# **Self-Driving Car**

# **Project Report**

****

V 1.0

**Table of Contents**

[**Self-Driving Car** 0](file:///C:\Users\mukund.aggarwal\Desktop\RCCarIntelEdison.docx#_Toc466915246)

[**Project Report** 0](file:///C:\Users\mukund.aggarwal\Desktop\RCCarIntelEdison.docx#_Toc466915247)

[**Project Overview** 2](#_Toc466915248)

[**Objectives** 2](#_Toc466915249)

[**Detail Description** 2](#_Toc466915250)

[**Features** 3](#_Toc466915251)

[**Points considerations** 4](#_Toc466915252)

[**Challenges** 5](#_Toc466915253)

[**Usability** 5](#_Toc466915254)

[**Limitations** 5](#_Toc466915255)

[**Capability Enhancements** 6](#_Toc466915256)

[**Appendix A: Hardware and Software** 6](#_Toc466915257)

# **Project Overview**

We aim to build a solution that can be used for various purposes in the present era of automation. We are building a car that is having features to make it autonomous and self-sufficient to do a number of things like Track Following, Obstacle avoiding, distance measuring etc. We can easily increase the functionality according to our requirement and new features to meet the present demand.

# **Objectives**

Our objective is to build a self-driving car. Car will be having features like obstacle avoiding, edge detection, cover a particular path with curves and avoid collisions. We will use various sensors to do the same with the Intel Edison board.

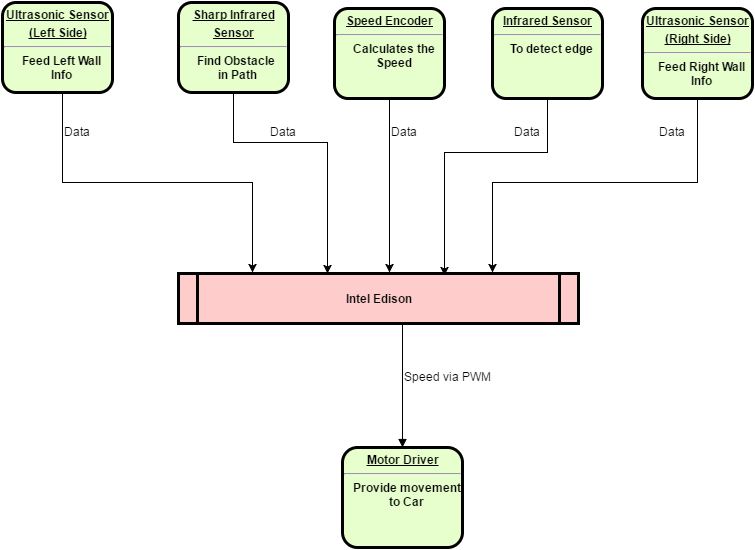
# **Detail Description**

The Obstacle avoidance/Track following car is always one of the favorite starter project of the College Students and the level goes so high that many working professionals engage in the same to take it to a next level. For example, google is engaged in making the autonomous car in the real scenario. Even the use cases are numerous and not just confined to the competitions for instance in the industries and warehouses where robotic cars/cranes autonomously pick and place the components, several robotic cars send to the outside the space, Bomb detecting and diffusing vehicle, all-terrain vehicle etc. The use case and the scope of the project is limitless.

Sensors sense the various factors required by the car to start and cover the track. Like ultrasonic sensor send the obstacle feed in the way to the microcontroller. The controller looks the data and analyses the next step for the motors which are driving the car. It then sends the signal to the speed controllers. Speed is continuously tracked by the Speed encoder and taking an account of this value, controller lowers or increase the speed. The speed control is done on the PWM pins of the controller.

As you can see the Block diagram :

**Block Diagram**



# **[Features](#_Features)**

**Direction Control -** Motors places in the car are attached to the PWM (Pulse Width Modulation) pins of the Controller to control the speed of the car through the controller.

PWM is used to avoid the sharp turns and to provide smooth turning to the car. Car has various features

**Wall Collision Avoiding** – With the help of Sharp IR sensor and ultrasonic sensor, car will not collide with the wall at any time. It will try to defer from the position or will completely get stopped.

**Track Following** – Track Following is done through Ultrasonic sensors placed at the sides of the Car. Ultrasonic Sensor sense the distance from each wall and update the controller. Controller based on the Input provided by the sensors calculates the position of the car and decide the speed of the motor.

**Obstacle avoiding** – Obstacle is any hurdle present in the way. To detect the obstacle Sharp IR sensor is used. Sharp IR sensor can detect the obstacle from approx. 80 cm. We are using it right now to find two types of obstacle i.e. long distance and short distance obstacle. When the car will meet long range obstacle, car will try to avoid the obstacle by moving to the other lane. While in the case of short range obstacle, car will stop completely.

**Edge Detection** – Car is provided by a sensor to detect the edge and avoid the falling of the car while moving in forward direction. The IR sensor placed in the Front of the Car continuously detect the path and signals the controller when it does not able to find the path below it. It is placed at a distance of the car to help the car to get stopped when moving with a speed providing enough time to car to deaccelerate.

**Night Mode** - Car has the feature that it can cover a track in the night. It does not requires a light source to cover a distance provided that no obstacle is present in the path.

**Working**

All the components are directly attached to the Intel Edison board which provides the Intel Atom processor which is a very powerful processor to work on the data provided by the Sensors to run the car. The car starts by sensing the distance from the Left wall and Right wall from the two Ultrasonic Sensors. Intel Edison works on this data and tries to calculate the position of the car. According to the calculations it provides the PWM output to the Motor drivers. PWM means pulse Width Modulation which is used to control the Voltage level of the motor so our speed of the motors can be controlled. Using the PWM helps us in avoiding the sharp turns taken by the Car and provides a smooth turning experience. While moving, Sharp IR sensor gets activated to sense any obstacle in the path. Sharp IR sensor is provided at the left side of the car (later we need to provide with more sensors). Once the Sharp IR detects an obstacle, it signals the controller about the obstacle and controller then turn the car in the another direction to avoid collision. While turning, controller keep a check on the all on another side of the path to avoid collision from the wall. The car than keeps an account of the obstacle and does not try to get back in the lane. This is done through the Ultrasonic Sensor on the Right side. If Ultrasonic Sensor gets the empty space on the original path, the car again comes back in the original path. All the things are right now working fine for the Right hand driving and yet to be tested for the left hand side driving though it can work fine on any side without any obstacle.

# **Points considerations**

1. Car is pre-loaded with three modes to move in a particular row i.e. Right, Left and Center.
2. All the sensors used in the car are modular and are easily replaceable if they malfunction. There is no need to dissemble the whole PCB and do the assembling again.
3. The code is very generic and adding more utilities is easy.
4. With the help of wall collision avoider, care will not collide with the wall in any circumstances.
5. Speed of the car is manageable.
6. Power supply Unit is capable of providing various voltages if there will be devices work on different voltage level.
7. Chassis of the car is very light and after placing all the components and battery, the weight of the car is still less as we tried to choose the chassis of acrylic material instead of iron or steel.

# **Challenges**

1. Making the sensors modular so that they can be replace easily.
2. Power management of the board as the Intel Edison board and the motors sometimes consume a large amount of current which was not possible to provide through normal battery.
3. Tracking the obstacle at the time of turning is a tough call and therefore we have coded in the car that it can track a particular type of obstacle and at a particular distance. The distance is decide on the basis of the min radius of the track so that car will not think the wall as the obstacle.

# **Usability**

Other use cases are as follows

• Robotics Competition

• Warehouses – Automatically pick and place goods.

• Toys

• Drones (Technology is same)

• No man Zones – These things are suitable for conditions like radioactive places, etc.

• Bomb detecting and Spy car – For the Army purpose

• Soil collector – Collect soil at various places

• Carry goods – Big cranes are used to carry goods from one place to another

• Fire Brigade – Are very useful to send in the area fire with fire extinguishers.

• Autonomous car (Google technology to drive the car without driver)

# **[Limitations](#_Limitations)**

1. The curves of the track should be known.
2. Size and Shapes of the obstacle should be pre-defined. Like right now the car will not be able to detect any obstacle which is short in height, short in dimension etc.
3. The Obstacle should be present on the right side of the track.
4. The surface should clean and smooth otherwise tyres will not be able to run smoothly on the track.
5. The edge detection is present in the front of the car only. It will not detect any edge from sides or back of the car.
6. Car will not be able to detect the obstacles in the dark. It needs light to detect the obstacles.

# **Capability Enhancements**

The car made right now is in the starting phase and yet there are lot of features that can be added in the car to make it more autonomous and Intelligent.

1. We are replacing the sensors with the Intel Real Sense camera to trace the path and obstacle. Than we will be able to provide the Car the Real World Inputs which it really has to face in the real scenarios.
2. Sensors used are not of the industrial grade. If required we can go for the sensors that can provide us the feedback of the car according to the data provided by them.
3. Chassis of the car is of acrylic material which is not very robust for some scenarios. We can use more robust material for the future version.
4. For the prototype we are using low speed motors and that too on 5 v instead of 12 volt to increase the ease in our development. But we can go for the high speed motors.
5. We are planning to use the WIFI to upload the code instead of using wires for the same.

# **Appendix A: Hardware and Software**

The car uses a number of sensors to collect the data, microcontroller at different level to analyze the data.

Sensors are basically used for the purpose of the obstacle avoidance, controlling the speed of the car.

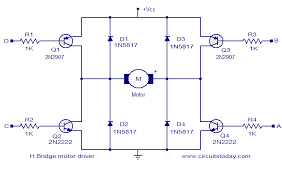
**Intel Edison**

Intel® Edison Development Platform is the first in a series of low-cost, product-ready, general purpose compute platforms that help lower the barriers to entry for entrepreneurs of all sizes—from pro-makers to consumer electronics and companies working on the Internet of Things (IOT). Intel® Edison packs a robust set of features into its small size, delivering great performance, durability, and a broad spectrum of I/O and software support.

**Power Supply Unit**

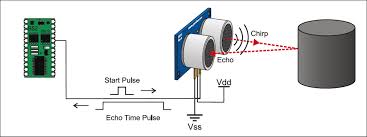
Power supply is one great need for any component to work. Sometimes the need of the power for every device varies and the power supply unit should be able to provide the exact requirement of the component. Presently we are using two 3.7V lithium Ion battery with a 5 volt limiter to provide the exact voltage for the devices. Although, we can use various voltage level in future if required. We can also use the Power adapter provided by the Intel Edison Board.

**Electronic Speed Controller**

****

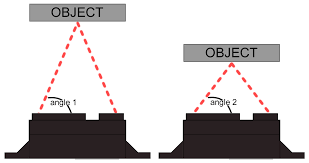
The electronic speed controller is the component that drives the motors according to a PWM signal it receives from the Arduino. It is connected to a 5V battery.

**Ultrasonic distance sensors (HC-SR04)**



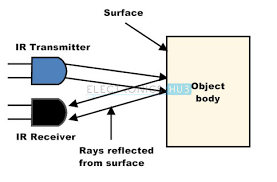
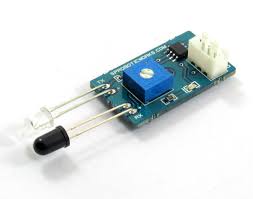
The ultrasonic sensors calculate distance by transmitting an ultrasonic wave pulse and measuring how much time it took to return, after reflecting on an object. Each sensor is connected to two common (digital) Arduino pins.

**SHARP infrared sensors**

****

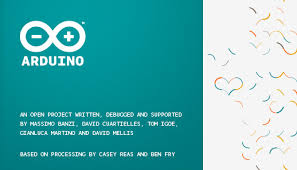
The infrared sensors work in a similar manner to the ultrasonic ones. Depending on the distance of the object that they face, they return the equivalent amount of voltage. Each of them is connected to one analog pin on the Arduino. Keep in mind that the specific sensors can reliably measure distances between 4 and 80 centimeters.

**Speed Encoder**

****

The speed encoder is attached to a wheel and allows us to calculate how much distance it has traveled. Having decorated the inside of the wheel with black stripes, the sensor changes its output state whenever it detects a black surface. Therefore, by counting how many state changes have taken place and knowing how much the wheel travels while turning, we can estimate a pulses/distance ratio. The encoder is attached to an interrupt on the Arduino.

**Software**



**Arduino IDE** – Intel Edison has the same Arduino Uno pin compatibility. Plus it provides the convenience to use the same software which we use for the Arduino to control the Arduino side of the Intel Edison Board