**‘‘Physical and Virtual Car Driving Simulator Design and Implementation’’**

# 

# Declarations

We the member of Group 3, 5th year Electrical and Computer Engineering department, stream of Control Engineering student of 2012/13E.C, hereby declare that this Thesis is our original work and that it has not been submitted in full, by any other person for a final year project in any other university/institution.

Name of Participant’s Signature Date

1) . . .

2) . . .

3) . . .

4) . . .

5) . . .

6) . . .

This Thesis has been submitted for examination with my approval as College supervisor.

Name of Advisor Signature Date .

# *Abstract*

*In today’s world almost everything that have physical means can be simulated virtually with the proper combination of hardware and software. For example, actions that can cause injury to a people are better tested virtually before going out on field and performing them directly. Some example of these actions could be like flying plane, sailing boat, shooting different heavy weapons, and also driving vehicles. This project is focused on simulating the vehicle driving by building and developing full setup of hardware and virtual car controlling program. The advantage of simulating car driving virtually is unlimited; it’ll help us avoid the injury that one may cause while training either to one’s self or to others, it can be used to train children’s how to drive without exposing them to any danger, we can also use it as game for entertainment. This project can be implanted in various industry like car training center, game zone, schools, and even for individuals.*

# Acknowledgement

First of all, we would like to thank the almighty, the creator of the universe GOD. We would like our next gratitude to go to our department for providing us with necessary tools like computers and internet connection. We also like to thank our advisor Mr. Firew D. for his great and unlimited support throughout the completion of this project. We don’t want to miss this chance to thank all of our instructors for their role on sharing the lifetime knowledge’s, all staff of ECE. Finally, we would like to acknowledge our colleague who filled our gap by giving us tools which we didn’t have and were critical to build hardware Mr. Getinet T.

# Table of Contents

[Declarations I](#_Toc62013446)

[Abstract II](#_Toc62013447)

[Acknowledgement III](#_Toc62013448)

[Table of Contents IV](#_Toc62013449)

[List of figures VI](#_Toc62013450)

[List of appendices VII](#_Toc62013451)

[List of acronyms’ VIII](#_Toc62013452)

[CHAPTER ONE 1](#_Toc62013453)

[INTRODUCTION 1](#_Toc62013454)

[1.1. Motivation 1](#_Toc62013455)

[1.2. Statement of problem 2](#_Toc62013456)

[1.3. Objective 3](#_Toc62013457)

[1.3.1. General objective 3](#_Toc62013458)

[1.3.2. Specific objective 3](#_Toc62013459)

[1.4. Scope of project 3](#_Toc62013460)

[1.5. Significances 3](#_Toc62013461)

[CHAPTER 2 5](#_Toc62013462)

[LITERATURE REVIEW 5](#_Toc62013463)

[2.1. Car driving procedure in real life 5](#_Toc62013464)

[2.2. Virtual simulator 8](#_Toc62013465)

[2.3 Serial communication 8](#_Toc62013466)

[2.4. Related works 9](#_Toc62013467)

[CHAPTER 3 12](#_Toc62013468)

[METHODOLOGY 12](#_Toc62013469)

[3.1. System design 13](#_Toc62013470)

[3.2. Material used 13](#_Toc62013471)

[3.3. User experience 19](#_Toc62013472)

[3.4. Overall system flowchart 20](#_Toc62013473)

[3.5. Used software tools 23](#_Toc62013474)

[3.6. Implementation of the system 24](#_Toc62013475)

[CHAPTER 4 26](#_Toc62013476)

[RESULT AND DISCUSSION 26](#_Toc62013477)

[4.1. Experiment and Result 26](#_Toc62013478)

[CHAPTER FIVE 29](#_Toc62013479)

[CONCLUSION AND RECOMMENDATION 29](#_Toc62013480)

[5.1. Conclusion 29](#_Toc62013481)

[5.2 Recommendation and feature work 29](#_Toc62013482)

[References 32](#_Toc62013483)

# List of figures

[Figure 1: Manual gear of vehicle [6]. 7](#_Toc62013413)

[Figure 2: Automatic gear of vehicle [7]. 8](#_Toc62013414)

[Figure 3: Serial and parallel data transfer picture for comparison [10]. 9](#_Toc62013415)

[Figure 4: block diagram of the system design. 12](#_Toc62013416)

[Figure 5: potentiometer with pin diagram. 13](#_Toc62013417)

[Figure 6: resistor diagram. 14](#_Toc62013418)

[Figure 7: LED photo. 14](#_Toc62013419)

[Figure 8: On/Off switch. 15](#_Toc62013420)

[Figure 9: Arduino uno with pin diagram. 15](#_Toc62013421)

[Figure 10: Jumper wires this is male type in both sides. 16](#_Toc62013422)

[Figure 11: rubber band. 16](#_Toc62013423)

[Figure 12: steep tube. 17](#_Toc62013424)

[Figure 13: wood board. 17](#_Toc62013425)

[Figure 14: card board. 17](#_Toc62013426)

[Figure 15: Hot glue / hot melt adhesive. 18](#_Toc62013427)

[Figure 16: Black Tape. 18](#_Toc62013428)

[*Figure 17:* Flowchart of user experience / actions without technical detail. 19](#_Toc62013429)

[*Figure 18:* Flowchart of working of the system and user experience with technical detail. 22](#_Toc62013430)

[Figure 19: Electrical design of the system. 24](#_Toc62013431)

[Figure 20: Mechanical design gear and steer of the car controller. 24](#_Toc62013432)

[Figure 21: mechanical design of controller side view and back view. 25](#_Toc62013433)

[Figure 22: When one of the side light switch is pressed. 26](#_Toc62013434)

[Figure 23: When the hazard light is turned on. 26](#_Toc62013435)

[Figure 24: when brake is pressed. 27](#_Toc62013436)

[Figure 25: car moving forward and backward. 27](#_Toc62013437)

[Figure 26: the car turning toward the direction of the steer. 28](#_Toc62013438)

[Figure 27: block diagram of improved system with physical stimulation. 30](#_Toc62013439)

[Figure 28: block diagram of system improved with behavior analyzing capability. 31](#_Toc62013440)

# List of appendices

[**Appendix 1: The program code.** I](#_Toc62012821)

[**Appendix 2: Timeline/schedule** IX](#_Toc62012822)

[**Appendix 3: Budget** X](#_Toc62012823)

# List of acronyms’

**3D :** 3 Dimensional

**D:** drive

**DVI:** Digital Visual Interface

**FPD-Link:** Flat Panel Display Link

**HDMI:** High-Definition Multimedia Interface

**HD-SDI:** high-definition serial digital interface

**HMA:** Hot melt adhesive

**IDE:** Integrated development environment

**LCD:** Liquid crystal display

**LED:** Light emitting diode

**N:** neutral

**P:** parking

**PC:** Personal computer

**POT:** potentiometer

**PS/2:** Personal System/2

**R:** reverse

**Serial ATA:** Serial AT Attachment

**Serial SCSI :** Small Computer System Interface

**USB:** Universal Serial Bus

**VGA:** Video Graphics Array

**VR:** Virtual reality

# 

# CHAPTER ONE

## INTRODUCTION

The history of transport is largely one of technological innovation. Advances in technology have allowed people to travel farther, explore more territory, and expand their influence over larger and larger areas [1]. People used different means of traveling like horse, boat, plane, train, car and so on. Nowadays we have 1.4 billion cars on the road [2]. As the demand for car transportation increased, car accidents have also increased potentially. The reason for these car accidents could be like driving car after using alcohol, using different drugs that may malfunction one’s brain, being distracted by different devices like phone, driving by exceeding maximum speed limit, and not having enough driving experience or not having good technical knowledge on driving. In our country too, according to the study conducted by Debela Deme Jima and Md Mahabubul Bari on causes of car accidents on Adama-Addis express way, they have mentioned the frequency and percent of car accidents that occur as a result of technical problem which is 271 frequency which accounts for 37.4% of accidents within 2 and half years [3]. This project is focused on minimizing the car accidents that may happen as a result of not having enough driving experience or technical knowledge to drive, by designing a way for a driver to train without first going out to the field where he/she can cause injury to one’s self or others. We provide this by designing a physical and virtual simulator for the car driving training purpose. But the scope of the project should not be only limited to training, since it can also be used for entertainment purpose and in different organizations like game zones and even for individual use.

It is usual that human being constantly tries to avoid any danger that may happen as a consequence of any action it takes. It’s not different in case of car driving too. Because, before any driver starts to ride car on road, he/she has to train well. Even though, taking training reduces the accidents that may occur in the future, the current training system by itself might be fatal if things get wrong, since it’s done in the field. That’s why we need virtual simulator. Virtual simulator refers to the process by which a certain aspect of reality is simulated in a virtual world/environment in such a way that participants can explore it and perform tests on that aspect as this it would be in the real world by using 3D objects and environments to create immersive and engaging learning experiences [4]. In this project we will design a physical set of hardware that can replace parts of car that plays main role in driving like steer, brake, fuel pedal, automatic gear, brake light, hazard light, side light, and interface all these forementioned devices communicate with the 3D car model and simulate the real-world aspect of car driving for training purpose and entertainment purpose as well.

## Motivation

The main motivation behind this project is car accidents that are really terrifying to hear. And most of these accidents are caused by driver’s fault. For example, according to a study 94% of car accidents are caused by driver [5]. Most of the time drivers that do not have sufficient experience tend to cause more accidents. And as a technology student’s it’s not right to just observe and hear all this tragedy and do nothing about it. That’s why we have decided to build a safe way to train a driver to their full talent virtually before they go out to the field where they could expose both others and themselves to possibly a fatal danger.

Beside what is mentioned above this project can also be used as an entertainment. And if we can save lives and reduce possible accidents whilst having fun on the way, there is really no reason not to get motivated to build such a system. As entertainment this simulator can be used in different schools, different game zones and even can be owned by individuals.

## Statement of problem

As we know current car driving training system is held on the field with a limited resource. Which leaves it prone to different accidents. When there is a new student who came to train at a training center, he/she will take the theoretical training on car driving first. But eventually the time will come when he/she must go out on the field to apply what he/she have learned in theory. The problem is that there are training centers which don’t have an open field for training purpose, which means they use a main road to train trainee.

We also know that machines are not perfect, which means there are things we can do to make sure the vehicle is in good shape before we go out to the field, but we can’t be 100% sure whether it is safe or that it won’t get broke. That means parts that are critical to car driving like brake, gear, steer, lights and so on might malfunction. For a well-trained driver this might be manageable or he/she might be able to reduce risks. But, for a new trainee on top of the pre-existing tension about learning new thing, the malfunctioning of these parts will add severe anxiety which will lead to different rush and poor decisions that could cause damage to property and dangers that might be fatal.

The other problems this project will be solving is the training center limited resource which have direct effect on quality of training to be delivered. This limited resource is usually insufficient number of training cars. Training centers might be prone to this problem as a result of different causes like the car price being too expensive and not having enough field space. And if they don’t have enough resource’s they will be forced to train student’s as fast as they could to be able to accept new students.

The problems listed above are related to the training center. There are consequences of the forementioned problems that are reflected on the driver’s career. Since the driver doesn’t have enough training, he/she may cause damage to property and worse may cause fatal dangers.

This project tends to solve the three issues listed above. The first issue is related to the field training, since this virtual car simulator is at stationary and in a safe room, there is no way that a person operating the machine or a trainee can cause any danger to him/her self or to the outside world, as well as property. The second issue is related to the limited resources. As we have mentioned earlier the two resources that may be limited are car and field space. The price of the actual car is far greater than the price that it cost us to build the simulator. Which means we can build multiple simulators with the same price that we use to buy one car. As of the field space we can store multiple simulators in a small room since the size is only the size of one seat of the car. Since the third issue is the consequence of the two issues, if we solve them it wouldn’t be a problem at all.

## Objective

### General objective

To design and implement the car driving simulator using the combination of hardware and 3D software car model.

### Specific objective

The following are specific objectives of this project:

* General study of 3D Simulator.
* General study of car driving procedures.
* Study of virtual simulator.
* Select hardware components for simulator.
* Develop car controller program for car model.
* Develop Arduino program for the simulator.
* Design each parts of the car.
* Hardware and software interfacing.
* System implementation.
* Testing the system.

## Scope of project

The scope of the system is unlimited that it can be used in car driving training centers, school, game zone, and even individual. In training centers, it is used to train new trainee in controlled environment. In school, it can be used to teach students how the car driving works and also can be used as source of refreshment during break time for school staff including students. The system can also be implemented in game zone as an entertainment device for customer and source of income for the game zone owner. The last place that we can use the system at is home. Yes, an individual can also buy and install the system at home both for training purpose or for entertainment purpose for family.

## Significances

As it’s been mentioned under the problem statement the current training system is dangerous both during training and after training. Because of this the main significance of this project is to reduce and/or eliminate the danger that damage property and may cause fatal harm to humans and other animals. Even if reducing or eliminating danger is the main significance of the project it can also be used for entertainment purpose which can be enjoyed by everyone of different age without any restriction. These are the significance of the project in general in the next paragraphs we will see the detail.

The significance of this project in terms of reducing danger can be seen in two different ways. The first is the danger during training and the second is the danger after training which means when the driver is on the career. If training centers have access to the simulator machines that is getting built in this project then they will significantly solve the two issues mentioned like insufficient training and resource limitation.

The other significance of this project is the positive consequence of this project solving the issues during the training then that will significantly reduce or eliminate the problem that the driver may cause during his/her career because of insufficient training.

Generally, if we can reduce or eliminate the human injury and property damage, indirectly, we are also saving a resource. Which means there will not be a wasted property or money that may impact individual’s or even worse country’s economy.

# CHAPTER 2

# LITERATURE REVIEW

As it has been mentioned on this paper earlier the aim of this project is to design the physical and virtual car driving simulator using graphical car on computer and set of hardware than can mimic partial function of the car driving. First this paper discusses the car driving process in real life and how this virtual simulator works in this chapter before going into the details. But it is clear that there are a lot of researchers who have worked on this interesting area of study to simulate car driving using set of software’s and hardware’s regardless of the limitations. So, before we rush into design of the system, we have realized that it’s necessary to review different literature in the area of this project. Because, it’ll help us understand the working of the system more but also after we have reviewed those research’s we have seen a lot of gaps and limitation. Which in turn help us avoid to make same mistake. The reviewed literature and their limitation have been discussed roughly on the next section below.

### 2.1. Car driving procedure in real life

Even though the aim of this project is not to educate on how to drive car, it’s very important to discuss the procedure of driving car. Because the system to be built is going to mimic the car driving experience as much as possible. There are two types of vehicle based on type of their gear shift mechanism; manual and automatic. This project is going to mimic vehicles with automatic gear. But both types will be discussed in detail as following.

#### 2.1.1. Driving vehicles with manual gear

The very first thing we should do when driving car for the first time is to examine the mirrors if they are in the correct orientation, if they are not adjusting them to optimum orientation. Adjust your side and rear-view mirrors so you can see what's behind you and to the side. Make sure you are in a secluded area and that you have enough space in front of you so you can move at different speeds without running into something. After mirror gets adjusted get to know the position of the gear. With the engine off, move the stick around. All gear arrangements are in an "H" shape, but the location of reverse varies. On some stick, reverse is to the right and down. On other stick shifts reverse is to the left and up. Then orient your leg to the position of pedals. Most of the car with manual gear have 3 pedals namely Clutch, Brake and Fuel. With each used for three different purpose [6].

Now start the car but before starting the car gear should be put on neutral. To start the car, push the clutch in all the way in, hold it down, and turn the key. Make sure the car is in neutral before releasing the clutch. The neutral slot is the empty space between the gears, represented by the bar in the middle of the "H" to see if car is in neutral, wiggle the stick gently to see if gear can move left and right. If it can, it is in neutral. Now the engine is running, doing nothing and going nowhere, which is just fine for this first step [6].

Shift the gear to first. Now push the clutch in all the way, and put the car into first gear using the gear stick. We usually find first gear by pushing the stick left and then away from driver. Keep the clutch in or the car will be stalled! But it is nothing to worry about: The car will be stalled trying to shift into gear, but like any other activity practice makes perfect. Release the Emergency Brake. Now that the car is running, make sure your emergency brake (handbrake) is off. Since you are on flat ground, the car won't roll anywhere while you are figuring out what to do. Release the Clutch and give the engine some gas once you are in first gear, slowly release the clutch until you feel the car just start moving. Begin to slowly add some gas by pushing down the gas pedal a little bit. Give it enough gas to move the tachometer needle to like "1" or "2," that is, 1000 to 2000 rpm. If you give too little gas, the car will stall. If you give too much, you begin to burn the clutch, which is no big deal for a beginner, but over time can wear out the clutch [6].

Release the Clutch after gears engage Now that you are giving the car a little bit of gas, don't release the clutch fully just yet. Release it when you can feel the gears engage. This will feel like a vibration you can sense through the sole of your foot. Once you get to that point, you can release the clutch, and drive around without using the clutch, just as you would in an automatic transmission car.

Shifting into higher gears. Let's say you are in first gear going into second. Follow these procedures:

1. Put the clutch in, and then move the stick to second. On most cars, to go from first to second, you pull the stick towards you, that is, down to the bottom leg of the "H."
2. Slowly release the clutch to the biting point, and then add a little bit of gas.
3. Let the clutch all the way out and continue driving around.

Going from second to third and then fourth gear is the same process, and even easier. In order to make a totally smooth shift, you have to balance the gas and the clutch very well. Don't worry, this comes with practice. But the main question here is how to know when to shift gear. The answer to this is as following, gear should usually be shifted at 2500 to 3000 rpms. The speed to shift gears varies with the car you are driving, so look at the instruction manual to determine the speeds. The speed for each gear isn't as important for upshifting as it is for downshifting. Eventually you will be able to tell when to shift by the way the engine sounds and feels.

All the above paragraphs under section how to drive car with manual gear explains in detail how one can drive a car with manual gear. But it’s also important to summarize the content of this section to make it memorable for anyone who wants to learn how to drive.

Summary of driving car with manual gear: -

1. Check/Adjust the mirror both rear and side.
2. Know where the gears are.
3. Know the pedals and align to our foot.
4. Start the car in neutral.
5. Shift in to the first gear.
6. Release the emergency/hand brake.
7. Release the Clutch and give the Engine some gas.
8. Release the clutch fully slowly after the gear engages while pressing the gas pedal.
9. Shifting in to higher gears.



Figure 1: Manual gear of vehicle [6].

It’s not enough to learn only how to drive, how to stop and how to drive car in reverse is also a must to know. To stop the car, you don't need to press in the clutch every time you press the brake to slow down. But when coming to a full stop at a stop sign or traffic signal, press in on the clutch when you are almost to a full stop and shift into neutral, or you will stall the car. To drive car in reverse from a full stop, press the clutch in all the way and move the gear shift into reverse.it is also important to look behind car and use mirrors before giving the engine a little gas. Ease off on the clutch when you feel the reverse gear engage. Continue to look behind you and use your mirrors while moving in reverse.

#### 2.1.2. Driving car with automatic gear

Driving cars with automatic gear procedure can be summarized in to the following points: -

1. Find a flat, paved place with no one around so you can practice.
2. Adjust your mirrors so you can see behind and to the side of your car.
3. Familiarize yourself with the pedals: brake on the left, gas on the right.
4. With the car in park and the emergency brake on, turn the key to start the engine.
5. With your foot on the brake, release the emergency brake.
6. With your foot still on the brake, put the car in drive.
7. Lift your foot off the brake and slowly press down on the gas pedal.
8. Practice gently pressing the brake. At first you will press too hard and the car will jerk to a stop. With practice, stopping and starting will be smoother.

The procedure to drive automatic gear car is the almost same as that of manual geared car driving procedure. But this section discusses the main difference in their procedure leaving out steps that are the same as the first one. Steps that are different are like the missing clutch, and the “H” shape gear is no more. We don’t need the clutch since we don’t have the manual gear. Automatic geared vehicle has P (parking), R (reverse), N(neutral), D(drive) as shown in figure 2 below for different car driving situations.

The gear is set to P when parking, R to move reverse/backward direction, N to stop the car, and D mode while driving. As we can see there is difference between manual and automatic since the automatic one handles gear change internally the manual one uses the personnel to shifts its gear from one to another.



Figure 2: Automatic gear of vehicle [7].

### 2.2. Virtual simulator

Virtual Simulation is the use of 3D objects and environments to create immersive and engaging learning experiences. A simulation that takes place in a virtual environment. Refers to the process by which a certain aspect of reality is simulated in a virtual world/environment in such a way that participants can explore it and perform tests on that aspect as this it would be in the real world [9].

## 2.3 Serial communication

In telecommunication and data transmission, serial communication is the process of sending data one bit at a time, sequentially, over a communication channel or computer bus. This is in contrast to parallel communication, where several bits are sent as a whole, on a link with several parallel channels [10].

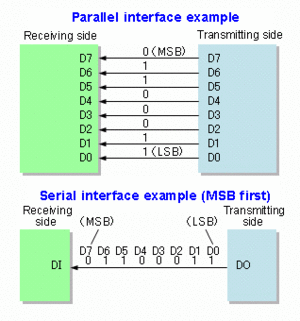
Many serial communication systems were originally designed to transfer data over relatively large distances through some sort of data cable. Practically all long-distance communication transmits data one bit at a time as shown in figure 3, rather than in parallel, because it reduces the cost of the cable. The cables that carry this data (other than the serial cable) and the computer ports they plug into are usually referred to with a more specific name, to reduce confusion. Keyboard and mouse cables and ports are almost invariably serial such as PS/2 port, Apple Desktop Bus and USB. The cables that carry digital video are serial such as coax cable plugged into a HD-SDI port, a webcam plugged into a USB port or Firewire port, Ethernet cable connecting an IP camera to a Power over Ethernet port, FPD-Link, etc. Other such cables and ports, transmitting data one bit at a time, include Serial ATA, Serial SCSI, Ethernet cable plugged into Ethernet ports, the Display Data Channel using previously reserved pins of the VGA connector or the DVI port or the HDMI port.

Figure 3: Serial and parallel data transfer picture for comparison [10].

### 2.4. Related works

Since we are not the first peoples to conduct a project in 3D vehicle driving simulator, we have tried to look at the related project in order to understand the working principle of the system, the limitation, the strength and every information we could get. Regarding the 3D Simulator the following paper has been summarized as following:

Mrs. Smita Rukhande, Mrs. Rupali Deshmukh (2018), in their paper they implemented system Virtual Drive that will provide the learner a virtual realistic environment to learn driving without being in any danger. Virtual car simulator will avoid the danger of being on road directly and providing training virtually which is much safer. The system consists of functions such as dynamic feedback, hardware interface for user. The simulator provides a realistic, highly immersive driving experience. Even special driving scenarios, such as tight corners, braking, emergency cases and accident situations are simulated realistically and sufficiently dynamically [6].

Petr Bouchner, et al (2005), They mentioned in their paper that they set up three different devices. The very first arrangement of their simulation device used a common PC steering wheel with two pedals with a sequential gear shifter (or automatic shifting was applied). The second device uses a special three-pedal system (with a possibility of involvement of the clutch if required) and an H-pattern gear shifter. Although this setup has been replaced by a real part of the car cockpit, they came back to it when incorporating Head Mounted Display. The third one is named “compact" simulator and it is the closest to reality (concerning ergonomics) because it uses a complete real car body. The tested person sits in a real cockpit and the virtual scenery is projected on the screen wall in front of the car hood. The results of measurements using such a device should not be loaded by an error caused by the difference between the simulator and a real car cockpit [7].

Dipl.-Eng. Andreas Thanheiser, et al (2012), as they mentioned in their paper for the simulation environment Dymola from Dassault Systems has been used because of the object-oriented modeling language Modelica. The advantage is among others the possibility to easily realize systems with different physical domains coupled in one system. This is especially the case in electric vehicle systems here at least electric and mechanic systems are coupled [8].

H. S. Kang, et al (2004), mentioned in their paper that the driving simulator for this research is developed using PC-based workstations that are capable of producing high fidelity graphics at reasonable cost. The VR-based simulator gives a driver on board the impression that he drives an actual vehicle by predicting vehicle motion caused by the driver input and feeding back the corresponding visual, motion, audio and proprioceptive cues to the driver. This research is intended to provide a test bed for simulating driving related task using virtual reality technology [9].

Ambar Yadav and Arti Singh (2014), have published a paper on virtual driving and as they have mentioned on their paper virtual driving simulator is a device that allows user to feel a life-like experience of driving an actual vehicle within virtual reality. It is effectively used for studying the interaction of a driver and vehicle and for developing new vehicle systems, human factor study, and vehicle safety research by enabling the reproduction of the actual driving environments in a safe and tightly controlled environment. Mostly vehicle simulators consist of physical mockups as examples steering wheel, gearshift and pedals. These are essential in trying to simulate real conditions, then it’s become as a drawback for system becomes more expensive, more huge (non-mobile), and then limited to reflect changes on the vehicle type, dimensions, or interior design [10].

Johan Janson Oltsam (2006), This paper titled “Simulation of vehicles in a driving simulator using microscopic traffic simulation” describes a model that generates and simulates surrounding vehicles for a driving simulator. The proposed model generates a traffic stream, corresponding to a given target flow and simulates realistic interactions between vehicles. The model is built on established techniques for time-driven microscopic simulation of traffic and uses an approach of only simulating the closest neighborhood of the driving simulator vehicle. In our model this closest neighborhood is divided into one inner region and two outer regions. Vehicles in the inner region are simulated according to advanced sub-models for driving behavior while vehicles in the outer regions are updated according to a less time-consuming model. The presented work includes a new framework for generation and simulation of vehicles within a moving area. It also includes the development of an enhanced model for overtaking’s and a simple mesoscopic traffic model [11].

Mikael Lebram, et al (2006), This paper presents the design and architecture of a mid-range driving simulator developed at the University of Skövde. The aim is to use the simulator as a platform for studies of serious games. The usage of video game technology and software has been a central design principle. The core of the simulator is a complete car surrounded by seven screens. Each screen is handled by a standard PC, typically used for computer games, and the projection on the screens is handled by budget LCD-projectors. The use of consumer electronics, standard game technology and limited motion feedback makes this simulator relatively inexpensive. In addition, the architecture is scalable and allows for using commercial video games in the simulator. Observations from a set of experiments conducted in the simulator are presented in this paper. In these experiments driving school students were instructed to freely explore a driving game specifically designed for the simulator platform. The result shows that the level of realism is sufficient and that the entertainment value was considered to be high. This opens the possibilities to employ and use driving simulators for a wider set of applications. Our current research focuses on its use with serious games for traffic education [12].

Even though we have reviewed the above papers, this paper hasn’t copied any circuit design or required code from any of the above-mentioned researches. Everything that is going to be developed; all the design of the architecture, code, and algorithm is our own except those which are patented and licensed to the public, and platforms which are core to the function of the system but their construction is out of scope of this project like car graphics. Being said that this paper also mentions all references since those researches are public and they deserve the credit for the job they have done to make a work easier for car simulator design.

# CHAPTER 3

# METHODOLOGY

In this chapter will try to discuss the approach followed to complete the project by discussing about the methods used to gather data, to assemble the circuit, to write the car controller program, and interfacing of the designed hardware with the computer to be used for simulation from start to end step by step. In order to do so we have used different diagrams as well as tables and figures to clearly show the procedures to make the whole process as understandable as possible. This chapter also tries to discuss all required materials as brief as possible including their pin configuration and other necessary data by referring to their manual or documentation provided by either the vendor of the tools or other third parties.

This section of paper discusses the approaches as well as materials needed to work on this project. Divided into sections the system design, materials used, user experience, and last the flowchart of the overall system. But as introduction it’s better to mention the flow of the overall system construction here.

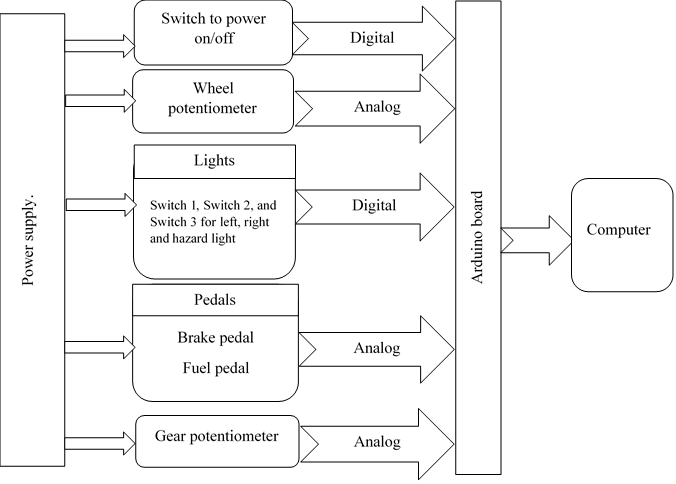


Figure 4: block diagram of the system design.

First each hardware parts of the simulator like steer, light switches, gearshift, pedals, and holders will be constructed separately and then assemble all of them mechanically. After all components got assembled together, we will interface it with Arduino. Then connect the Arduino to computer using USB. Then the Arduino program that communicates serially with the car controller program will be developed. The program that receives the data to control the virtual car by listening to the port the Arduino is communicating to will be developed.

## 3.1. System design

This section will discuss the overall system design which covers the components used, block diagram of the architecture of the system, the design of the simulating software and how they communicate to the designed hardware. The components are Rotary potentiometer, Resistor, LED, Electrical switch, Arduino, Jumper wires, Mechanical spring, Steel tube, Wood board, Card board, Hot glue, and Tape. Let’s see the role of each components one by one with their respective functions. In figure 4 of block diagram, we will visualize the system design.

## 3.2. Material used

The following are list of materials used for the construction of the system with their detail.

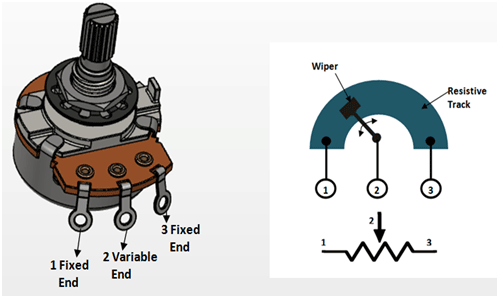
**Rotary potentiometer: -** is the most common type vary their resistive value as a result of an angular movement. Rotating a knob or dial attached to the shaft causes the internal wiper to sweep around a curved resistive element. The most common use of a rotary potentiometer is the volume-control pot. But for this project we use it to adjust the fuel, brake, and steer by varying the analog input to the Arduino. The figure 5 shows the picture of potentiometer alongside its electrical property

Figure 5: potentiometer with pin diagram.

**Resistor:** - is a passive two-terminal electrical component that implements electrical resistance as a circuit element. In electronic circuits, resistors are used to reduce current flow, adjust signal levels, to divide voltages, bias active elements, and terminate transmission lines, among other uses. Resistor is used to limit current flow to LED’s in this project.



Figure 6: resistor diagram.

**LED: -** a light-emitting diode (LED) is a semiconductor device that emits light when an electric current is passed through it. Light is produced when the particles that carry the current (known as electrons and holes) combine together within the semiconductor material. In this project they are used to mimic the role of car lights like brake light, hazard light, and side lights.

Figure 7: LED photo.

**Electrical switch: -** is an electrical component which can make or break electrical circuit automatically or manually. Switch is mainly working with ON (open) and OFF (closed) mechanism. In our case we used switch to control LED’s. the type of switch used is SPST (single pole, single throw).



Figure 8: On/Off switch.

**Arduino Uno: -** is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output pins that may be interfaced to various expansion boards and other circuits. Arduino have the ability to communicate serially with external hardware. And in this project, it will be used to take the analog input form the physical car controller and send digital data using serial communication to be used by the computer to command the car model.

Figure 9: Arduino uno with pin diagram.

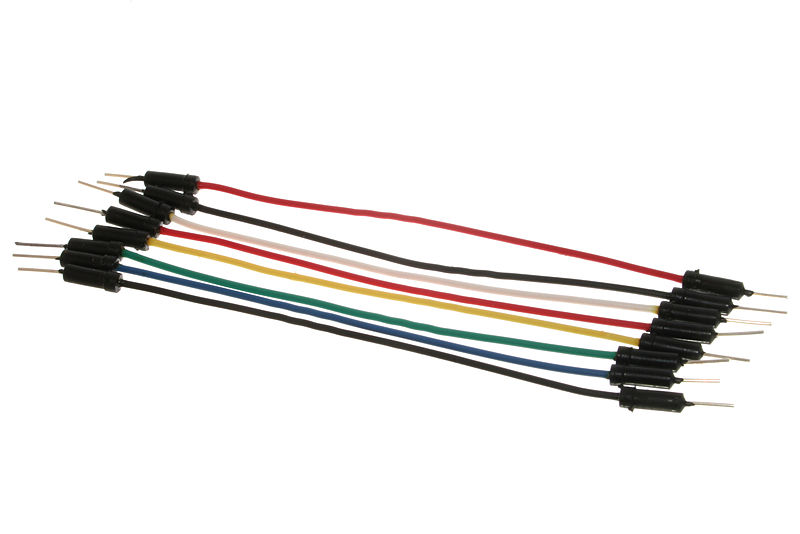
**Jumper wires: -** are wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed. We will also use cable wires to connect component’s where the jumper wire couldn’t cover.

Figure 10: Jumper wires this is male type in both sides.

**Rubber band: -** is an elastic object that stores mechanical energy. They are typically made of rubber. The use of this rubber band in this project is to pull back the two pedals (brake and fuel) after the force exerted on them is removed.



Figure 11: rubber band.

**Steel tube: -** is a versatile material that is available in round, square, and rectangular shapes. It is typically made from a variety of different alloys, like aluminum and titanium, that provide its strength and other notable properties. For this project we will be using the round shaped tube to hold the steer to the floor of the hardware.



Figure 12: steep tube.

**Wood board: -** is a flat, thin, rectangular piece of wood which is used for a particular purpose. In this project we used wood board to attach components like steel tube which holds the steer, pedal holder, spring, and gearshift holder.



Figure 13: wood board.

**Card board: -** is a generic term for heavy-duty paper-based products having greater thickness and superior durability or other specific mechanical attributes to paper; such as foldability, rigidity and impact resistance. For this project it will serve as a cover to make the card board look less boring by giving it some color.



Figure 14: card board.

**Hot glue: -** also known as Hot melt adhesive (HMA), is a form of thermoplastic adhesive that is commonly sold as solid cylindrical sticks of various diameters designed to be applied using a hot glue gun. The gun uses a continuous-duty heating element to melt the plastic glue, which the user pushes through the gun either with a mechanical trigger mechanism on the gun, or with direct finger pressure. We used it to glue the component’s together.



Figure 15: Hot glue / hot melt adhesive.

**Tape: -** according to oxford dictionary is a narrow strip of material, typically used to hold or fasten something. In this project we used tape to cover the steer to make it have the black color, and also strengthen the bond of the card board.

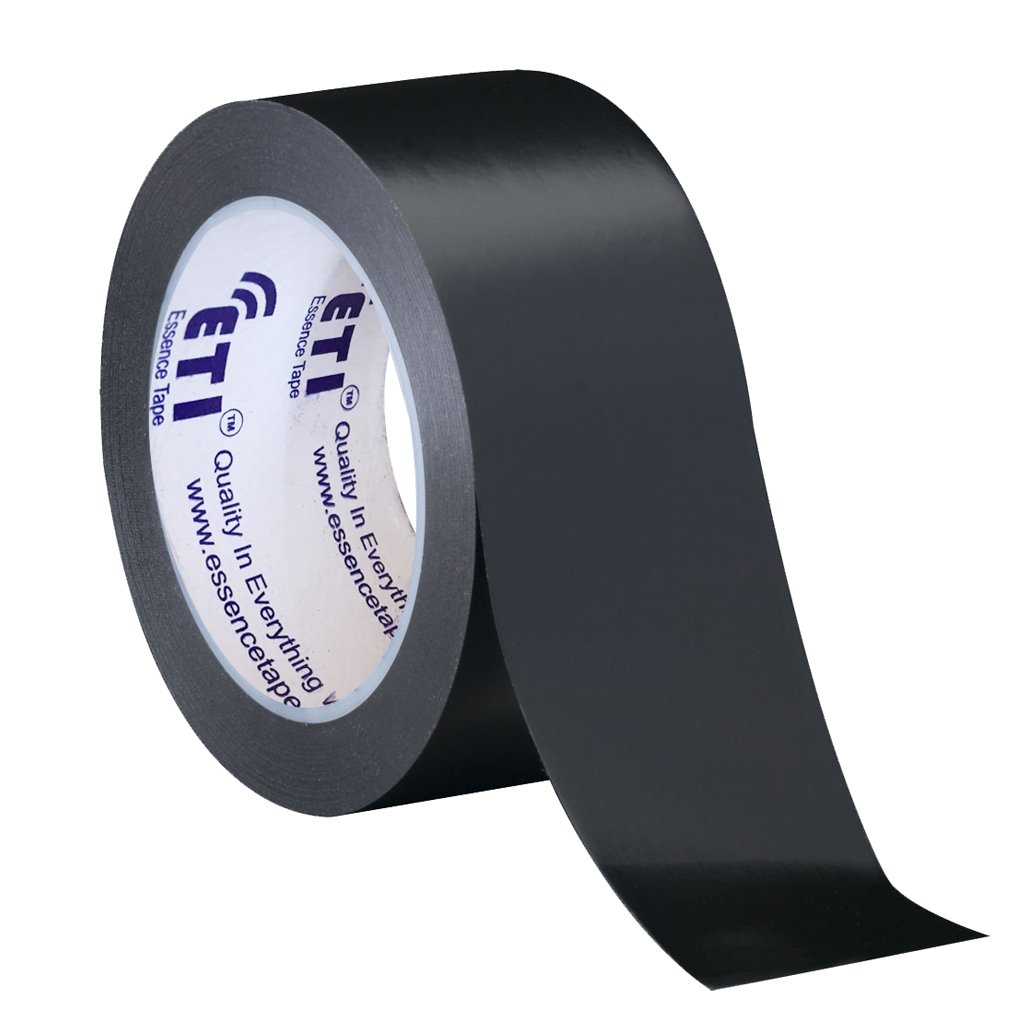


Figure 16: Black Tape.

## 3.3. User experience

This section of methodology is to clarify the overall steps that the driver goes through. It lists all task the user does and experiences in form of flow chart.

Adjust gear

Reverse /Forward

Adjust light to the direction

Press fuel pedal

Use steer for direction

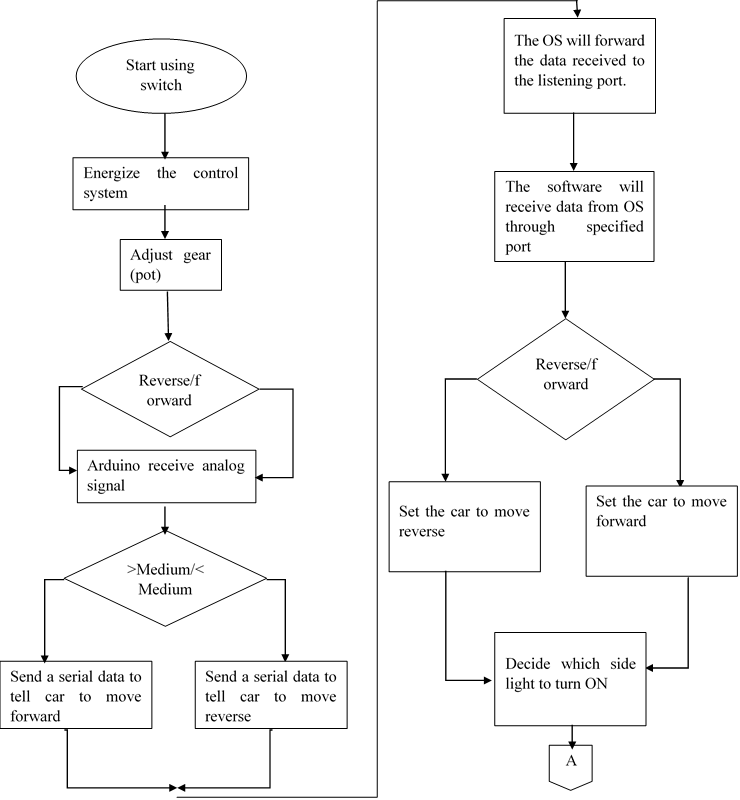
Stop/continue

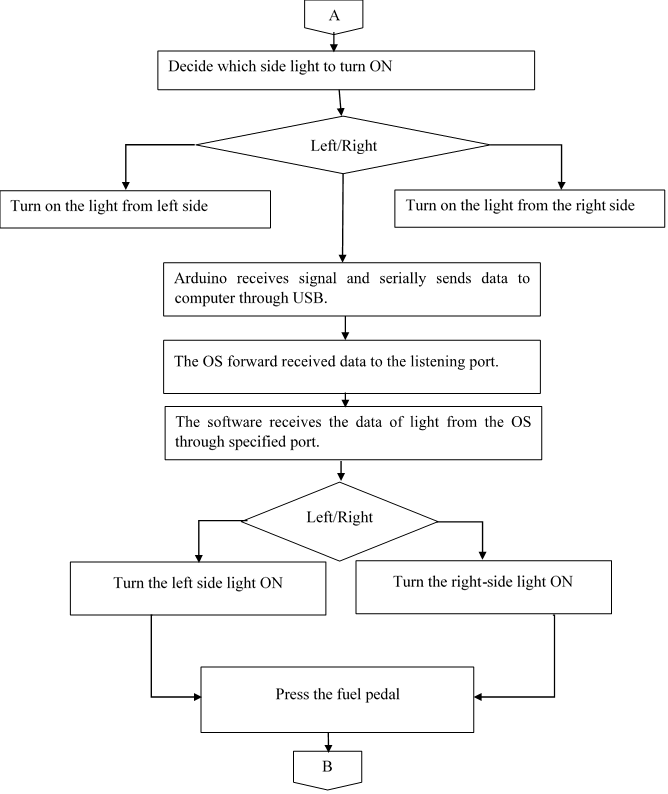
Brake

*Figure 17:* Flowchart of user experience / actions without technical detail.

## 3.4. Overall system flowchart

Overall working of the system will be written in detail right after the end of this flowchart.





Press the fuel pedal

Turn the physical side lights OFF

Arduino will sense the analog input from fuel pedal & send it serially to OS using USB.

OS will forward the data to the port listening.

The virtual software will receive the signal to keep continuing.

Stop/continue

Press the brake pedal

Use steer to adjust direction.

B

C

Press the brake pedal

Arduino will receive analog data from brake pedal and send it to OS.

The OS will forward data to the listening port.

The software will receive brake information through specific port and slows down the car and turn ON the brake light.

Driver switch OFF the whole device using switch used to turn ON.

Physical brake light will turn ON.

C

*Figure 18:* Flowchart of working of the system and user experience with technical detail.

To explain the whole system development methodology, we can divide it into 3 parts the development of the car controller program, the development of Arduino program and the development of the actual hardware to be used with the software system.

To develop the car controller program, we had to first have the car to be controlled. To get that we used a Unity game editor which enabled us access different car models and vehicle physics which are required to control it. In unity editor there is a concept known as asset which is a prepackaged set of tools, models and scripts that are great foundation to develop any custom game apps upon them. In this project a Race Car Kit asset is utilized which is a great fit for the need of the project, with good physics, lighting, sound, and control. The asset comes with different scene from that we used the vehicle scene. Then there is a script called “vehiclecontrol.cs” which is used to control the car and has the required physics library inclusion. This script has been utilized to use a “System.IO.Ports” library of C# which is used to accept the controlling signal that comes from Arduino. But the program is unable to do so by itself it requires a library/package called “Ardity” which is named from Arduino + Unity which enabled us to connect the car object to the serially coming data.

The development of the Arduino program used the C language on Arduino official IDE. First inputs and outputs were identified. In this project, 3 POT and 7 Switches were Inputs and 4 LED and 1 Serial port were Outputs. The analog data of POT is received and merged together and sent serially to Unity, then the unity car controller program will split the incoming data and put it on some useful purpose. 7 Switches are used to control car side lights, car hazard light, and gear shifting system. The switches attached to the Arduino by a pull up resistor switching mechanism. And the pots are attached to the Arduino’s analog input.

The building of the controller hardware was done by utilizing the components listed above under section of Materials Used. Getting components were really challenging but with some compromise to the design it is finally done. First the Wood Board is attached to the shaft made of steel tube that holds the Steer. And then brake and pedal also got attached to the Wood board with 2 potentiometers attached to them. The Steer is connected to one potentiometer which is attached to the head of the shaft. All components like switches and potentiometers are attached to the Arduino using the jumper wires. After all the above parts have been developed are attached together using tape and hot glue fix.

## 3.5. Used software tools

Actually, now a day everything is computable virtually so we are going to use a software called Proteus to develop the circuit and simulate first and use a game developing engine called Unity for the development of game simulation that will later be integrated with the hardware designed. We are also using an Arduino programmer or Arduino IDE to write code for the system. The language we are using will be C for (Arduino) and C# (for Unity car controller) programming language. Beside Proteus and Arduino IDE, we have also used a software like MS Excel to generate a time line in form of Gantt chart, MS word to write document, and to draw different flowchart’s and mind maps regarding our project.

## 3.6. Implementation of the system

This system has both mechanical and electrical design which is illustrated below as following:



Figure 19: Electrical design of the system.

Mechanical design is as following by different view like side, back and parts:

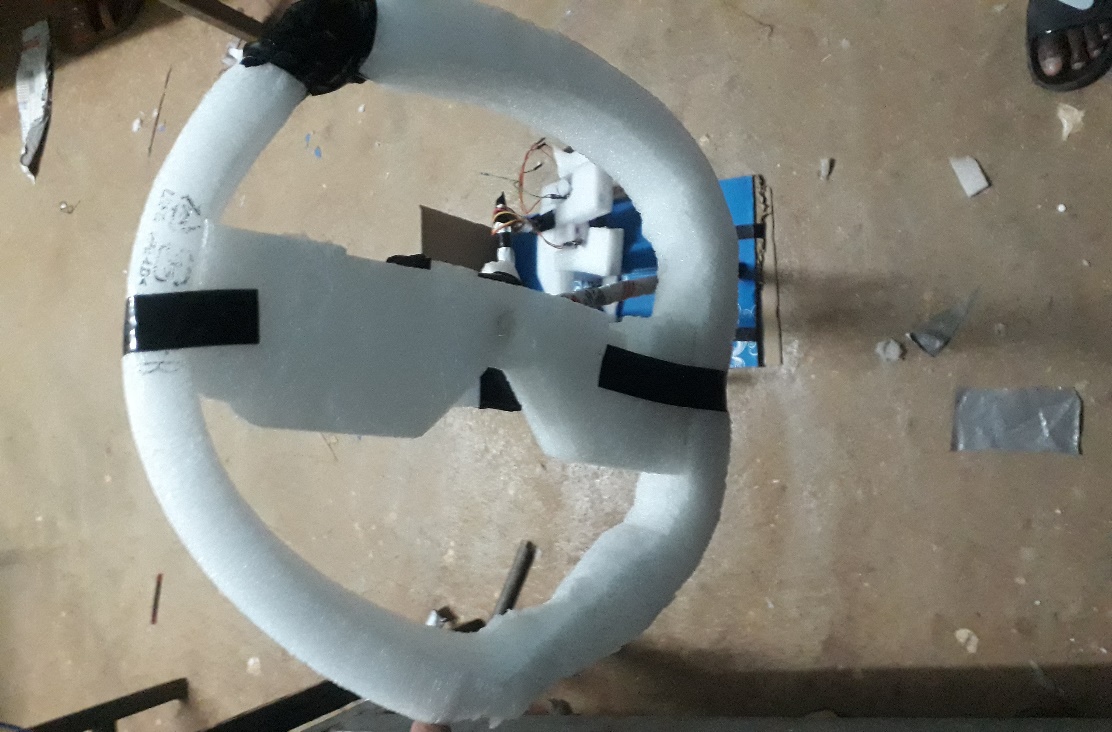


Figure 20: Mechanical design gear and steer of the car controller.

The following two pictures will be the implemented system holder that has Steer, and the two pedals brake and fuel.



Figure 21: mechanical design of controller side view and back view.

These chapter we have clearly stated almost everything’s that are related overall design which is the material used, the configuration of each materials, tools used, the flow of our program, structure of our circuit, and so on. Result and recommendation will be stated in the two next chapter.

# CHAPTER 4

# RESULT AND DISCUSSION

Now we have already designed our system and now it’s time to test our circuit for connection errors, as well as our code for bugs. Then, we will see what the result of our project will look like on different input and different conditions. For each of command and conditions we have taken the screen shoot of the output by snipping tool provided by Windows.

## 4.1. Experiment and Result

This system’s output can be generalized by grouping it to four part namely Forward movement of the vehicle, Reverse/Backward movement of the vehicle, Direction control or steering the vehicle, and using the side lights and brake lights. Let’s see them one by one by producing the required signal to generate each output.

When one of the side light switch is pressed one of the yellow LED will blink as following:

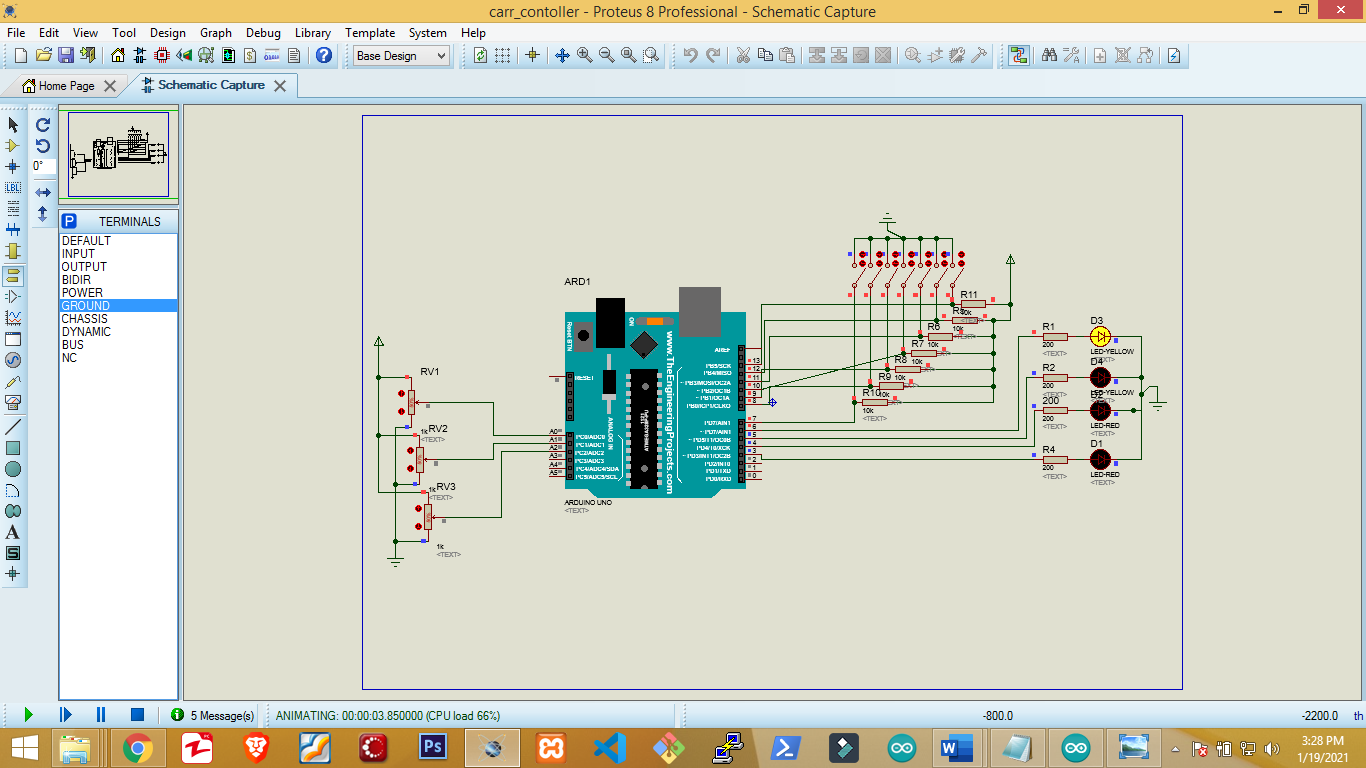


Figure 22: When one of the side light switch is pressed.

When the hazard light is turned on the following result will come which is blinking of the two yellow side light.

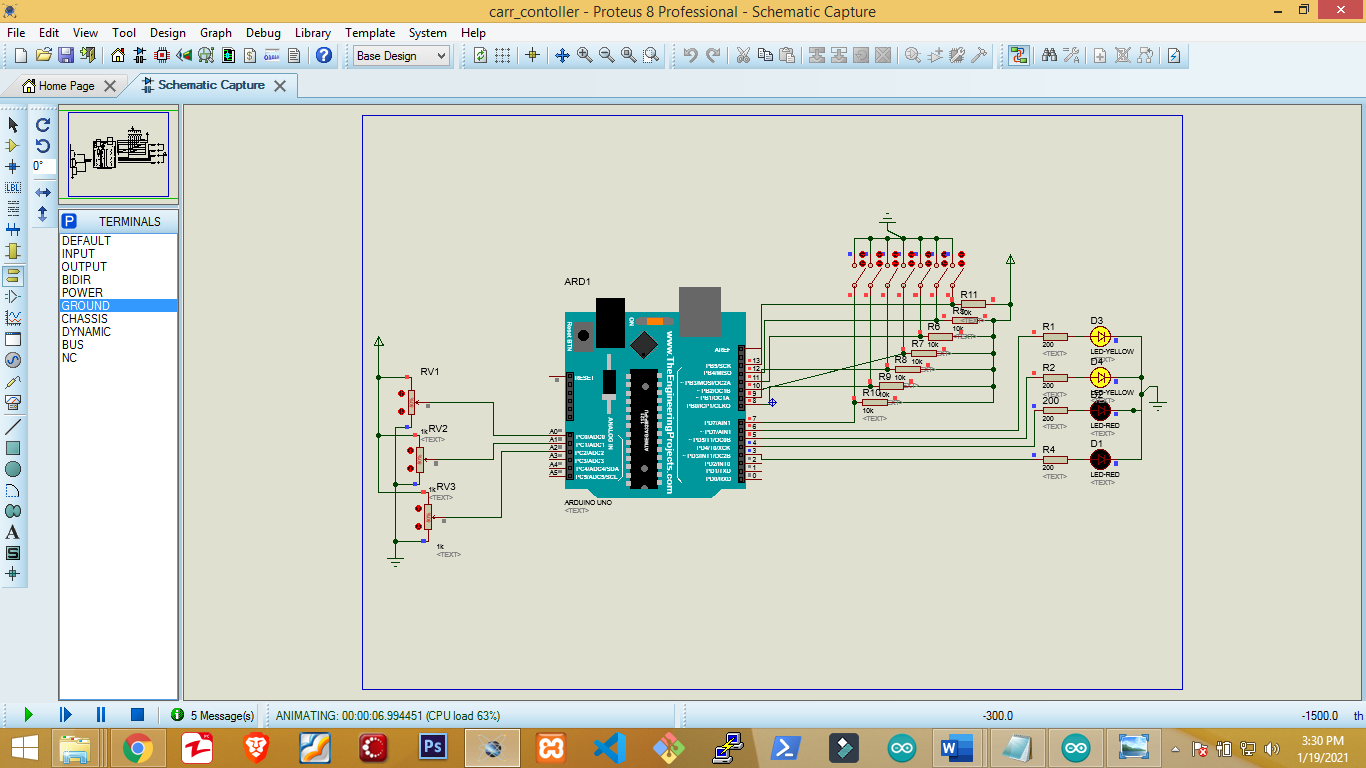


Figure 23: When the hazard light is turned on.

The other light condition is when we hit the brake the two red light will lit up as following:

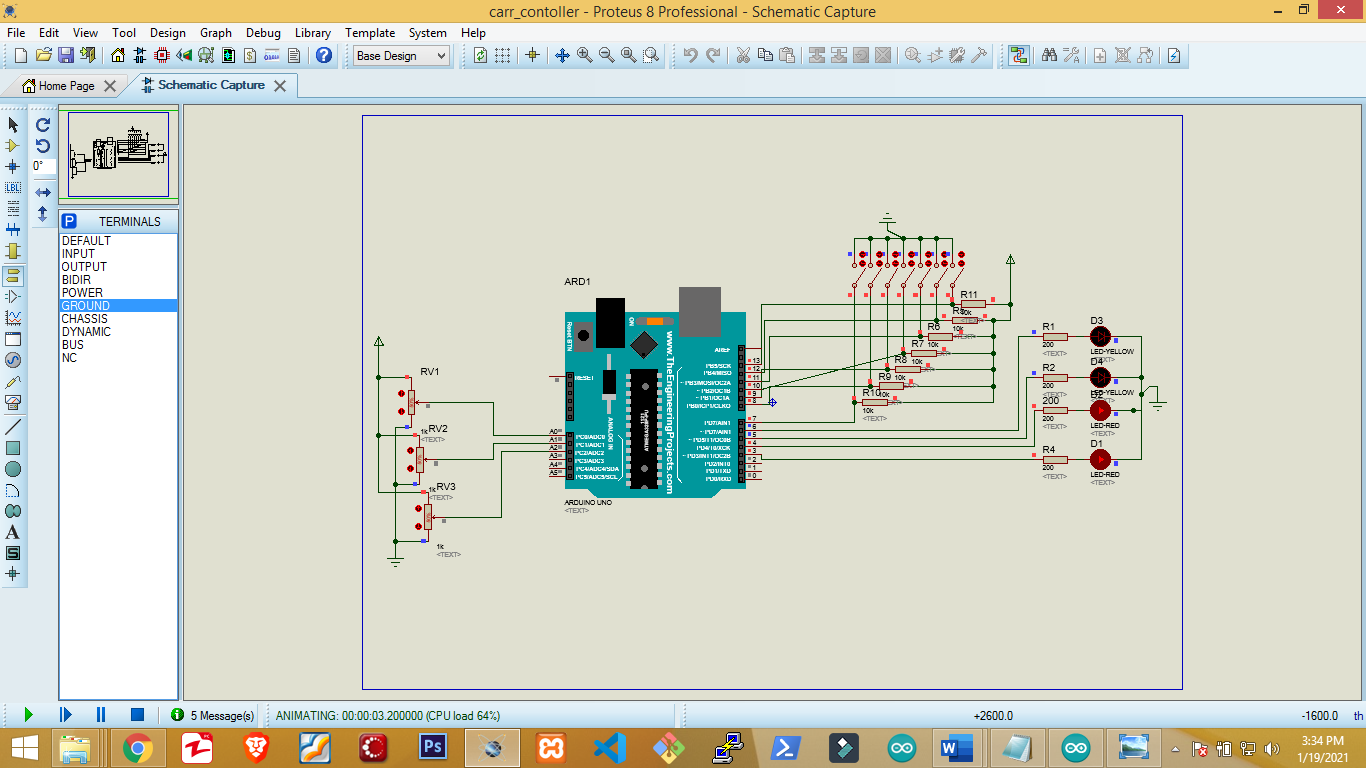


Figure 24: when brake is pressed.

Now the light part output is covered let’s see the output of the car model that is being controlled in different direction and motion.

Since the light on car is not visible on printed document for both forward and reverse its good idea to use just one picture as demonstration.

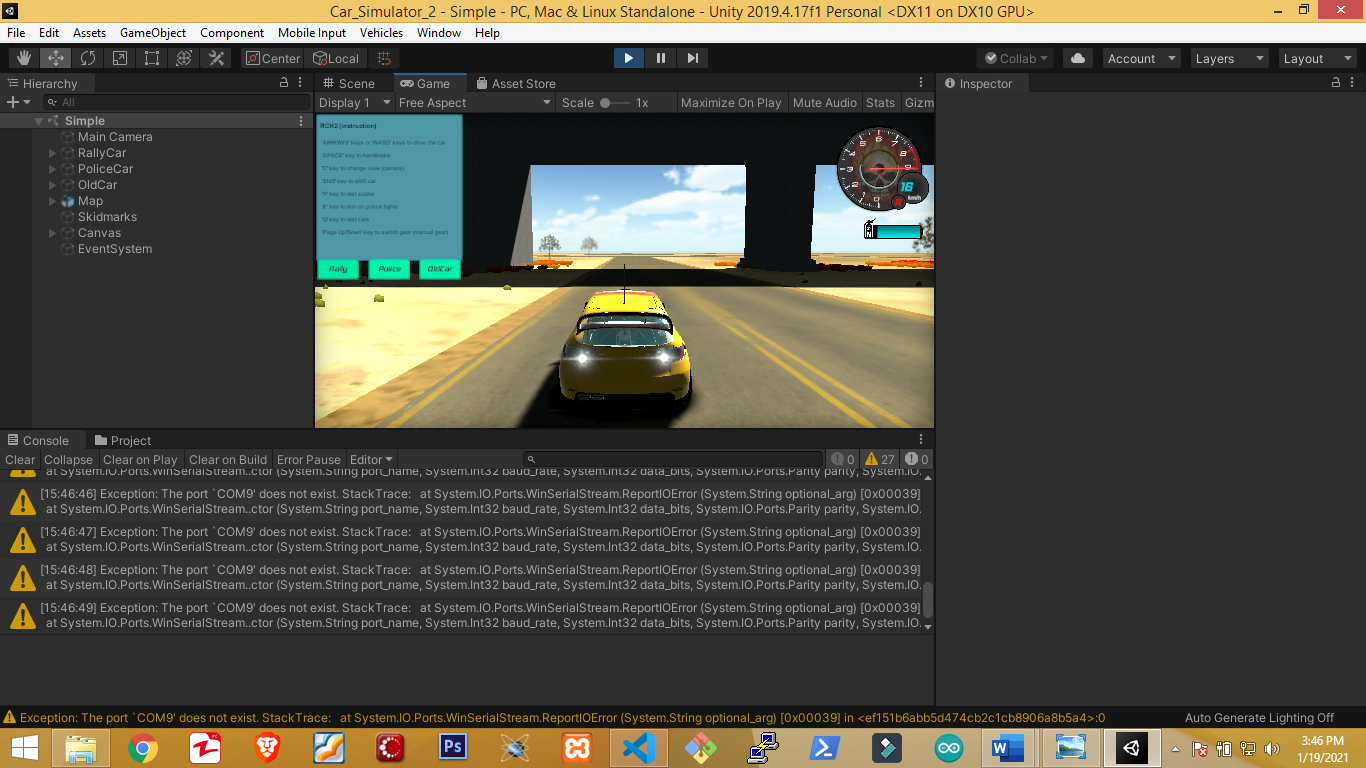


Figure 25: car moving forward and backward.

When steering the car position will look like the following based on the selected direction:

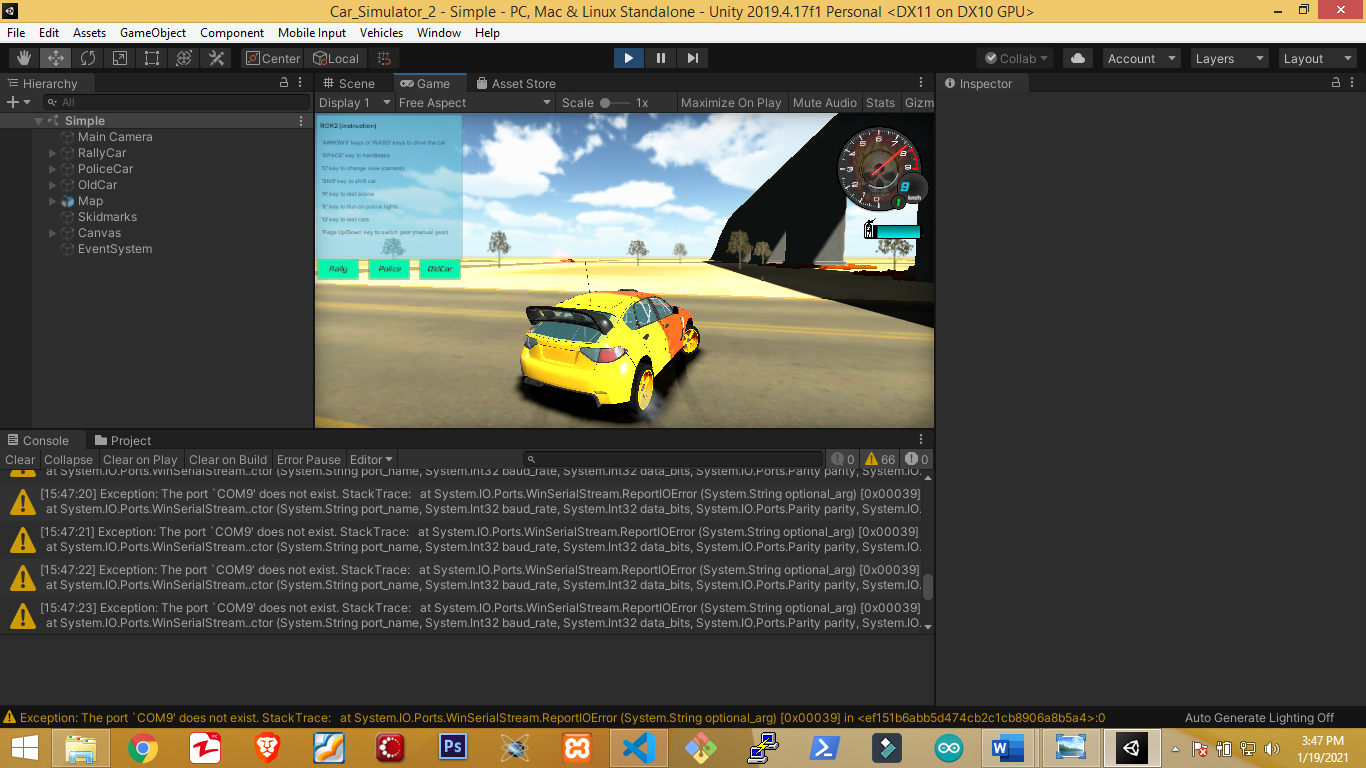


Figure 26: the car turning toward the direction of the steer.

So far, we have seen what we can do with the system and what we can’t do by using screenshot of all possible outputs. And we can conclude that the system has worked the way we wanted. Except for the limitations stated in chapter 1 and other factors like incapability of the user. Actually, there is no system that is fully perfect on its first development. Ours also needs an update of both hardware and software program which will be stated in the next chapter as a recommendation for further study and tool development in this area.

# CHAPTER FIVE

# CONCLUSION AND RECOMMENDATION

It’s obvious that there is no system that is fully perfect on its first development. Which means every system in our world needs to be improved starting from the day of its development. The update to the device is not only because of the limitation and error but also to improve the user experience toward the system. And also, to meet the need of the situation at a time of update. This chapter will address our findings conclusion and future recommendation.

## 5.1. Conclusion

This project is designed to solve problems related to car driving which can be either during training or during after training as a result of inadequate training. It solves this issue by designing a physical and virtual car driving simulator from different electrical and mechanical components as well as set of computer programs.

There are a lot of research’s done around this area which solves different problems some were developed for entertainment, some were developed to aid the study of medical treatment’s through virtual reality technology, and some were developed to simulate the operation of advanced machines and different equipment’s before testing them on field.

This project is designed using the electrical components like Potentiometer, switch, resistor, jumper wires, Arduino uno, and LED’s. Part’s used for mechanical design of this project are steel tube, rubber band, card board, and wood board. Set of software’s used during development of this project are Arduino IDE, Unity game editor for Arduino program and vehicle simulation respectively. To hold each hardware pieces together hot glue, and tape were used.

The result of this project, as it has been mentioned earlier the inputs to the Arduino are the three potentiometer and the seven switches each for different purpose. The analog input of the Arduino we will get the value between 0 and 1023. But the steer and acceleration of the car control program uses the value between -1 and 1 to control the car both vertical and horizontal direction. So, this 0-1023 needs to be mapped to the value between -1 and 1 but rather than transferring float data through serial communication we map the data to -100 up to 100 and then convert it to the value between -1 and 1 after received by the car control program by dividing it by 100. Then the value obtained will be used for both steering and accelerating of the vehicle.

## 5.2 Recommendation and feature work

Virtual simulation is an emerging field nowadays with rapid growth, the application of this in different industry like medical institution, training center for different vehicles, and also entertainment center makes the area interesting and also makes it require great quality which needs to be worked on by every researcher in the area. This project can be further improved by adding for example it doesn’t provide the mechanism to make the driver feel the physical impact it is having while driving the car this could be like ups and downs while driving on non-flat terrain, a sense of vibration during collision and so on. This system doesn’t analyze the behavior of the driver by providing it with some input on the display could be a person standing on the road and recording how it reacts to the situation.

The following is the block diagram of the improved system with physical stimulation toward the person driving the car.

Person sends driving signal to Arduino.

The Arduino send the data to the system serially.

The system analyses the data received and decide the dynamics and different parameters of the driver seat.

Then the control signal to the driver seat will be sent to the driver seat controller.

Figure 27: block diagram of improved system with physical stimulation.

Generally, the system can be made better by adding physical simulation for the driver, and also by adding data analytical program to analyze the behavior of the driver which enable one to understand itself. The block diagram of how this improved system can be implemented have also been shown using the two diagrams above.

The following diagram is the block diagram of the system after improved with the analyses of behavior of the driver.

The display with different event will be fed to the driver.

Driver takes action based on the event and the signal is sent to the Arduino.

The Arduino takes the signal received and send it to the computer serially.

The computer will send the data to the data storing database and the car control program.

The database will store the data which will be fetched by data analysis program with the details of the event and driver.

The car will be controlled according to the signal received.

The data analysis program will decide/guess the behavior of the driver whom data is provided using different pre-defined algorithm.

Then the result will be again stored for future analysis and comparison for the driver profile.

Figure 28: block diagram of system improved with behavior analyzing capability.

# References

|  |  |
| --- | --- |
| [1] | public, "Wikipedia.org," Wikimedia Foundation, Inc, [Online]. Available: https://en.wikipedia.org/wiki/History\_of\_transport. [Accessed 18 12 2020]. |
| [2] | [Online]. Available: https://carsguide.com.au/car-advice. [Accessed 18 12 2020]. |
| [3] | Debela Deme Jima, et al "Full Length Research Paper Trafﬁc Accident Causes and Its Counter measures on Addis Ababa-Adama Expressway," pp. 17-20, 07 12 2018. |
| [4] | "www.igi-global.com," [Online]. Available: https://www.igi-global.com/dictionary/virtual-simulation/35045. [Accessed 18 12 2020]. |
| [5] | "bmclawyres.com," bmclawyers, [Online]. Available: https://www.bcmlawyers.com/what-percentage-of-car-accidents-are-caused-by-human-error/. [Accessed 19 12 2020]. |
| [6] | "axleaddict.com," [Online]. Available: https://axleaddict.com/safety/Easiest-Way-to-Learn-How-to-Drive-Manual. [Accessed 17 01 2021]. |
| [7] | happyroman, "vectorstock," [Online]. Available: https://www.vectorstock.com/royalty-free-vector/manual-gear-shifter-vector-2769035. [Accessed 04 January 2021]. |
| [8] | "https://www.steeringdevelopments.co.uk," [Online]. Available: https://www.steeringdevelopments.co.uk/wp-content/uploads/2019/05/electronic-gear-selector.jpg. [Accessed 15 01 2021]. |
| [9] | "www.sim-one.ca," [Online]. Available: https://www.sim-one.ca/content/virtual-simulations-virtual-patients. [Accessed 17 01 2021]. |
| [10] | "wikipedia," [Online]. Available: https://en.wikipedia.org/wiki/Serial\_communication. [Accessed 18 01 2021]. |
| [11] | Mrs. Smita Rukhande, et al "Implementation of Virtual Drive (3D Simulator)," *IRJET,* vol. V, no. 02, 2018. |
| [12] | Petr Bouchner, et al "CAR SIMULATION AND VIRTUAL ENVIRONMENTS FOR INVESTIGATION OF DRIVER BEHAVIOUR," 04 03 2015. |
| [13] | Dipl.-Ing. Andreas Thanheiser, et al "Virtual Electric Vehicle Design using Real-world Coupled Realtime Simulation," vol. V, p. 0051, 2012. |
| [14] | Kang, et al"A PC-based driving simulator using virtual reality technology," in *ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, 2004. |
| [15] | Ambar Yadav, et al"Driving Simulator," 2013. |
| [16] | J. Olstam, et al"Simulation of vehicles in a driving simulator using microscopic traffic simulation," 2006. |
| [17] | Mikael Lebram, et al "A Driving Simulator Based on Video Game Technology," 2006. |
| [18] | [Online]. Available: https://5.imimg.com/data5/CU/NF/MY-12011637/packing-rubber-band-500x500.jpg. [Accessed 18 01 2021]. |

**Appendix 1: The program code.**

**Car controller part:**

void Update()

{

if (!carSetting.automaticGear && activeControl)

{

if (Input.GetKeyDown("page up"))

{

ShiftUp();

}

if (Input.GetKeyDown("page down"))

{

ShiftDown();

}

}

}

public float car\_direction;

public bool brake\_value=false;

void OnMessageArrived(string msg){

if (msg == "b")

{

brake\_value = true;

}

else if (msg == "a")

{g

brake\_value = false;

}

else

// {

car\_direction = msg.split(",");

car\_direction = float.Parse(msg);

if (car\_direction != 0){

car\_direction = car\_direction/100;

}

else {

car\_direction = 0;

}

}

}

void OnConnectionEvent(bool success){

// Debug.Log(success?"Device connected" : "Device disconnected");

}

void FixedUpdate()

{

// speed of car

speed = myRigidbody.velocity.magnitude \* 0.7f;

if (speed < lastSpeed - 10 && slip < 10)

{

slip = lastSpeed / 15;

}

lastSpeed = speed;

if (slip2 != 0.0f)

slip2 = Mathf.MoveTowards(slip2, 0.0f, 0.1f);

myRigidbody.centerOfMass = carSetting.shiftCentre;

if (activeControl)

{

if (controlMode == ControlMode.simple)

{

accel = 0;

brake = false;

shift = false;

if (carWheels.wheels.frontWheelDrive || carWheels.wheels.backWheelDrive)

{

// Debug.Log(car\_direction[0]);

steer = Mathf.MoveTowards(steer, Input.GetAxis("Horizontal"), 0.2f);

accel = Input.GetAxis("Vertical");

brake = false;

// Debug.Log(Input.GetKey(KeyCode.LeftShift));

shift = Input.GetKey(KeyCode.LeftShift) | Input.GetKey(KeyCode.RightShift);

}

}

else if (controlMode == ControlMode.touch)

{

if (accelFwd != 0) { accel = accelFwd; } else { accel = accelBack; }

steer = Mathf.MoveTowards(steer, steerAmount, 0.07f);

}

}

else

{

accel = 0.0f;

steer = 0.0f;

brake = false;

shift = false;

}

**Arduino part:**

//led ls rs bl r f n to rs ls bs

// 6 5 3,4 7, 8, 9 13 10 11 12

float steer;

float brake;

float accelaration;

void setup() {

// put your setup code here, to run once:

Serial.begin(19200);

for (int i=3;i<7;i++){

pinMode(i,OUTPUT);

}

for (int j=7; j<14; j++){

pinMode(j,INPUT);

}

}

void loop() {

// put your main code here, to run repeatedly:

//the whole program starts only when this switch is pressed

if(digitalRead(13)){

//left side

if (digitalRead(11)){

while(digitalRead(11)){

digitalWrite(6, HIGH);

delay(2000);

digitalWrite(6, LOW);

}

}

else if (!digitalRead(11)) {

digitalWrite(6, LOW);

}

//right side light

if (digitalRead(10)){

while(digitalRead(11)){

digitalWrite(5, HIGH);

delay(2000);

digitalWrite(5, LOW);

}

}

else if (!digitalRead(10)) {

digitalWrite(5, LOW);

}

//brake light

if (digitalRead(12)){

digitalWrite(3, HIGH);

digitalWrite(4, HIGH);

}

else if (!digitalRead(12)) {

digitalWrite(3, LOW);

digitalWrite(4, LOW);

}

//reverse/forward/neutral

if (digitalRead(7)){

Serial.print("r");

}

else if (digitalRead(8)) {

Serial.print("f");

}

else if(digitalRead(9)){

Serial.print("n");

}

delay(200);

//steer

steer = map(analogRead(A0), 0,1023, -100,100);

Serial.print(steer);

delay(200);

//acceleration

accelaration = map(analogRead(A1), 0,1023, -100,100);

Serial.print(accelaration);

delay(200);

//brake

brake = map(analogRead(A2), 0,1023, -100,100);

Serial.print(brake);

delay(200);

}

else {

Serial.print(“turn on the car”);

}

}

**Appendix 2: Timeline/schedule**

**Appendix 3: Budget**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| R.no | Item name | Pcs. | Price per Pcs. (ETB) | Total (ETB) |
|  | Rotary potentiometer | 3 | 35 | 105 |
|  | Resistor | 6 | 3 | 18 |
|  | LED | 6 | 3 | 18 |
|  | Electrical switch | 3 | 25 | 75 |
|  | Arduino Uno | 1 | 550 | 550 |
|  | Jumper wires | 1 pack | 150 | 150 |
|  | Hot glue | 1 | 1150 | 1150 |
|  | Tape | 3 | 20 | 60 |
|  | | | | 2126.00 |