servo motor and buzzer. Then the rail gate will open.

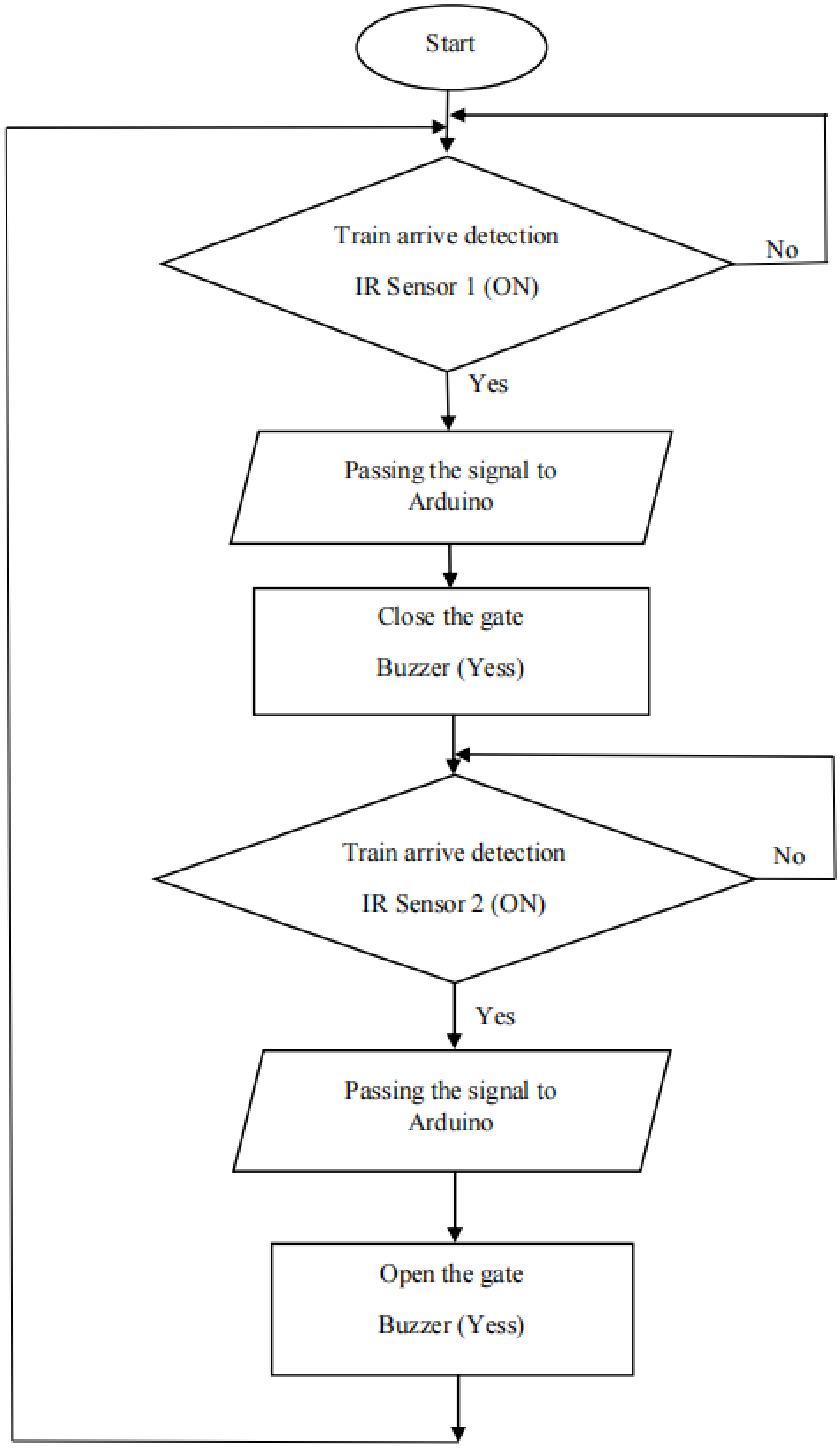


Figure: 3.7 Flowchart of Arduino UNO

3.6 Summary

Some materials and some tools have been used to create this project. And this chapter discusses all component lists, Circuit diagrams, and Block diagram.

CHAPTER 4

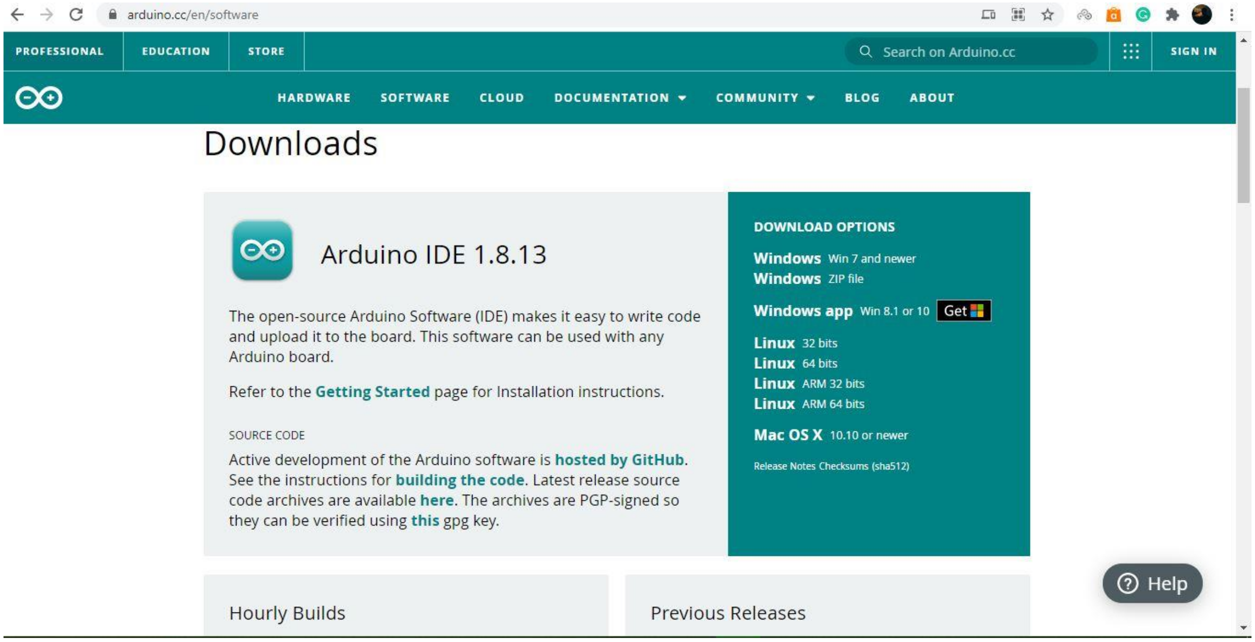
SOFTWARE ANALYSIS AND HARDWARE IMPLEMENTATION

4.1 Introduction

In this chapter, we will discuss the software used to upload coding to Arduino UNO. And we will also discuss how it works.

4.2 Preparing Arduino IDE

First, we need to download the Arduino IDE Software. Arduino IDE is available at the official website of Arduino. So, we have to go to this link, https://www.arduino.cc/en/software and we can download it easily.



Downloads

Arduino IDE 1.8.13

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the **Getting Started** page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is **hosted by GitHub**. See the instructions for **building the code**. Latest release source code archives are available **here**. The archives are PGP-signed so they can be verified using **this** gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file
Windows app Win 8.1 or 10 Get
Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits
Mac OS X 10.10 or newer

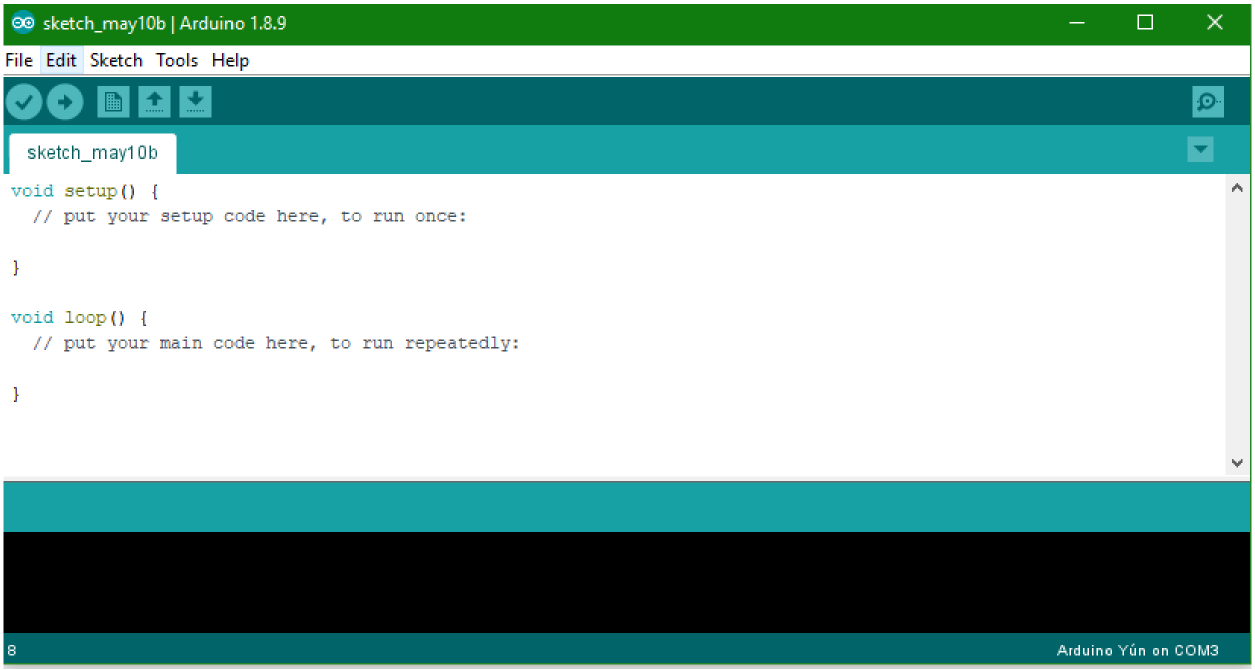
Release Notes Checksums (sha512)

Hourly Builds

Previous Releases

Figure: 4.1 After entering the link

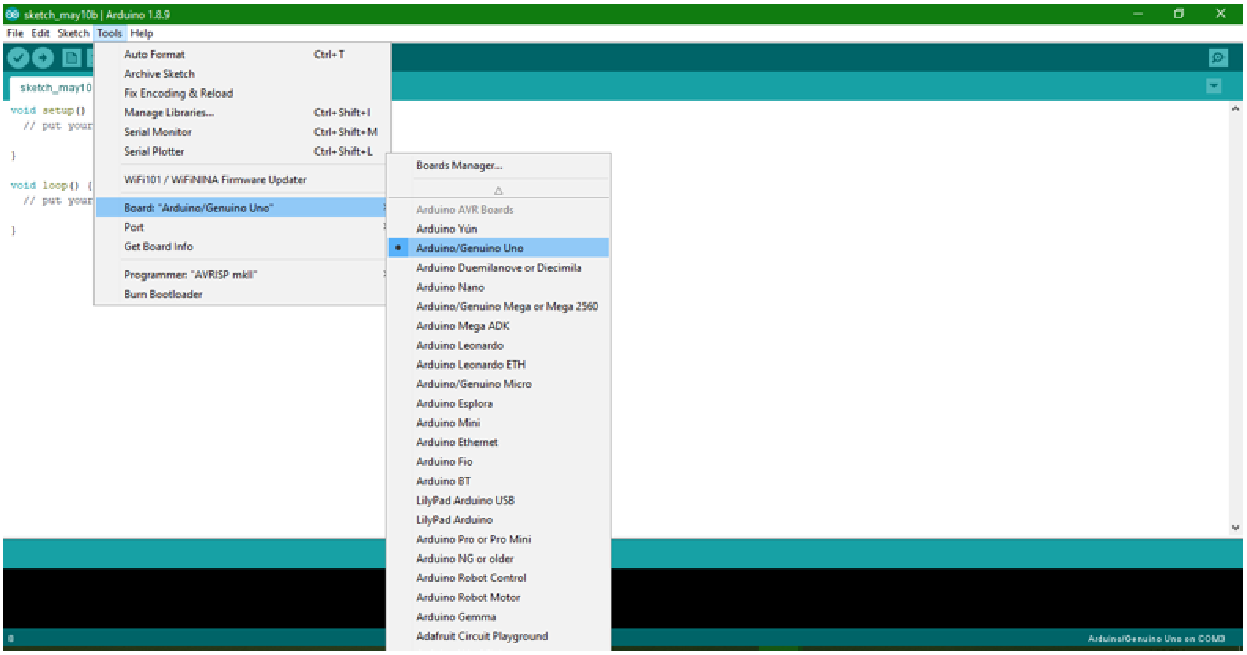
The software needs to be installed at the end of the download.



```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Figure: 4.2 Opening of Arduino IDE

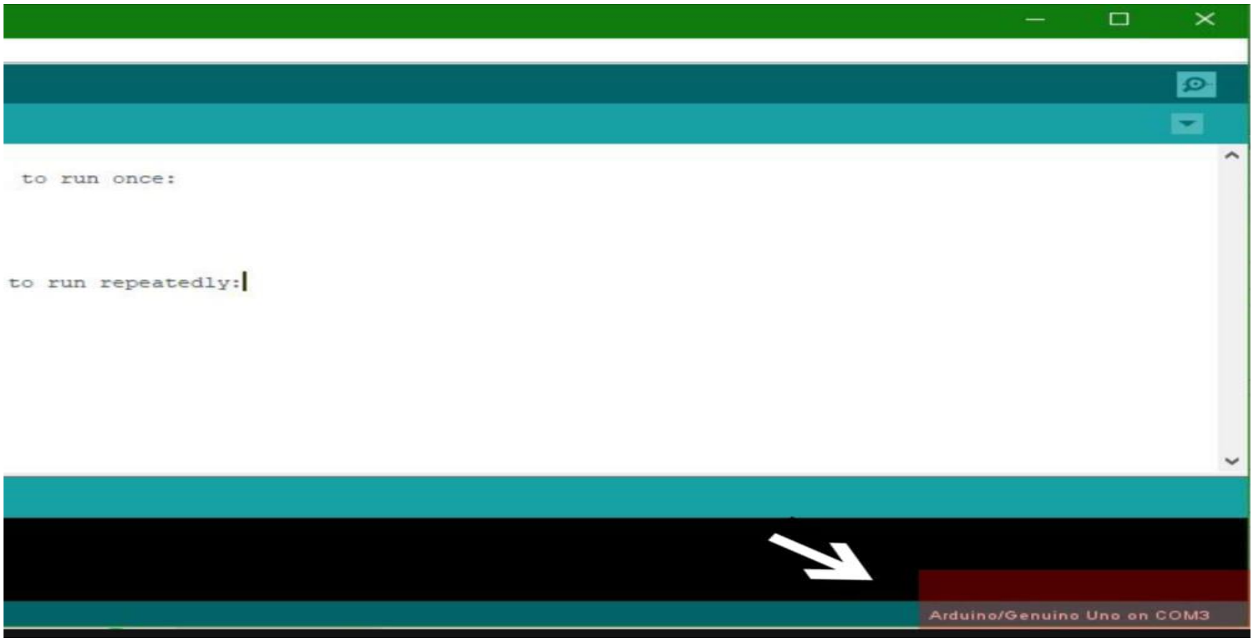
In this project we used Arduino UNO. So, we have to select Tools->Board ->Arduino Uno



```
void setup() {
  // put your

}

void loop() {
  // put your

}
```

Figure: 4.3 Select Arduino UNO

After selecting the Arduino UNO, we will see the selected board at the bottom of the software.



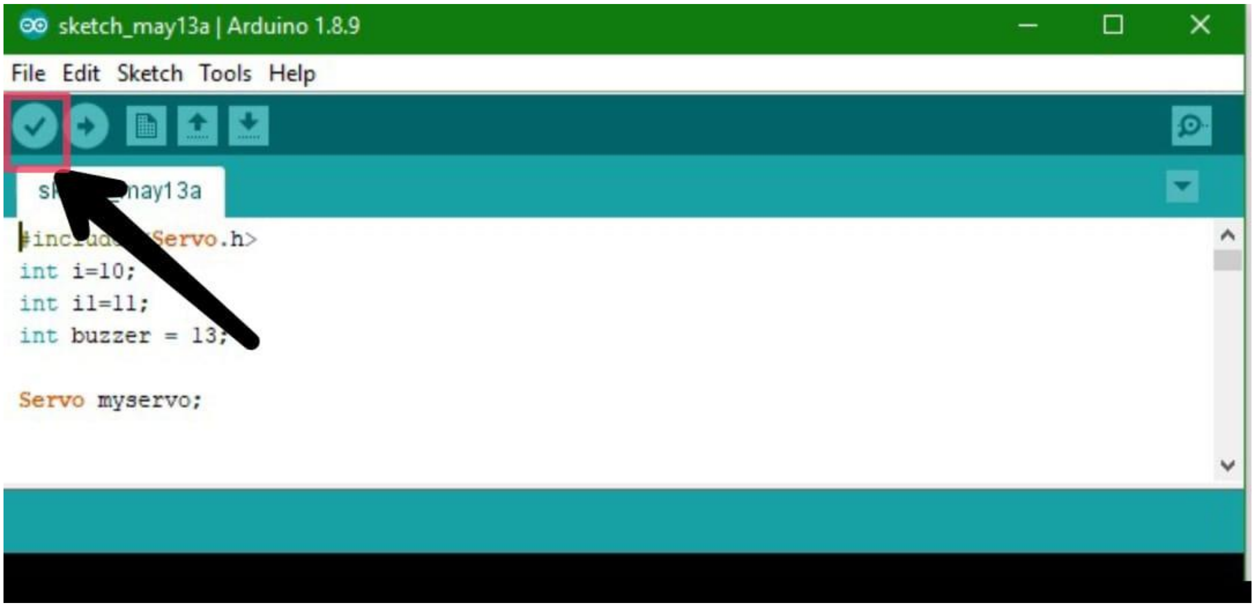
```
to run once:


to run repeatedly:
```

Figure: 4.4 After selecting the Arduino UNO

Now, this software is ready to work.

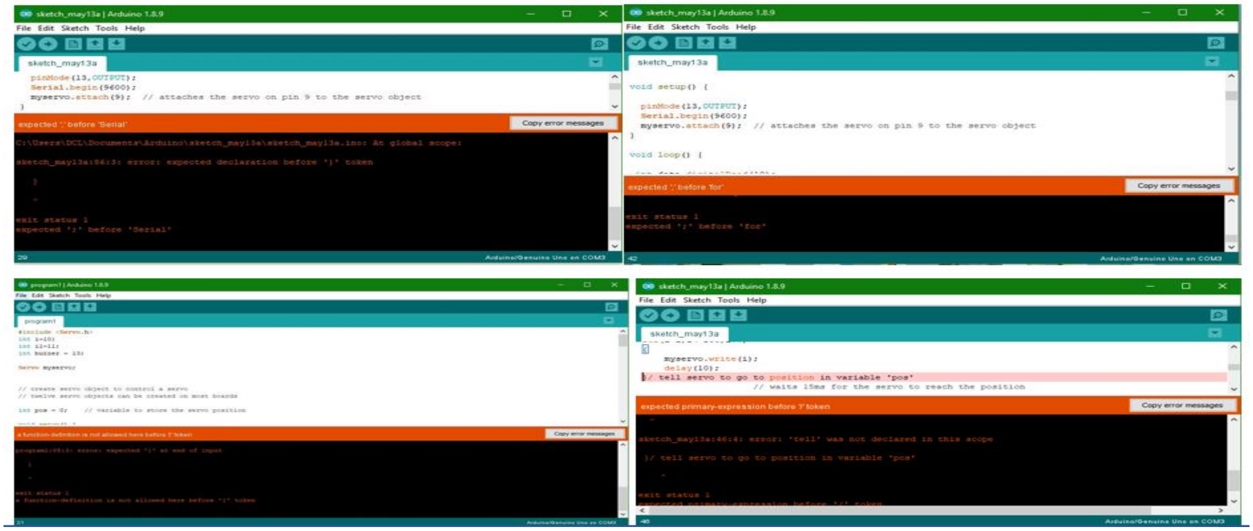
4.3 Program Writing

When we write a code, we will check if it is correct or not. Because, sometimes we make mistakes in coding. So, we need to click on the verify option to verify this.



```
#include <Servo.h>
int i=10;
int i1=11;
int buzzer = 13;

Servo myservo;
```

Figure: 4.5 Verify Option



sketch_may13a | Arduino 1.8.9

```
  pinMode(13,OUTPUT);
  Serial.begin(9600);
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}
```

expected ';' before 'Serial'

```
C:\Users\DCL\Documents\Arduino\sketch_may13a\sketch_may13a.ino: At global scope:

sketch_may13a:56:3: error: expected declaration before '}' token

   }

   ^

exit status 1
expected ';' before 'Serial'
```

29 — Arduino/Genuino Uno on COM3

sketch_may13a | Arduino 1.8.9

```
void setup() {

  pinMode(13,OUTPUT);
  Serial.begin(9600);
  myservo.attach(9);  // attaches the servo on pin 9 to the servo object
}

void loop() {
```

expected ';' before 'for'

```
exit status 1
expected ';' before 'for'
```

42 — Arduino/Genuino Uno on COM3

program1 | Arduino 1.8.9

```
#include <Servo.h>
int i=10;
int i1=11;
int buzzer = 13;

Servo myservo;

// create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position
```

a function-definition is not allowed here before '{' token

```
program1:[illegible] error: expected '}' at end of input

  }

  ^

exit status 1
a function-definition is not allowed here before '{' token
```

21 — Arduino/Genuino Uno on COM3

sketch_may13a | Arduino 1.8.9

```
    myservo.write(i);
    delay(10);
}/ tell servo to go to position in variable 'pos'
                // waits 15ms for the servo to reach the position
```

expected primary-expression before '/' token

```
  ^

sketch_may13a:46:4: error: 'tell' was not declared in this scope

 }/ tell servo to go to position in variable 'pos'

    ^

exit status 1
expected primary-expression before '/' token
```

46 — Arduino/Genuino Uno on COM3

Figure: 4.6 Pictures of some errors

After uploading the code, Sometimes arduino does not run properly. This means there is something wrong with the coding. This time we need to re-check every line of coding. When we find an error, we will read the text of the 'error' option.

So, we have to write the code very carefully.

4.4 How to upload coding

It is very easy to upload. First we need to make sure that, there are no error items when we verify it. Then we will connect the Arduino UNO to the PC or Laptop by cable. After connecting to the PC or Laptop, a LED light will flash on the Arduino UNO. That means, the Arduino UNO has been connected. Then select the upload option. We will see the upload option at the top of the software.

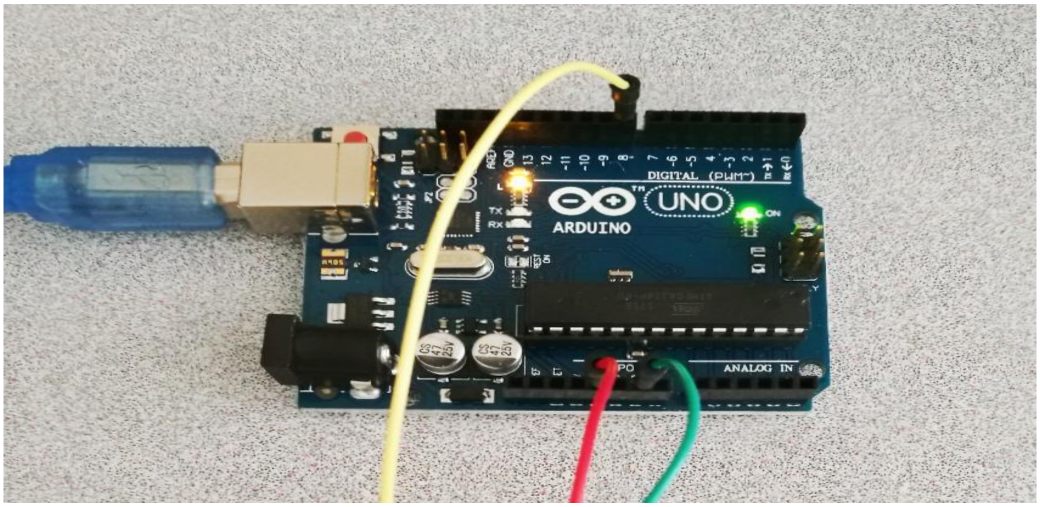
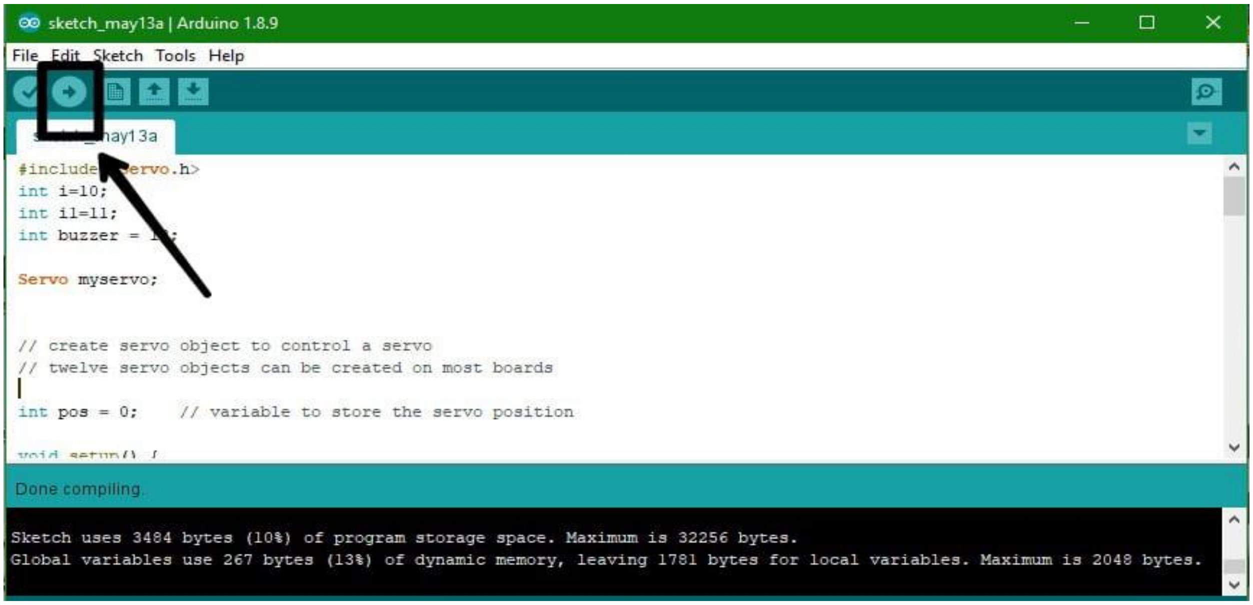


Figure: 4.7 After connecting the cable



```
#include <Servo.h>
int i=10;
int i1=11;
int buzzer = 1[illegible];

Servo myservo;


// create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position

void setup() {
```

Done compiling.

```
Sketch uses 3484 bytes (10%) of program storage space. Maximum is 32256 bytes.
Global variables use 267 bytes (13%) of dynamic memory, leaving 1781 bytes for local variables. Maximum is 2048 bytes.
```

Figure: 4.8 Select Upload Option

4.5 Final View

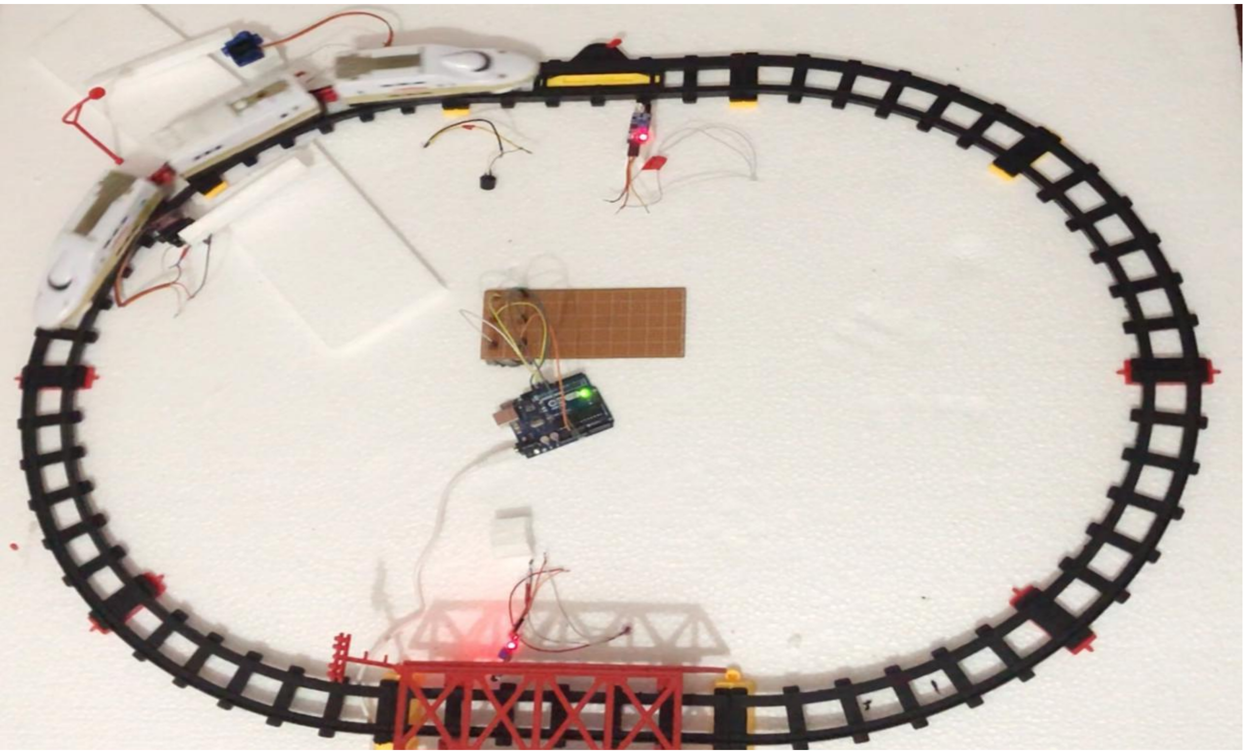


Figure: 4.9 Final View

After connecting all the components, we will run this project to see if there is any problem or not. If this project is run properly, we can understand all the components are properly attached.

4.6 Summary

We need Arduino IDE Software to upload code in Arduino UNO. So, we have discussed about this software and how it works.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 Introduction

In this chapter, we will discuss all the result of this project.

5.2 Our Project

This project is ready to run after connecting all the equipment.

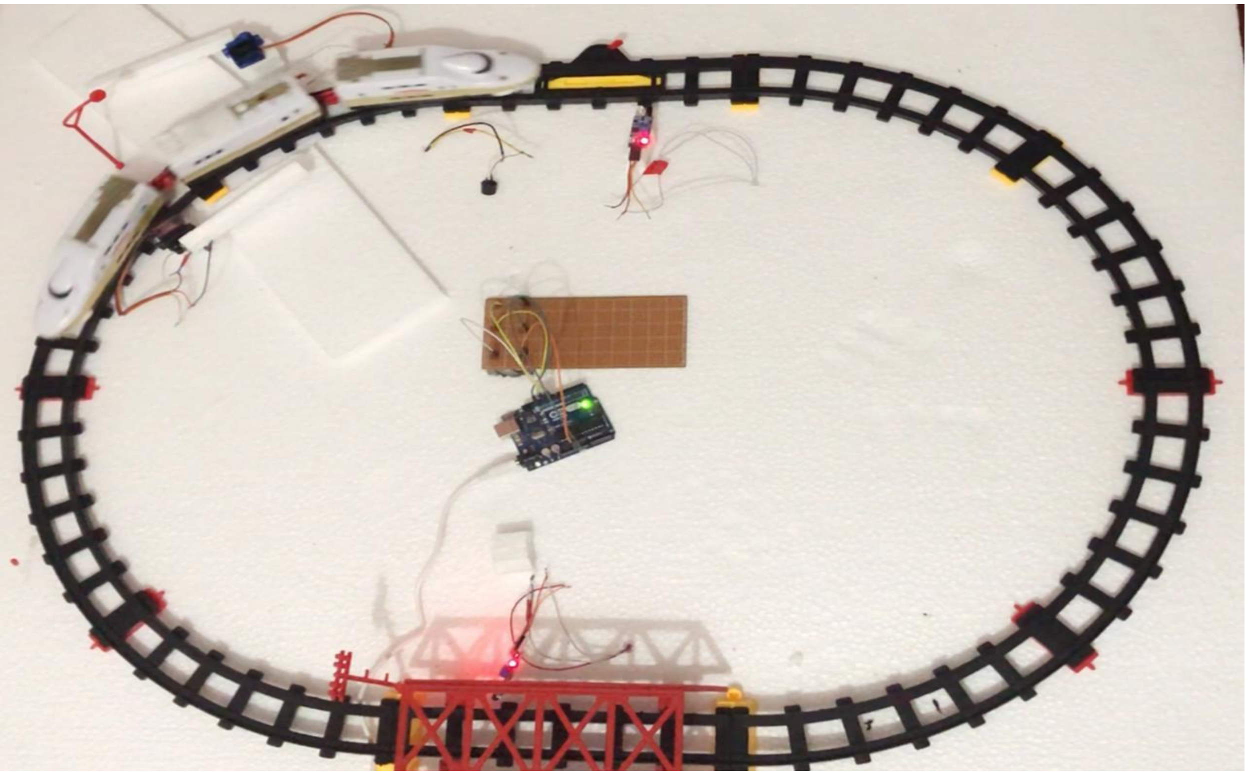


Figure: 5.1 Our Project

When we connect the power supply to the Arduino UNO, we hear a little bit of sound from the buzzer.

5.3 Result

The name of this project is Automatic railway gate control system with using Arduino UNO, IR Sensor and Relay. The main goal of this project is to protect us from train accidents.

In addition to Arduino, we have added weight sensor, homemade IR sensor to make this project more efficient. These have been used for warning signals.

The weight sensor will be placed some distance away before the IR sensor which is attached to the Arduino. We will place this sensor under the rail line which will turn on the LED light on the road with help of the relay and it will help the drivers to understand that at any time the rail crossing will be closed and the train will cross the rail crossing. When the train will come in front of the IR sensor after the LED light on the road is turned on. Then the IR sensor sends a signal to the Arduino. And then the Arduino sends a signal to the servo motor and buzzer. This time the rail crossing will be closed.

If a vehicle gets stuck at a rail crossing while the rail crossing is closed, this time a red light will flash on the rail line. So that the train driver realizes that vehicle is stuck at the rail crossing.

At the same time, one more signal will go to the control room to solve this problem. After disconnecting the power from the control room and restarting it, the rail crossing will be open and the vehicle stuck in the rail crossing will be able to pass over the rail crossing. Then the LED light will be turned off. Then the train will be able to move easily to its own destination.

The second weight sensor is placed after the second IR sensor which is attached with Arduino. When the train passes over the second weight sensor, the LED light that is at the rail crossing is turned off. Now vehicles will be able to cross the rail crossing after the LED light is turned off.

We need to be more vigilant to avoid accidents and we must follow the rules of the road.

5.4 Cost Analysis

Serial	Components	Price in (BDT)
1	Arduino UNO	350/-
2	IR Sensors	300/-
3	Servo Motors	300/-
4	Connecting wire	100/-
5	Buzzer	60/-
6	Relay	80/-
7	Connectors	100/-
8	IR LED	10/-
9	IR Receiver	10/-
10	220 ohm Resistance	10/-
11	4.7 k Resistance	10/-
12	1k Resistance	10/-
14	BC548 Transistor	15/-
15	Red LED	15/-
16	White Sponge Board	200/-
17	Glue Gun Stick	100/-
18	Vero Board	90/-
19	Toy Train	600/-
20	Battery 9v	60/-
21	Pencil Battery	30/-
22	Adapter	80/-
23	Other	100/-
	Total Cost	**= 2630/-**

Table: 5.1 Cost Analysis

5.5 Discussion

We have tried to build a rail crossbar in an easy way through this project. And also tried to add something new to this project.

For example, we have used homemade IR sensors to detect road vehicles. And also we built a weight sensor using the relay that sends a warning signal before the crossbar closes. This will help us from accidents.

5.6 Summary

The results and discussion are discussed in this chapter. And it also listed that what was needed to build this project and how much it cost.

CHAPTER 6

CONCLUSIONS

6.1 Conclusions

Some rail crossings in our country have crossbars and those are controlled manually. But most of the rail crossings do not have any crossbar.

Because of this, every year we face many accidents. So, every rail crossing in our country needs to be digitalized. This will save many people from accidents.

So, we have tried to give an initial idea about the importance of rail crossing through this project.

6.2 Limitations

We have faced some limitations and challenges in this project. There are,

1. If any sensor does not work, there will be a big problem.
2. The IR sensor works at a certain distance. If the distance increases, the sensor will not work properly.
3. The power supply cannot be cut off without any reason.

6.3 Advantages

1. We can make it easily.
2. Automatic rail gate control systems reduce the time for gates to remain closed.
3. It will help to protect us from accidents.
4. The rail gate opens and closes automatically after the arrival or departure of the train is detected.

5. The weight sensor will send a signal before the crossbar closes.
6. Before closing the crossbar, a red light will on at the railway crossing.
7. The IR sensor will send a signal to the rail line if a vehicle gets stuck at a rail crossing.

6.4 Future Scopes of the Work

1. The system can be implemented more efficiently by incorporating more efficient sensor networks.
2. LED displays can be used at rail crossing gates.
3. We can use GSM module to make the project more efficient.

REFERENCES

[1] https://www.arduino.cc/

[2] https://www.electronicshub.org/automatic-railway-gate-controller/

[3] https://engineering.eckovation.com/automatic-railway-gate-control-system/

[4] https://engineering.eckovation.com/automatic-railway-gate-control-system/

[5] https://en.wikipedia.org/wiki/Arduino_Uno

[6] https://en.wikipedia.org/wiki/Relay

[7] https://en.wikipedia.org/wiki/Passive_infrared_sensor

[8] https://www.fierceelectronics.com/sensors/what-ir-sensor

[9] https://www.electronicsforu.com/technology-trends/learn-electronics/ir-led-infrared-sensor-basics

[10] https://en.wikipedia.org/wiki/Servomotor

[11] https://www.electrical4u.com/what-is-servo-motor/

[12] https://en.wikipedia.org/wiki/Buzzer

[13] https://www.futek.com/weight-sensor

[14] https://en.wikipedia.org/wiki/Pressure_sensor

[15] https://en.wikipedia.org/wiki/Veroboard

[16] https://www.youtube.com/watch?v=-P1hWdsD51E

[17] https://www.youtube.com/watch?v=gRHLo6Qud1E

[18] https://www.youtube.com/watch?v=pTGWT_fr_e8

[19] https://www.electronicshub.org/automatic-railway-gate-controller/

[20] https://www.circuitstoday.com/automatic-railway-gate-control-using-arduino-ir-sensor

[21] https://engineering.eckovation.com/automatic-railway-gate-control-system/

[22] https://www.instructables.com/How-to-Make-IR-Sensor/

[23] https://www.instructables.com/How-to-Make-an-IR-Object-Sensor-With-Arduino/

[24] https://www.youtube.com/watch?v=2yi3SsYbwHQ

[25] https://www.youtube.com/watch?v=9DbyyTATiME

[26] https://www.youtube.com/watch?v=toa5bd4pRJo

[27] https://www.youtube.com/watch?v=toa5bd4pRJo

[28] https://en.wikipedia.org/wiki/List_of_rail_accidents_in_Bangladesh

[29] https://www.dhakatribune.com/hashtag/train-accident

[30] https://relays.weebly.com/latching-relay.html

[31] https://electronics.stackexchange.com/questions/350703/can-latching-relay-make-the-switch-if-the-coil-is-powered-by-either-of-its-switc

[32] https://www.youtube.com/watch?v=pGSF4PG40X4

[33] https://www.arduino.cc/en/software

[34] https://www.youtube.com/watch?v=NJIbSI6qG6Q

[35] https://proteus.soft112.com/

[36] https://www.youtube.com/watch?v=NJIbSI6qG6Q&t=29s

[37] https://www.youtube.com/watch?v=Pd754nSIr_E

[38] https://transmitter.ieee.org/makerproject/view/63c1c

Appendix

```
#include <Servo.h>
int i=10;
int i1=11;
int buzzer=13;

Servo myservo;

// create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;
// variable to store the servo position

void setup() {

  pinMode(13,OUTPUT);
  Serial.begin(9600);
  myservo.attach(9);
// attaches the servo on pin 9 to the servo object
}

void loop() {

 int data=digitalRead(10);
  int data1=digitalRead(11);
  if(data==0 &&data1==0)
  {
  digitalWrite(13,1);
    Serial.println("ACCIDENT OCCURED");
  }

  else

  {

 digitalWrite(13,0);
if(data==0)
{
  Serial.println("data 0");

 digitalWrite(13,1);
for(i=i;i<=100;i++)
{
    myservo.write(i);
    delay(10);
}
// tell servo to go to position in variable 'pos'
                    // waits 15ms for the servo to reach the position
  }

  else{

      digitalWrite(13,0);
  }



if(data1==0)
{
   digitalWrite(13,1);
 Serial.println("data 1");



for(i=i;i>=10;i--)
{
  Serial.println("loop");
    myservo.write(i);
    delay(10);
}
 // tell servo to go to position in variable 'pos'
                       // waits 15ms for the servo to reach the position
  }



      else
      {


        digitalWrite(13,0);
      }


  }
  }
```