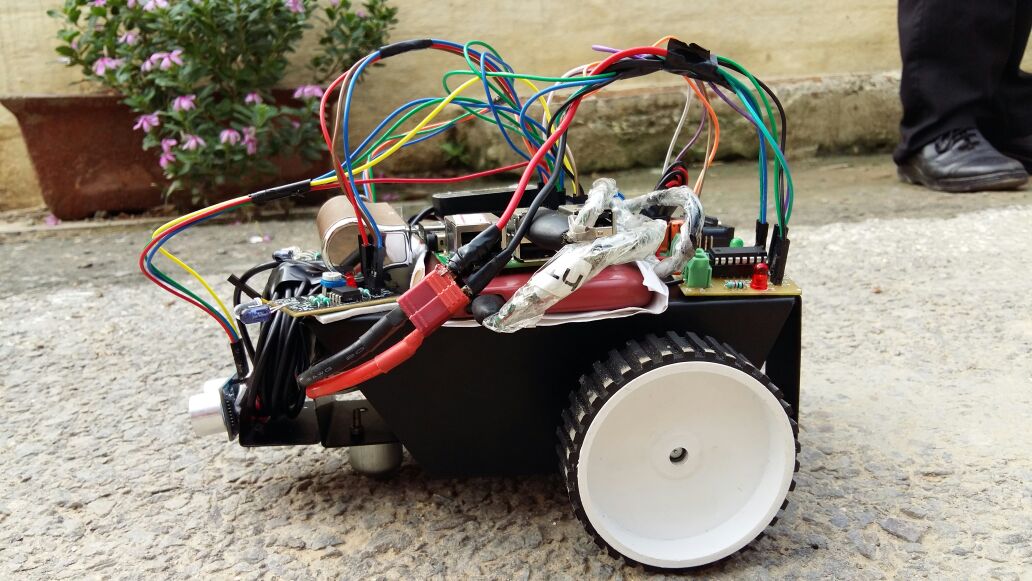
**SMART CAR**

****

##### A SUMMER TRAINING PROJECT REPORT

###### **Submitted by**

###### AASHNA KAINTH

###### 

Under the guidance of

Dr. SRN Reddy

HOD CSE DEPT,

Associate Professor, IGDTUW

***in partial fulfillment for the award of the certificate***

***in***

**“Build Your Smart ioT Device”**

**Organizing by CSE Dept, Indira Gandhi Delhi Technical University For Women, Delhi in collaboration with Microsoft**

(From 6-06-2016 to 15-07-2016)

**CONTENTS**

|  |  |
| --- | --- |
| **Certificate** | **I** |
| **Acknowledgement** | **Ii** |
| **Abstract** | **iii** |
| **List of Publications** | **Iv** |
|  |  |
|  |  |

**CERTIFICATE**

This is to certify that Summer Training Project Report entitled “**SMART CAR**” submitted by AASHNA KAINTH (06),RICHU (17), YUKTI SHARMA (19) is an authentic work carried out by them at Indira Gandhi Delhi Technical University for Women, Delhi under my guidance during STP7 in 2016. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Date: Signature of the Guide

**Dr. S.R.N Reddy**

**HOD CSE DEPT**

**Associate Professor**

**Kashmere Gate, Delhi, 110006**

Signature of the Students

**ACKNOWLEDGEMENT**

I take this opportunity to express a deep sense of gratitude to Mr. SNR Reddy, Dept of Computer Science Engineering, IGDTUW for their guidance ,help and useful suggestions throughout the project.

I express my gratitude to the embedded lab staff of cse department for providing me the infrastructure to carry out the thesis work and I would also like to thank all my classmates for their valuable suggestions and helpful discussions.

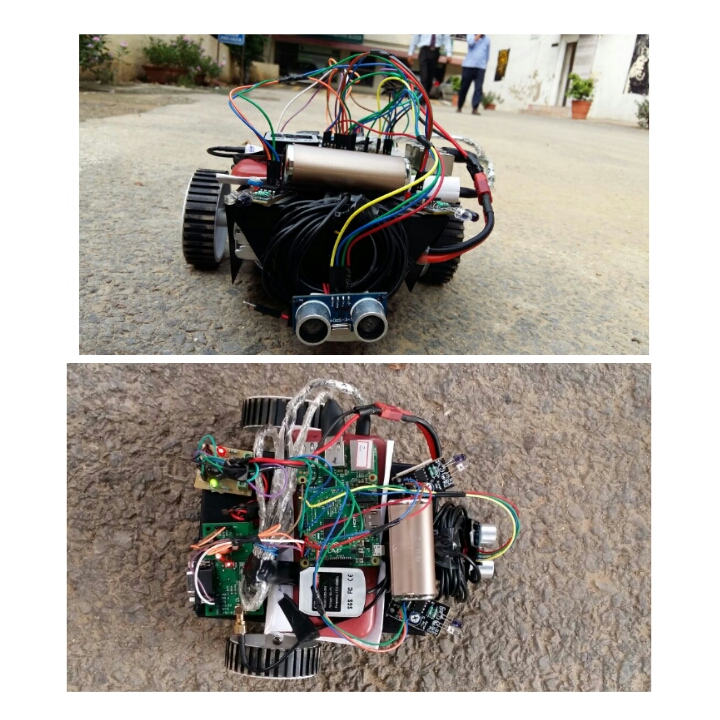
**AASHNA , YUKTI , RICHU**

**B.Tech, IGDTUW**

**ABSTRACT**

The project aims to build an autonomous car prototype using Raspberry Pi as a processing chip. Infrared sensors along with an ultrasonic sensor is used to provide necessary data from the real world to the car. The car is capable of reaching the given destination safely and intelligently thus avoiding the risk of human errors.

Problem statement: We will be giving our robot coordinates of starting and finishing point and it has to drive it’s way down to the final destination and avoid any obstacle in the path.



**CHAPTER-1 : INTRODUCTION**

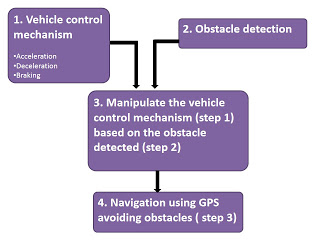
**1.1 Overview**

The degree of vehicle automation is continuously rising in all modes of transport both on public traffic infrastructure and in-house transport within company grounds, in order to improve the productivity, reliability, and flexibility of transport. In this way the concept of an Autonomous Car is put forward.

An autonomous car (driverless car, self-driving car, robotic car) is a vehicle that is capable of sensing its environment and navigating without human input.

Autonomous vehicles detect surroundings using radar, lidar, GPS, odometry, and computer vision. Advanced control systems interpret sensory information to identify appropriate navigation paths, as well as obstacles and relevant signage. Autonomous cars have control systems that are capable of analyzing sensory data to distinguish between different cars on the road, which is very useful in planning a path to the desired destination. The biggest benefits of self-driving cars are that they will help to make roads safer and people's lives easier.

1.2 Generic Autonomous Car Design

****

1. Vehicle control mechanism:

We must care about the vehicle control mechanism with the maximum payload of that vehicle based on which the components can be picked up.

There are two ways by which the vehicle can be controlled

1. Fixed speed control

2. Variable speed control

In fixed speed control, the vehicle can be either moved(with a constant speed) or stopped. A simple relay logic controller is enough to perform this type of control. Logic sequence can be sent via the microcontroller to the input of the relay driver to switch the motors ON/OFF as well as to control their rotating direction.

2. Obstacle detection:

There are several ways of detecting obstacles in robotics. To keep it as simple as possible for beginners, I consider the ultrasonic proximity sensor(for eg: HC SR04). The obstacle sensing range can be set as you wish and the sensor can be mounted over a servo motor to get 180 degree field of view. Don’t forget to use a pull up resistor to avoid garbage values in the sensor readings. Also provide enough delay time in the servo sweep so that the echo pulse can reach the receiver completely.

3. Manipulating the vehicle based on obstacle detected:

Let us consider the maximum range of the sensor is 4 meters and anything within 2meters is considered as obstacle. The following algorithm can be used to control the vehicle based on the sensor reading,

Function name() {

Read sensor value;

If (value>2meter)

Move forward

Else

Activate servo and scan for obstacles in left and right side of vehicle

If (more space in left than in right)

Move left

Else

Move right

} // continue loop

This algorithm detects the obstacle and manipulates the vehicle based on the sensor values. And the resultant vehicle will be an obstacle avoiding vehicle(with no motive, i.e, it moves randomly avoiding obstacles)

4. GPS based navigation:

The components required for this purpose are the GPS receiver/shield and a compass. Using the Haversine formula, the distance from the source to destination point can be calculated. Since the distance is calculated in straight lines, way points must be declared in case of destination points other than straight lines. Using the compass, determine the vehicles direction with respect to the destination point and turn the vehicle until both the values are equal (use atan2() to get the vehicle`s current direction with respect to destination using the GPS coordinates). Then, move the vehicle to the calculated distance (measure the distance moved using an encoder) and “you will reach your destination point”(with lot of errors obviously). While moving from the source to destination point, the obstacle detection and manipulation function will act as an interrupt function.

1.3 Project Background

The purpose of the project is to design an autonomous ground vehicle capable of navigating within the provided outdoor obstacle course. The robot must avoid all obstacles, while remaining within the marked lane and guide flags.

We will be using following components in our project:

**Raspberry Pi**

The Raspberry Pi is a single board computer that runs at 250 MHz based on the Broadcom

BCM 2835 system on chip. The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python.

**Ultrasonic Sensors**

Ultrasonic ranging sensors will be used to avoid collisions at times where the laser scanner and cameras cannot. Ultrasonic range finders are used because of their “bubble like” 3 dimensional forward field of sensitivity. Many ultrasonic range finders will be placed all over the UGV in order to create a fully enclosed detection zone.

**IR Sensors**

These sensors are used to detect any obstacle in left and right side of the vehicle. Whenever the vehicle wants to take a turn then these sensors become active . If they show no obstacle then only the vehicle turn that side.

**GPS**

The objective of the competition is to navigate to 8 GPS waypoints, so GPS will be critical to

winning the competition. The GPS pose will be combined with every other pose estimate to negate the built in error of 5 meters. This will make sure that the robot has an accurate measurement that doesn’t jump or drift throughout the round.

**Motors**

The UGV will be using four 24V, 400W, 2600 RPM electric scooter motors. These motors are rated at 22A each and weigh about 6lbs each. These specifications are what we expect since we wanted to make the design as rugged and reliable as possible. The motors will be capable of maneuvering effectively on all terrains expected at competition. Also, these motors will be capable of propelling our robot’s weight and the 20lb payload that is required during navigation at the competition.

**Bluetooth**

It is also a popular mechanism for short distance point to point or point to multi-point communication. Bluetooth has been designed to enable cordless connectivity between a multitude of devices and user terminals. Bluetooth is being used to connect: keyboards with computers, hands-free ear-pieces to phones, laptops to printers and to the internet. Another important usage for Bluetooth is to connect wireless user terminals, like phones or PDAs (Personal Digital Assistants) to different information servers.

1.4 Thesis objective and Scope

Every year, 1.2 million people die in automobile accidents and up to 50 million are injured.

Many of these deaths are due to driver error and other preventable causes. Autonomous

or highly aware cars have the potential to positively impact tens of millions of

people. Building an autonomous car is not easy. Although the absolute number of traffic

fatalities is tragically large, the failure rate of human driving is actually very small.

Traffic accidents are a major source of disability and mortality worldwide. Every year,

1.2 million people die and up to 50 million people are injured . Autonomous or highly

aware cars have the potential to reduce these numbers dramatically. However, building a

self-driving car that exceeds human driving performance is not easy. While traffic injuries

and deaths are large in number, they are relatively infrequent. In the United States, a

human driver makes a fatal mistake only once in approximately 88 million miles . As a

co-founding member of Stanford’s Racing Team, we have built several relevant prototypes of autonomous cars. Work on self-driving cars spans several decades . The DARPA Grand and Urban Challenge competitions offered a modern, uniform testing opportunity to examine the state-of-the-art in autonomous cars.

The problem statement expanded significantly to include much of what a typical human driver must do. For example, car detection, lane keeping, parking, and intersections were included. Important exclusions were high-speed (highway) driving, traffic lights, bicycles,

pedestrians, and the ability to drive without a highly accurate GPS course skeleton. Several

teams were successful although reliability was not perfect despite the abridged definition of

the problem. Nevertheless, it is remarkable how quickly the technology progressed. Today,

the Google driving project is expanding the code base from these efforts.

Of course, self-driving cars will not reduce the number of trips or kilometres travelled. On the contrary: self-driving cars have the potential to significantly lower the total cost per kilometre travelled and are thus likely to induce people to make more trips. Self-driving taxis and buses will emerge rapidly and offer mobility services for local and long distance traffic with great convenience and at extremely competitive prices because they can achieve much higher utilization rates than private cars (which stand idle more than 94% of the time), and because autonomous fleet vehicles will be engineered for the minimization of total cost of ownership and for the maximization of useful life.

Most urban self-driving taxis will be fully electric for reasons that are not primarily environmental but that are still good for the environment: Electric motors offer safety advantages (they can be used for emergency braking and to some degree for emergency steering). They are also much more durable (an electric motor easily lasts 1 million kilometres), less expensive and less complex than conventional engines. In addition self-driving taxis that operate in local traffic will not need huge battery packs when average trip sizes rarely exceed 15 kilometres and when they can drive themselves to the next high efficiency charging station as needed. Their batteries won’t be sized to last a whole day; they will need to be just large enough to service a little more than the trips of the morning peak – after which they can recharge.

There can be no doubt that self-driving taxis and buses will change the nature of urban mobility. Much more short-distance travel than today will occur in small, lightweight, extremely energy efficient self-driving taxis. Although this may lead to a certain increase in total miles travelled, the following effects combine to reduce greenhouse gas emissions:

Self-driving taxis will be mostly electric which reduces carbon emissions (approximately 25% less emissions compared to internal combustion engine)

Self-driving urban taxis will be smaller and much lighter than the average car which further reduces energy consumption per kilometre.

Self-driving taxis reduce demand for private cars and therefore reduce the sizable greenhouse gas emissions during vehicle manufacturing which are typically more than 10% of total life-cycle emissions of a car. According to some estimates, a self-driving car-sharing vehicle or taxi can eliminate 7 to 10 private cars. What a potential for greenhouse gas reduction in auto manufacturing!

Self-driving taxis facilitate multi-modal travel (taking an autonomous taxi to the train or bus station, continuing with bus or train, using an autonomous taxi for local transport at the destination)

Self-driving taxis facilitate ride sharing especially during peak hours and on certain routes.The big advantage of self-driving car technology is that it can accomplish several benefits at the same time: It increases the options for individual mobility and lowers the cost of individual mobility because of new driverless mobility services which through increased sharing, more efficient use and quicker adoption of alternative fuels reduces greenhouse gas emissions. Nobody will have to abandon their cherished car but the joint actions of the large group of less or only moderately affluent consumers who value the flexibility and cost-saving associated with self-driving mobility services will inexorably lead to a reduction of greenhouse car emissions. It is time for the political leaders searching for solutions to combat climate change to take notice!

1.5 Applications of the System

The main drivers for achieving autonomous driving is the reduction of traffic accidents by eliminating human error, increasing road capacity and traffic flow by reducing distance between cars and making use of traffic management information, relieving the car occupants from driving and navigation activities and allowing them to engage in other activities or rest. The biggest benefits of self-driving cars are that they will help to make roads safer and people's lives easier.

* Avoid traffic collisions (and resulting deaths and injuries and costs) caused by human driver errors such as reaction time, tailgating, distracted or aggressive driving.  
  Reduction in labour costs if human driver isn't required.
* Increased roadway capacity and reduced traffic congestion due to reduced need for safety gaps and the ability to better manage traffic flow.
* Removal of constraints on occupants' state – in an autonomous car, it would not matter if the occupants were under age, over age, unlicensed, blind, distracted, intoxicated, or otherwise impaired.
* Reduction in the need for traffic police and premium on vehicle insurance.
* Reduction of physical road signage – autonomous cars could receive necessary communication electronically (although physical signs may still be required for any human drivers).
* Increased time in daily leisure activities or work productivity with the replacement of commuting hours.

The big advantage of self-driving car technology is that it can accomplish several benefits at the same time: It increases the options for individual mobility and lowers the cost of individual mobility because of new driverless mobility services which through increased sharing, more efficient use and quicker adoption of alternative fuels reduces greenhouse gas emissions. Nobody will have to abandon their cherished car but the joint actions of the large group of less or only moderately affluent consumers who value the flexibility and cost-saving associated with self-driving mobility services will inexorably lead to a reduction of greenhouse car emissions. It is time for the political leaders searching for solutions to combat climate change to take notice!

**CHAPTER-2 : Existing Technologies**

Such vehicles are an applied use of increasingly sophisticated artificial intelligence and robotics capabilities. These technological advances are allowing society to fundamentally reconsider the vehicles available to us, and the infrastructure which they are part of. This report looks at two major categories of autonomous and unmanned vehicles: (i) autonomous cars and (ii) unmanned aerial systems (UAS), as well as briefly considering other types of autonomous and unmanned vehicles. The development, as well as the commercial, public and consumer applications of these technologies will be considered, followed by the accompanying risks, and the foreseeable implications for insurers.

The forces behind the development of these technologies are diverse, and represent how traditional transport industries can undergo changes in response to a new generation of machinery. Taking part in the autonomy race, in addition to established car and aircraft manufacturers, are global technological and idea innovators such as Google and Amazon, components manufacturers such as Bosch and Continental, small start-ups and university researchers. Their achievements could represent the biggest change to vehicles since the motorcar replaced the horse and cart. Autonomous vehicles are vehicles which are capable of driving themselves. In order to do this, the vehicle must be able to perceive its environment, make decisions about where is safe and desirable to move, and do so.

**Self driving Car made by GOOGLE:**

Google has created an autonomous car because it is believed that Aging or visually impaired loved ones wouldn't have to give up their independence. Time spent commuting could be time spent doing what you want to do. Deaths from traffic accidents—over 1.2 million worldwide every year—could be reduced dramatically, especially since 94% of accidents in the U.S. involve human error.

In Google cars, they navigate through many complicated scenarios on city streets.

Their cars use their sensors and software to sense objects like pedestrians, cyclists, vehicles and more, and are designed to safely drive around them. Like any driver, a self-driving car needs to constantly answer these questions.

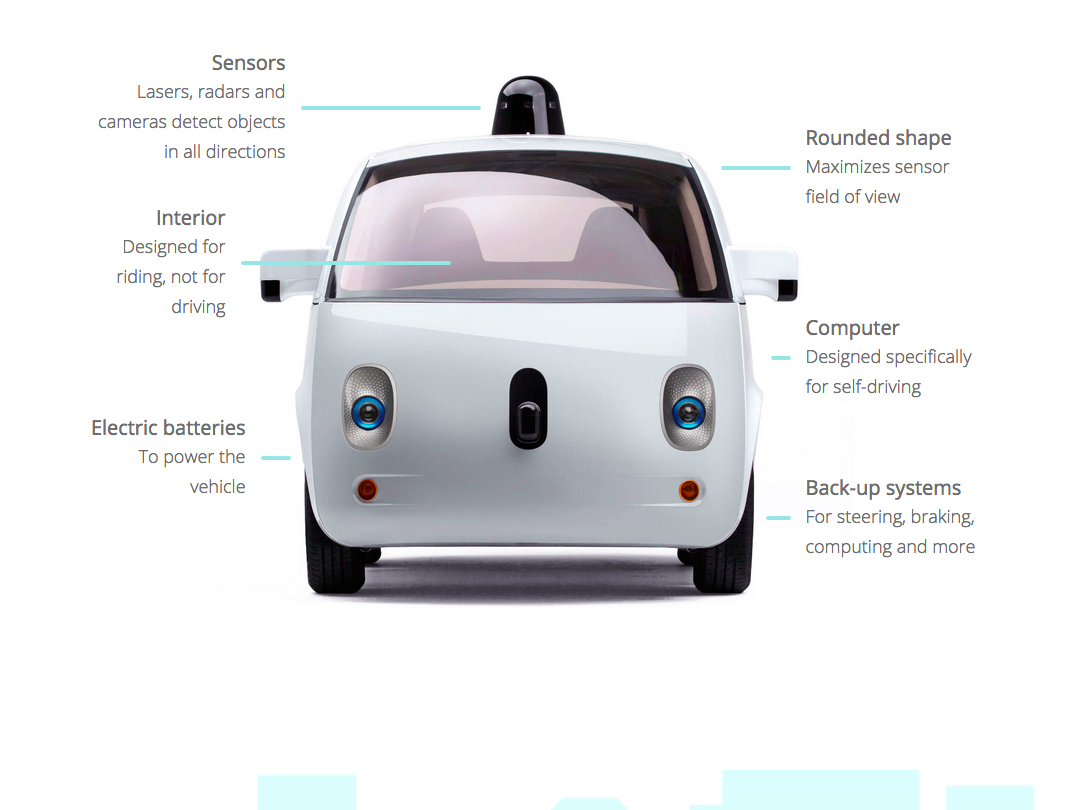
The car processes both map and sensor information to determine where it is in the world. The car knows what street it's on and which lane it's in.

Sensors help detect objects all around us. The software classifies objects based on their size, shape and movement pattern. It detects a cyclist and a pedestrian in this case.

The software predicts what all the objects around us might do next. It predicts that the cyclist will ride by and the pedestrian will cross the street.

The software then chooses a safe speed and trajectory for the car. The car nudges away from the cyclist, then slows down to yield to the pedestrian. Their self-driving prototypes rely on their sensors and software to drive themselves.

Now Google is working towards vehicles that take you where you want to go at the push of a button. They started by adding components to existing cars like our Lexus SUVs, then began designing a new prototype from the ground up to better explore what should go into a fully self-driving vehicle. They removed the steering wheel and pedals, and instead designed a prototype that lets the software and sensors handle the driving.



Google have been working on our project since 2009, but the dream of self-driving cars goes back much farther.

It started as early as the 1939 New York World’s Fair where visitors were presented a vision of automated highways. In the mid 2000s, the Defence Advanced Research Projects Agency (DARPA) organized the Grand Challenges where teams gathered to compete with self-driving vehicles. In 2009, Google started the self-driving car project, including team members who had already dedicated years to the technology.

In 2009, they started testing their self-driving technology with the Toyota Prius on freeways in California.

In 2012, they began testing with the Lexus RX450h. At this point, we had completed over 300,000 miles of testing on freeways.

Next, they shifted focus to city streets, a much more complex environment than freeways.

They unveiled an early construction of our new prototype vehicle in 2014. It’s designed from the ground up to be fully self-driving.

After months of testing and iterating, they delivered the first real build of our prototype vehicle in December 2014.

They’ve self-driven more than 1.5 million miles and are currently out on the streets of Mountain View, CA, Austin, TX, Kirkland, WA and Metro Phoenix, AZ.

Their testing fleet includes both modified Lexus SUVs and new prototype vehicles that are designed from the ground up to be fully self-driving. There are test drivers aboard all vehicles for now. They look forward to learning how the community perceives and interacts with us, and uncovering situations that are unique to a fully self-driving vehicle.

**Tesla Autonomous Car:**

Tesla released a self-driving feature called autopilot to customers in a software update last year. The electric carmaker, led by tech billionaire Elon Musk, says those who choose to participate in the “public beta phase” will help refine the technology and make streets safer sooner. Tesla drivers already had logged some 130 million miles using the feature before a fatal crash in Florida in May made it the subject of a preliminary federal inquiry made public on Thursday.

The divergent approaches reflect companies with different goals and business strategies. Tesla’s rapid-fire approach is in line with its image as a small but significant auto industry disruptor, while Google — a tech company from whom no one expects auto products — has the luxury of time.

Tesla’s Model S has an advanced cruise control feature called Autopilot, which uses cameras and radar to detect the car’s position in lane, the proximity of other cars and the speed limit. It can control the car’s speed and steering to keep it in the middle of the lane, reacting to other cars and changing lanes on command.

**Bosch Dedicates More Than 2,000 Engineers To Driver-Assistance Technology**

Bosch, one of the world’s largest automotive suppliers, has responded to an increase in demand by dedicating more than 2,000 engineers to driver-assistance systems. The company counts Google, Tesla, and Porsche as clients, and has managed to outfit two Tesla vehicles to make them fully autonomous (at a steep price). Bosch is also partnering with GPS maker Tom Tom for the mapping data necessary for this endeavor. The company has agreed with the projection that 2020 will see driverless cars in action, at least on highways. In an April 2016 interview, a Bosch marketing director reiterated the company’s commitment not just to the automation of vehicles, but connectivity and electric vehicles as well. It also recently considered taking a stake in the HERE mapping company.

**Delphi Retrofits Cars With Autonomous Tech, Partners With VC-Backed Startup**

Delphi, a large automotive parts supplier headquartered in the UK, has created a network of software and sensors that can be outfitted into existing car models to make them autonomous. Last April, an Audi SQ5 outfitted with Delphi technology drove itself 3,000 miles across the US, doing 99% of the driving by itself. In January 2016, Delphi showed off a new autonomous driving concept at CES. The concept’s human-machine interface attempts to address the stepping-stone stage before full (or Level 4) automation is ready. The car is designed to encourage consumers to trust that the car can drive itself, while still keeping drivers vigilant so they can take the wheel if necessary. Delphi has also partnered with the VC-backed company Quanergy to deploy low-cost LiDAR systems.

**Ford Announces Plan To Research Autonomous Vehicles**

In early 2015, Ford announced its “Smart Mobility Plan” to move the company forward on innovation, including vehicle connectivity and autonomous vehicles. This plan culminated in the formation of Ford Smart Mobility LLC in March 2016, a new subsidiary focused on connectivity, autonomous vehicles, and mobility (e.g. car- and ride-sharing services). As part of its 10-year autonomous vehicle plan, Ford also announced that it would triple its test fleet to 30 total vehicles in January. It has pioneered the testing of self-driving cars in less friendly environments, such as snowy Michigan, as well as in complete darkness.

**General Motors Investing Heavily in Autonomous R&D, M&A**

General Motors has made waves in 2016 with a series of aggressive moves within the tech sphere. In January, the company bought up Sidecar‘s assets and invested $500M into Lyft, announcing plans to develop an on-demand network of self-driving cars. As mentioned above, March also saw GM’s landmark acquisition of autonomous tech startup Cruise Automation. Separately, GM has also been developing its own semi-autonomous technology in -ouse, with its Super Cruise technology slated to come to market on high-end Cadillac models in 2017. The Cruise acquisitions is unrelated to this product launch, however; GM has said that the Cruise acquisition would have “no impact” on its Super Cruise launch.

**Honda Also Testing Autonomous Cars, Offers Semi-Autonomous Features On Civic**

Honda has received approval from California to test autonomous vehicles on public streets (with restrictions on the number of vehicles and the testing methods). Like Apple, the automaker is also using the GoMentum Station proving ground, with 2,100 acres of testing area for its self-driving fleet. Honda also introduced semi-autonomous ADAS (advanced-driver assistance systems) options on its entry-level Civic, offering lane-keeping, automatic braking, and adaptive cruise control functionality. These features are also available on luxury models (offered by brands like Tesla, BMW, etc.), but are notable on a vehicle with a $20,440 base price.

**Hyundai Steps Up Investments in AI, Connectivity and Self-Driving Cars**

After debuting a 2014 TV commercial that showed a convoy of cars outfitted with Hyundai’s driver-assistance tech, Hyundai sounded a more conservative note last September. Its European head Thomas Schmid asserted that autonomous driving would come “by far not as quick as everyone says,” giving a timetable of 10 to 15 years. Nevertheless, the Korean motor group seems to be intensifying its efforts to compete in 2016, ramping up investments in AI and setting up a new business unit to develop “hyper-connected” and self-driving cars in the near future.

**Jaguar Land Rover Wants To Introduce Assisted Driving But Keep Driving Fun**

In June 2015, JLR Director of Research and Technology Wolfgang Epple stated that autonomous vehicles would run counter to the brand’s philosophy, as the company “doesn’t consider its customers as cargo.” He asserted that the Tata-owned companies would instead favor advanced assistance features that would help drivers without taking full control from them. However, in Feburary 2016, JLR joined a $7.9M UK program to further autonomous driving R&D, aiming to gather data on driving habits and test vehicle communications technology.

**Mercedes’ Self-Driving Concept Car Will Be Ready In 15 Years**

Last January, Mercedes unveiled their concept for the sleek F 015 autonomous vehicle, which the company says will be ready in 15 years. Until then, the company is taking other steps toward self-driving cars. In 2013, its Intelligent Drive semi-autonomous research vehicle drove 60 miles on a German highway and streets. Certain Mercedes models have a “Stop-and-Go” mode, which allows the cars to navigate themselves while in traffic. The company also has approval to test the cars in California, and some have been seen driving in the streets. The company is also considering setting up a large fleet of autonomous limousines for on-demand access (which would presumably compete with or supply cars to car-hailing services).

Microsoft Begins to Forge Autonomous Partnerships With Automakers

Though late to the game compared to other tech giants, Microsoft has begun to dip its toes into self-driving car research. Its initial strategy appears to focus on collaborations, such as a November 2015 deal with Volvo that will see the companies collaborating in autonomous vehicle R&D and leveraging Microsoft’s HoloLens technology. Recently in March 2016, Microsoft and Toyota also announced the expansion of their five-year-old partnership to develop new vehicle connectivity and telematics services. The new organization will also support Toyota’s research in robotics, AI, and self-driving car development (see Toyota’s other partnerships below). Microsoft has also reportedly weighed taking a stake in the HERE high-definition mapping service, currently owned by BMW, Daimler, and Volkswagen.

Mobileye Powers The Driver-Assistance Technology Behind Self-Driving Vehicles

Mobileye, a $9B automotive supplier, provides many of the chips and advanced driver-assistance systems that are used by manufacturers for autonomous vehicles (including Tesla). The company has not announced plans to manufacture cars themselves.

Nissan/Renault Promises ‘Significant Autonomous Functionality’ By 2020

At last month’s New York Auto Show, Chairman and CEO of Nissan and Renault Carlos Ghosn promised that the group would have 10 vehicles on sale by 2020 with “significant autonomous functionality.” Nissan unveiled its first public prototype in 2013 at the Nissan 360 event in California, and has since been testing an autonomous Nissan LEAF on the roads of Tokyo. Just days ago, Nissan and Toyota also announced a joint effort to develop standardized “intelligent” maps, perhaps in response to German automakers’ acquisition of the mapping company here.

Nvidia Adapts GPU Technology To Create ‘Supercomputer’ For Self-Driving Cars

At CES in January 2016, GPU and semiconductor company Nvidia surprised many by unveiling the Nvidia Drive PX2, a powerful computing platform for autonomous cars. Packing 8 teraflops of processing power, the platform is robust enough to support deep learning, sensor fusion, and surround vision — all key elements of a potential self-driving car.

Toyota Expands Autonomous Partnerships, Collaborates With University Of MichiganToyota appears to have reversed course from its 2014 claims that it would not develop a driverless car on safety grounds. Last year, it announced a $1B budget for autonomous driving research. Toyota has also also hired professors and researchers from Stanford University, MIT, and the entire staff of the autonomous vehicle company Jaybridge Robotics. Just this April, it also announced its third US university partnership with an engineering powerhouse, the University of Michigan. Toyota plans to divide labor among its research partners, with the University of Michigan campus responsible for fully autonomous cars, Stanford working on partially autonomous vehicles, and MIT working on machine learning.

**Uber Looks to Develop Or Source Self-Driving Car Fleet**

Uber CEO Travis Kalanick is noted champion of his ride-sharing company embracing autonomous cars. The company has made several moves in that direction, such as poaching nearly the entire Carnegie Mellon Robotics Lab (40 engineers) to work on the project in Pittsburgh. Uber has also partnered with the University of Arizona to develop better mapping and optical safety technology. In 2015, the company acquired both mapping startup deCarta and mapping assets from Microsoft, which could be leveraged to drive autonomous vehicle efforts, among other initiatives. Rumors persist that Uber is also sniffing around major automakers to source a large order of self-driving cars. In March, Uber reportedly placed an order for up to 100,000 of Mercedes’ flagship S-Class, which are not yet fully autonomous but offer certain semi-autonomous functionalities.

**Volkswagen Pushes Autonomous Research As It Tries To Move Past Emissions Scandal**

In 2015, Volkswagen revealed the V-Charge project, where a Volkswagen e-Golf equipped with sensors, 3D maps, etc. will find open parking spaces in a garage and park without human input. The company suggests that there will be a prototype for demonstration available within four years. Besides this, in March 2016 VW Group CEO Matthias Muller announced that the board had just signed off on a huge autonomous driving initiative, boldly claiming that their goal was to “[bring] these technologies to market faster than the competition.” The Group’s head of digitalization asserts that self-driving cars will be “commonplace” by 2025.

It should be noted as of April 2016, Volkswagen’s market capitalization has been halved since its emissions scandal broke last September. This, combined with looming legal action and the loss of public trust, could hinder the development of autonomous vehicles across its subsidiaries (e.g., Audi, Scania, and so on).

**Volvo’s ‘Drive Me’ Program Targets Sweden, China, and Possibly US**

In addition to its trucking efforts detailed above, Volvo has also made progress with self-driving passenger vehicles. With a reputation for safety innovations, Volvo labeled its autonomous vehicle endeavors “Intellisafe,” with the goal of making Volvo cars “deathless” by 2020, when the company fully rolls out these features to the public. For now, Volvo is planning to give 100 Swedish customers early-access to an autonomous XC90 SUV in 2017 (with restrictions on the area, time, and context which autonomous mode will be used). The company has stated that it will accept full liability when its vehicles are in autonomous mode, and has just announced plans to expand its pilot program to China and the United States.

**Yutong Has Successfully Tested Driverless Buses**

Chinese bus manufacturer Yutong has been researching driverless buses since 2012. The company claims to have successfully navigated a bus on an inter-city road in central China’s Henan Province. The bus can switch between manual and automatic mode. The company has yet to announce a firm release date for its autonomous buses.

**CHAPTER 3 : Selection of Components**

There are several technologies used for monitoring and controlling the embedded systems. Also several processors, software programmers, GPS and Bluetooth modules are available in the market. We have to select one among this wide range of components which appropriately suits the design and helps in achieving the goals and objectives set for this project. This chapters gives details about the technologies and software programs and hardware components available in the market and explains about the selected ones.

**3.1 Selection of Technology**

The purpose of the project is to design an autonomous vehicle which is capable of navigating on road obstacle course. The robot must avoid all obstacles. We will be using Google API for navigation of the desired destination. Through GPS, we will be getting the present location. The desired destination will be fed by the user. And thus by using the Google API ,the smart car will be able to trace the path from starting point to the end point.

We will be using following technologies in our project:

**3.1.1 Internet (Web based)**

Even though the Internet is still a young technology, it's hard to imagine life without it now. Every year, engineers create more devices to integrate with the Internet. This network of networks crisscrosses the globe and even extends into space. Internet has also started to serve as a medium which allows the monitoring, control, and interaction with machine and devices. The Internet is used also in home automation which provides many features ranging from efficient use of energy to increased comfort, greater safety and security. Even from remote areas user can monitor and control the gate of his home, appliances and many other activities by using different sensors with the help of internet.

The Internet can be defined as the wired or wireless mode of communication through which one can receive, transmit information that can be used for single or multiple operations. The TCP/IP protocol makes the information transmission possible.

**Features of Internet**

* Remote Monitoring
* Real Time Monitoring
* Security

**Limitations of Internet**

* Additional cost of PC
* Internet access is mandatory

**3.1.2 Bluetooth**

Bluetooth is a highly versatile and attractive technology among other short range wireless technologies. It operates over an unlicensed frequency band of 2.4 GHz and links digital devices within a range of 10 meters to 100 meters at a speed of up to 3 Mbps depending upon the Bluetooth device class. Bluetooth technology was designed primarily to support simple wireless networking of personal consumer devices and peripherals, including cell phones, PDAs, and wireless headsets but now it is also used in automation that is if a user has to control or monitor devices within a limited range, Bluetooth is a cost effective solution.

Bluetooth networks have topology called a piconet. Piconets contain a minimum of two and a maximum of eight Bluetooth peer devices out of which one is master and others are slave. Slaves can only listen to master node whereas master is the only one which transmits. Devices communicate using protocols that are part of the Bluetooth Specification.

**Features of Bluetooth**

* Low operating cost
* Low power requirements
* Small size and fast response

**Limitations of Bluetooth**

* Limited range (up to few meters)
* No real time monitoring
* Access delay and Interference

**3.1.2 GPS (Global Positioning System)**

The **Global Positioning System** (**GPS**) is a space-based navigation system that provides location and time information in all weather conditions, anywhere on or near the Earth where there is an unobstructed line of sight to four or more GPS satellites. The system provides critical capabilities to military, civil, and commercial users around the world. The United States government created the system, maintains it, and makes it freely accessible to anyone with a GPS receiver.

The GPS concept is based on time and the known position of specialized satellites. The satellites carry very stable atomic clocks that are synchronized to each other and to ground clocks. Any drift from true time maintained on the ground is corrected daily. Likewise, the satellite locations are known with great precision. GPS receivers have clocks as well; however, they are not synchronized with true time, and are less stable. GPS satellites continuously transmit their current time and position. A GPS receiver monitors multiple satellites and solves equations to determine the precise position of the receiver and its deviation from true time. At a minimum, four satellites must be in view of the receiver for it to compute four unknown quantities (three position coordinates and clock deviation from satellite time).

**Features of GPS**

* High data rate
* Real time monitoring
* Accuracy

**Limitations of GPS**

* Usage cost is high
* Not suitable for all weather condition

**3.2 Selection of Processor**

While the microcontroller is a central part of any embedded system design, little consideration has gone into choice of device in the past. There are a wide range of microcontrollers currently on the market – all of them offer broadly similar features. These processors can be picked up for implementation based upon the individual requirements of the application. It is well-known that varying architectural, technological and implementation aspects of embedded microprocessors can produce widely differing performance and power specifications. There is an exhaustive list of processors used in embedded systems. Hardware characteristics of a few selected processors are given in this section.

With numerous kinds of processors with various design philosophies available at our disposal for using in our design, following considerations need to be factored during processor selection for an embedded system.

* Performance Considerations
* Power considerations
* Peripheral Set
* Operating Voltage
* Specialized Processing Units

**3.2.1 Atmel 8051**

The most commonly used set of microcontrollers belong to **8051** Family. **8051 Microcontrollers** continue to remain a preferred choice for a vast community of hobbyists and professionals. Through **8051**, the world became witness to the most revolutionary set of microcontrollers. Intel fabricated the original **8051** which is known as MCS-51. It operates on a voltage range of 2.7-5.5V.

**3.2.2 AVR**

The AVR is a modified Harvard architecture  8-bit  RISC  single chip  microcontroller which was developed by Atmel  in 1996. The AVR was one of the first microcontroller families to use on-chip flash memory for program storage, as opposed to one-time programmable ROM, EPROM, or EEPROM used by other microcontrollers at the time. AVRs are generally classified into following broad groups: tinyAVR, megaAVR, XMEGA, Application specific AVR and FPSLIC (AVR with FPGA) and 32-bit AVR. The difference between these devices lies in the available features. So for your project you should select an AVR that only includes the features that you need if you are on a strict budget

**3.2.2 ARM Cotex A7**

It has two target applications; firstly as a smaller, simpler, and more power-efficient successor to the Cortex-A8. The other use is in thebig.LITTLE architecture, combining one or more A7 cores with one or more Cortex-A15 cores into a heterogeneous systemTo do this it is fully feature-compatible with the A15.

Key features of the Cortex-A7 core are:

* Partial dual-issue, in-order microarchitecture with an 8-stage pipeline
* NEON SIMD instruction set extension
* VFPv4 Floating Point Unit
* Thumb-2 instruction set encoding
* Jazelle RCT
* Hardware virtualization
* Large Page Address Extensions (LPAE)
* Integrated level 2 Cache (0–1 MB)
* 1.9 DMIPS / MH

**3.2.3 Raspberry pi 2**

The Raspberry Pi 2 Model B is the second generation Raspberry Pi. It replaced the original [Raspberry Pi 1 Model B+](http://www.raspberrypi.org/products/model-b-plus/) in February 2015. Compared to the Raspberry Pi 1 it has:

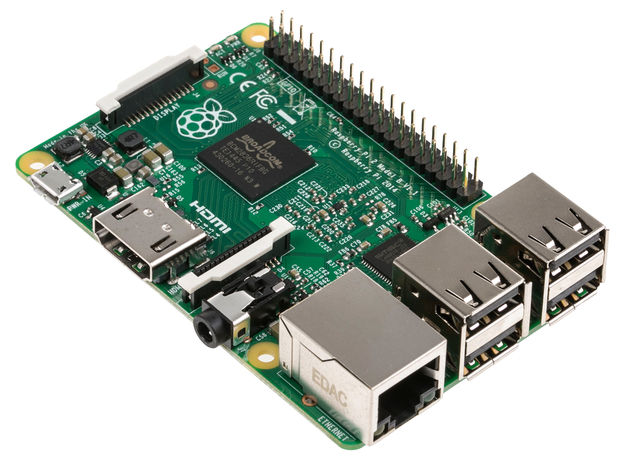
* A 900MHz quad-core ARM Cortex-A7 CPU
* 1GB RAM

Like the (Pi 1) Model B+, it also has:

* 4 USB ports
* 40 GPIO pins
* Full HDMI port
* Ethernet port
* Combined 3.5mm audio jack and composite video
* Camera interface (CSI)
* Display interface (DSI)
* Micro SD card slot
* VideoCore IV 3D graphics core

Because it has an ARMv7 processor, it can run the full range of ARM GNU/Linux distributions, including Snappy Ubuntu Core, as well as Microsoft Windows 10.

The Raspberry Pi 2 has an identical form factor to the previous (Pi 1) Model B+ and has complete compatibility with Raspberry Pi 1.



* + 1. **Selection of Software Development Tool**

Software development tool is a software package which is used to develop the software part of the system i.e. the software code. It consists of lots of supporting tools such as editor, compiler, project manager, simulator and debugger. Here for this project we need a tool which is used for programming ARM Cotex A7 controllers. The programming language used for programming the controller is Python language which is interpreted by a Python IDLE. User writes his code in python language on the editor and then interprets the code with a Shell compiler. This code is then simulated by a simulator and then it is debugged for the errors and burnt into the controller’s memory by a debugger. Few of the available tools are Python IDLE, Shell etc.All these tools are used for programming Adafruit’s raspberry pi series of microcontroller. These tools are explained in the following section.

**3.3.1 Python IDLE**

**IDLE** (**I**ntegrated **D**eve**L**opment **E**nvironmentor **I**ntegrated **D**evelopment and **L**earning **E**nvironment) is an integrated development environment for Python, which has been bundled with the default implementation of the language since 1.5.2b1. It is packaged as an optional part of the Python packaging with many Linux distributions. It is completely written in Python and the Tkinter GUI toolkit (wrapper functions for Tcl/Tk).

IDLE is intended to be a simple IDE and suitable for beginners, especially in an educational environment. To that end, it is cross-platform, and avoids feature clutter.

According to the included README, its main features are:

* Multi-window text editor with syntax highlighting, auto completion, smart indent and other.
* Python shell with syntax highlighting.
* Integrated debugger with stepping, persistent breakpoints, and call stack visibility.

IDLE has been criticized for various usability issues, including losing focus, lack of copying to clipboard feature, lack of line numbering options, and general user interface design; it has been called a "disposable" IDE, because users frequently move on to a more advanced IDE as they gain experience.

**3.5 Selection of Other Hardware Components**

This section gives the details of the communication modules i.e. GPS and Bluetooth modules,IR sensors, Ultrasonic Sensors, Magnetometer used in this system. These modules are used for communication between user and controller for giving control instructions to the smart car. The car motors responds to the sensor and the gps data.

**3.5.1 GPS**

The **Global Positioning System** (**GPS**) is a space-based navigation system that provides location and time information in all weather conditions, anywhere on or near the Earth where there is an unobstructed line of sight to four or more GPS satellites.The system provides critical capabilities to military, civil, and commercial users around the world. The United States government created the system, maintains it, and makes it freely accessible to anyone with a GPS receiver.

The GPS concept is based on time and the known position of specialized satellites. The satellites carry very stable atomic clocks that are synchronized to each other and to ground clocks. Any drift from true time maintained on the ground is corrected daily. Likewise, the satellite locations are known with great precision. GPS receivers have clocks as well; however, they are not synchronized with true time, and are less stable. GPS satellites continuously transmit their current time and position. A GPS receiver monitors multiple satellites and solves equations to determine the precise position of the receiver and its deviation from true time. At a minimum, four satellites must be in view of the receiver for it to compute four unknown quantities (three position coordinates and clock deviation from satellite time).

Each GPS satellite continually broadcasts a signal (carrier wave with modulation) that includes:

* A pseudorandom code (sequence of ones and zeros) that is known to the receiver. By time-aligning a receiver-generated version and the receiver-measured version of the code, the time of arrival (TOA) of a defined point in the code sequence, called an epoch, can be found in the receiver clock time scale
* A message that includes the time of transmission (TOT) of the code epoch (in GPS system time scale) and the satellite position at that time

*Conceptually*, the receiver measures the TOAs (according to its own clock) of four satellite signals. From the TOAs and the TOTs, the receiver forms four time of flight (TOF) values, which are (given the speed of light) approximately equivalent to receiver-satellite range differences. The receiver then computes its three-dimensional position and clock deviation from the four TOFs.

In practice the receiver position (in three dimensional Cartesian coordinates with origin at the Earth's center) and the offset of the receiver clock relative to the GPS time are computed simultaneously, using the navigation equations to process the TOFs.

The receiver's Earth-centered solution location is usually converted to latitude, longitude and height relative to an ellipsoidal Earth model. The height may then be further converted to height relative the geoid (e.g., EGM96) (essentially, mean sea level). These coordinates may be displayed, e.g. on a moving map display and/or recorded and/or used by some other system (e.g., a vehicle guidance system)

**GPS MODULE :**



**Features:**

* Build on high performance SiRF Star III chipset
* Average Cold Start time and under 45 seconds
* Low Power Consumption
* 20 channels “All-in-View” tracking
* 200,000 effective correlators for fast TTFF and high sensitivity acquisitions
* Integrated ARM7TDMI CPU with software engineering services and available for embedded customer defined applications.
* On Chip 1Mb SRAM
* Dual TTL level serial ports with one for GPS receiver command message interface, another one for RTCM-104 DGPS input.
* Support Standard NMEA-0183 and SiRF Binary protocol
* Support Accurate 1PPS Output Signal Aligned with GPS Timing
* Multi-Path Mitigation Hardware
* On-Board RTCM SC-104 DGPS and WAAS / EGNOS Demodulator
* Built-in a lithium battery make GPS fast positioning

**Specifications:**

* Compact Board Size 1" x 1" x 0.27"(25.4 x 25.4 x 7 mm) for easy integration into hand-held device
* Reacquisition Time 0.1 seconds

**3.3.2 Magnetometer**

**Magnetometers** are measurement instruments used for two general purposes: to measure the magnetization of a magnetic material like a ferromagnet, or to measure the strength and, in some cases, the direction of the magnetic field at a point in space.

### Types of magnetometer

There are two basic types of magnetometer measurement. *Vector magnetometers* measure the vector components of a magnetic field. *Total field magnetometers* or *scalar magnetometers* measure the magnitude of the vector magnetic field. Magnetometers used to study the Earth's magnetic field may express the vector components of the field in terms of *declination* (the angle between the horizontal component of the field vector and magnetic north) and the *inclination* (the angle between the field vector and the horizontal surface).

*Absolute magnetometers* measure the absolute magnitude or vector magnetic field, using an internal calibration or known physical constants of the magnetic sensor. *Relative magnetometers* measure magnitude or vector magnetic field relative to a fixed but uncalibrated baseline. Also called *variometers*, relative magnetometers are used to measure variations in magnetic field.

Magnetometers may also be classified by their situation or intended use. *Stationary magnetometers* are installed to a fixed position and measurements are taken while the magnetometer is stationary. *Portable* or *mobile magnetometers* are meant to be used while in motion and may be manually carried or transported in a moving vehicle.*Laboratory magnetometers* are used to measure the magnetic field of materials placed within them and are typically stationary. *Survey magnetometers* are used to measure magnetic fields in geomagnetic surveys; they may be fixed base stations, as in the INTERMAGNET network, or mobile magnetometers used to scan a geographic region.

HMC5883L (Digital Compass)



The Honeywell HMC5883L is a surface-mount, multi-chip module designed for low-field magnetic sensing with a digital interface for applications such as lowcost compassing and magnetometry. The HMC5883L includes our state-of-theart, high-resolution HMC118X series magneto-resistive sensors plus an ASIC containing amplification, automatic degaussing strap drivers, offset cancellation, and a 12-bit ADC that enables 1° to 2° compass heading accuracy. The I 2C serial bus allows for easy interface. The HMC5883L is a 3.0x3.0x0.9mm surface mount 16-pin leadless chip carrier (LCC). Applications for the HMC5883L include Mobile Phones, Netbooks, Consumer Electronics, Auto Navigation Systems, and Personal Navigation Devices. The HMC5883L utilizes Honeywell’s Anisotropic Magnetoresistive (AMR) technology that provides advantages over other magnetic sensor technologies. These anisotropic, directional sensors feature precision in-axis sensitivity and linearity. These sensors’ solid-state construction with very low cross-axis sensitivity is designed to measure both the direction and the magnitude of Earth’s magnetic fields, from milli-gauss to 8 gauss. Honeywell’s Magnetic Sensors are among the most sensitive and reliable low-field sensors in the industry.

Communication with the HMC5883L is simple and all done through an I2C interface.

Comes in a low-height, LCC surface mount package.

**Features:**

* Simple I2C interface
* 2.16-3.6VDC supply range
* Low current draw
* 5 milli-gauss resolution

**Dimensions:** 3.0x3.0x0.9mm

3.3.3 Ultrasonic Sensor



A special sonic transducer is used for the ultrasonic proximity sensors, which allows for alternate transmission and reception of sound waves. The sonic waves emitted by the transducer are reflected by an object and received back in the transducer. After having emitted the sound waves, the ultrasonic sensor will switch to receive mode. The time elapsed between emitting and receiving is proportional to the distance of the object from the sensor.

Ultrasonic sensors can detect movement of targets and measure the distance to them in many automated factories and process plants. Sensors can have an on or off digital output for detecting the movement of objects, or an analog output proportional to distance. They can sense the edge of material as part of a web guiding system.

Ultrasonic sensors are widely used in cars as parking sensors to aid the driver in reversing into parking spaces. They are being tested for a number of other automotive uses including ultrasonic people detection and assisting in autonomous UAV navigation.

Because ultrasonic sensors use sound rather than light for detection, they work in applications where photoelectric sensors may not. Ultrasonics are a great solution for clear object detection, clear label detection and for liquid level measurement, applications that photoelectrics struggle with because of target translucence. As well, target color and/or reflectivity do not affect ultrasonic sensors, which can operate reliably in high-glare environments.

Passive ultrasonic sensors may be used to detect high-pressure gas or liquid leaks, or other hazardous conditions that generate ultrasonic sound. In these devices, audio from the transducer (microphone) is converted down to human hearing range.

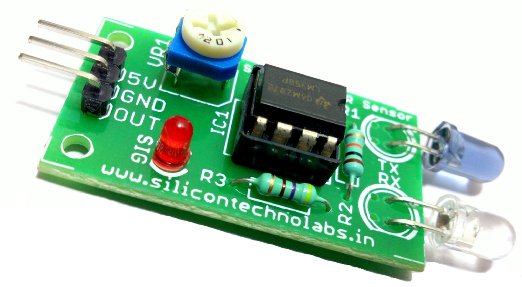
TARGET DETECTION

Sonic waves are best reflected from hard surfaces. Targets may be solids, liquids, granules or powders. In general, ultrasonic sensors are deployed for object detection where optical principles would lack reliability

DIGITAL OUTPUT

Sensing is only possible within the detection area. The required sensing range can be adjusted with the sensor's potentiometer or by electronic Teach-in (Teach-in button or remote Teach-in). If an object is detected within the set area, the output will change state which is visualized by the integrated LED

* + 1. IR Sensor



IR Sensors work by using a specific light sensor to detect a select light wavelength in the Infra-Red (IR) spectrum. By using an LED which produces light at the same wavelength as what the sensor is looking for, you can look at the intensity of the received light. When an object is close to the sensor, the light from the LED bounces off the object and into the light sensor. This results in a large jump in the intensity, which we already know can be detected using a threshold.

An **infrared sensor** is an electronic device, that emits in order to sense some aspects of the surroundings. An **IR sensor** can measure the heat of an object as well as detects the motion.These types of **sensors** measures only **infrared** radiation, rather than emitting it that is called as a passive **IR sensor**

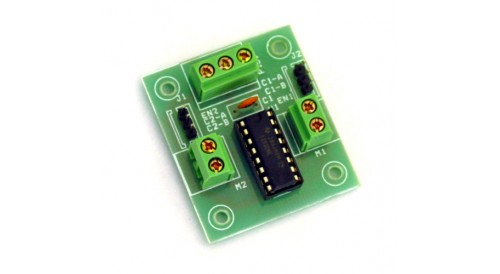
An IR sensor consists of an emitter, detector and associated circuitry. The circuit required to make an IR sensor consists of two parts; the emitter circuit and the receiver circuit.

The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode which is sensitive to IR light of the same wavelength as that emitted by the IR LED. When IR light falls on the photodiode, its resistance and correspondingly, its output voltage, change in proportion to the magnitude of the IR light received. This is the underlying principle of working of the IR sensor.

## Working Mechanism

An IR sensor is basically a device which consists of a pair of an IR LED and a photodiode which are collectively called a photo-coupler or an opto-coupler. The IR LED emits IR radiation, reception and/or intensity of reception of which by the photodiode dictates the output of the sensor. Now, there are so many ways by which the radiation may or may not be able to reach the photodiode.

* + 1. **L293D Motor Driver**



L293D is a dual H-bridge motor driver integrated circuit (IC). Motor drivers act as current amplifiers since they take a low-current control signal and provide a higher-current signal. This higher current signal is used to drive the motors.

L293D contains two inbuilt H-bridge driver circuits. In its common mode of operation, two DC motors can be driven simultaneously, both in forward and reverse direction. The motor operations of two motors can be controlled by input logic at pins 2 & 7 and 10 & 15. Input logic 00 or 11 will stop the corresponding motor. Logic 01 and 10 will rotate it in clockwise and anticlockwise directions, respectively.

Enable pins 1 and 9 (corresponding to the two motors) must be high for motors to start operating. When an enable input is high, the associated driver gets enabled. As a result, the outputs become active and work in phase with their inputs. Similarly, when the enable input is low, that driver is disabled, and their outputs are off and in the high-impedance state

**3.6 Summary**

First section explains different technologies used for this system and why GPS and Bluetooth are collectively selected. The Comparison between different microcontrollers is done and Raspberry Pi 2 is selected amongst them due to its inbuilt features. Then in next section the software tool Python IDE used for programming the controller is selected. The details of the selected components are also given in the next section. The design and implementation of the system is discussed in the following chapters.

**CHAPTER 4 : DESIGN APPROACH**

This chapter describes the design approach and conceptual design of a low cost SMART CAR. The system’s low cost needs are catered by employing a various module which are operated with the controlled devices and by using Google API for path detection along with GPS.

**4.1 Introduction to System Design**

The embedded systems consist of many modules, which are comprised of software components, hardware components and interfaces. All these modules can be independently modeled as complex systems. In order to achieve a correct implementation of a project all of the independent designs must work in synergy. Therefore, application of system design principles to the design of embedded system can dramatically streamline the design work and avoid future problems involved with integrating the modules to constitute a larger system. The embedded system is a system in which the processing unit is actually embedded between its peripherals and the system is designed to perform some predefined tasks. Being dedicated to certain tasks, the embedded system provides a very efficient solution compared to their general purpose counterparts. When designing complex systems, it is beneficial to approach the design via an architecture, which is structured as an integration of sub-systems. In this approach the designers identify the system requirements for subsystems, which are based on overall system requirements. The subsystems are designed independently and then interfaced to achieve the completed system architecture. This approach simplifies the procedures related to testing, debugging and integration of the subsystems, which are required to ensure proper design and working of the whole system and also divides the functionality to verify the working of the subsystems independently.

**4.2 Design Methodology**

A good design methodology can help the system design process in many ways. It can help to verify the system for functionality and for errors and helps realize a solution, which achieves all the goals of the system including manufacturing cost, performance and power consumption.

There are two approaches to the system design process. One is top-down design and other is bottom-up design. In top-down approach the design procedures are initiated from requirements for system integration. This approach involves arriving at the right solution after considering all the possible alternatives. The alternative is the bottom-up in which we start from the components to build a system i.e. the reverse of the top-down approach. The bottom-up design is needed because one does not know how the later stages of the project will turn out to be. In this remote monitoring and control system with automatic light controller we have considered a top-down design approach. The detailed description of the steps is given in the following sections.

**4.3 Requirements and Specifications**

The requirements are objective descriptions of the system, which includes functional as well as nonfunctional requirements. The requirements are the customer’s expectations.The requirements put monetary and timing constraints on the design, which will have to be considered along with the technical specifications. The designers need to incorporate these requirements and realize a system, which can perform the expected tasks. The system specifications are more focused on system implementation. They offer the designer a role map for the design of the system. The specifications have to be written carefully to ensure that they meet requirements. The system design has to fulfill the system requirements. The planning stage is the most important stage in the design cycle since it considers all the available resources and their efficiencies.

For this project many of available hardware and software components have been studied and compared and the optimal ones in terms of cost and specifications which best suits our requirements have been selected. Chapter 2 gives all the details of the existing technologies of driverless car.

We have selected RASPBERRY PI 2 microcontroller, GPS modem,HMC5883L Magnetometer,L293D motor driver,100 rpm dc brushed motors,HC-SR04 Ultasonic sensor and IR sensors as hardware components and Python 2.7 software programming tool is used to write the software code for our project. The selection of these hardware and software components along with their technical specifications is already discussed in chapter 3 of this report.

**4.4 Architecture of the system**

The system architecture can be served in two parts- Hardware architecture and Software architecture.

**4.4.1 Hardware Architecture**

Hardware of the system comprises of six main units-

* Microcontroller which is the heart of the system
* GPS module
* Magnetometer
* Motors and L293D Motor Driver
* Sensor part which consists of one Ultrasonic sensor and two Infra-Red sensors
* Power supply unit

RASPBERRY PI 2

The design adopts RASPBERRY PI 2 microcontroller from Adafruit. PI 2 models feature a Broadcom system on a chip (SOC), which includes an ARM compatible CPU and an on chip graphics processing unit GPU (a VideoCore IV). CPU speed ranges from 700 MHz to 1.2 GHz for the Pi 3 and on board memory range from 256 MB to 1 GB RAM. Secure Digital SD cards are used to store the operating system and program memory in either the SDHC or MicroSDHC sizes. Most boards have between one and four USB slots, HDMI and composite video output, and a 3.5 mm phono jack for audio. Lower level output is provided by a number of GPIO pins which support common protocols like I2C. Some models have an 8P8C Ethernet port and the Pi 3 has on board WiFi 802.11n and Bluetooth.

GPS MODULE

By using GPS Module, we get the present location of our smart car. By using the present location we will be guide our car from the present location to the desired location by using Goggle Api.

MAGNETOMETER

By using Magnetometer, our smart car will be able to orient it’s direction of movement according to the commands being fed by the Google Api geocode.

SENSORS( ULTRASONIC AND IR)

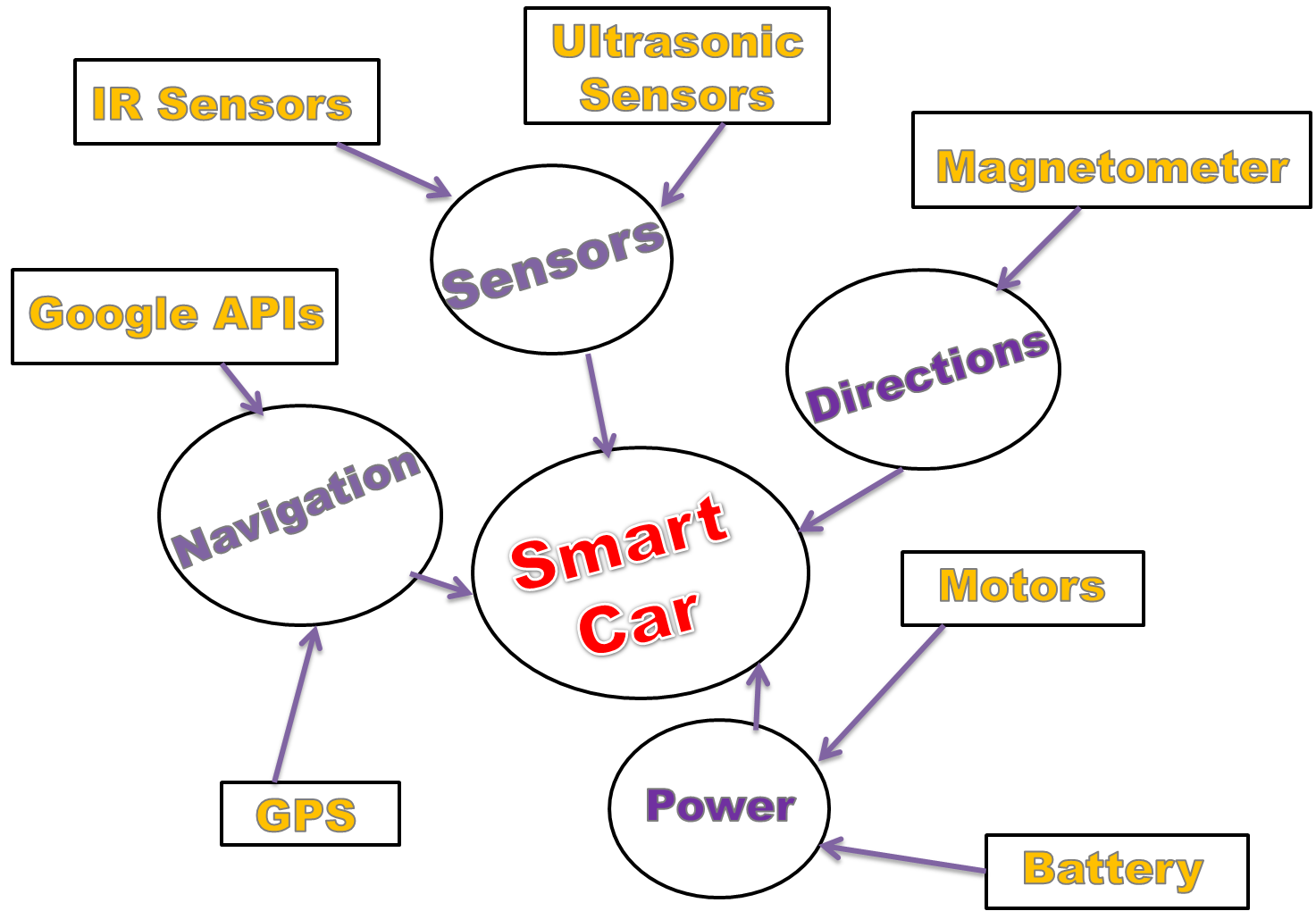
By using ultrasonic and IR sensor our smart car will be able to detect obstacles in front of it and thus will be able to avoid collisions.

MOTORS AND L293D MOTOR DRIVER

By using L293D motor driver our car is controlling the dc brushed motor .The motors are making the car move in all the four directions.

POWER SUPPLY

We are providing power to the raspberry pi and the motors by the usage of LIPO batteries.



The block diagram of system Hardware architecture is given in the fig above.

**4.4.2 Software Architecture**

***GOOGLE API ( GEOCODE )***

**Google APIs** is a set of application programming interfaces ([APIs](https://en.wikipedia.org/wiki/API)) developed by [Google](https://en.wikipedia.org/wiki/Google) which allow communication with [Google Services](https://en.wikipedia.org/wiki/Google_Services) and their integration to other services. Examples of these include Search, Gmail, Google Maps. Third-party apps can use these APIs to take advantage of or extend the functionality of the existing services.

The APIs provide functionality like analytics, machine learning as a service (the Prediction API) or access to user data (when permission to read the data is given). Another important example is an embedded Google map on a website, which can be achieved using the Static maps API, Places API or Google Earth API.

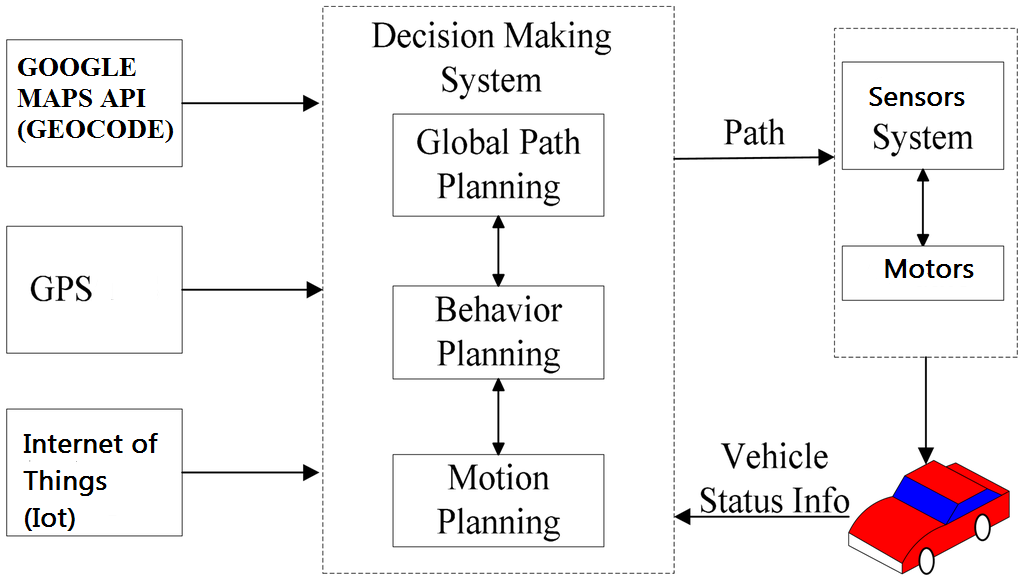
In this project , we are using the Google API to give instructions to the car to follow the prescribed path as per the instructions given by the Google API geocode which fetches data from the Google Maps.

**GOOGLE MAPS GEOCODING API**

**Geocoding** is the process of converting addresses (like "1600 Amphitheatre Parkway, Mountain View, CA") into geographic coordinates (like latitude 37.423021 and longitude -122.083739), which you can use to place markers on a map, or position the map.

The Google Maps Geocoding API provides a direct way to access these services via an HTTP request. The following example uses the Geocoding service through the Google Maps JavaScript API to demonstrate the basic functionality.

The software architecture of the system is given in fig below.



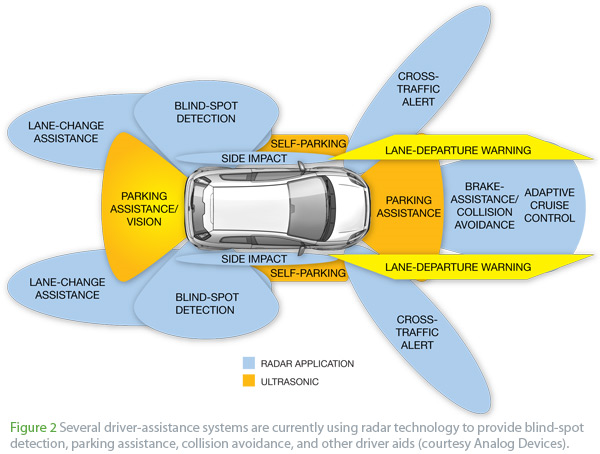
This diagram shows the flow of data and instructions between the entities of the system. The laptop or PC and mobile device with Internet connections are the instruction giving units which sends the state instructions to the pi and the Microcontroller including GPS ,Sensors and Motors are the controlling units which receives these instructions and controls the movement of the car accordingly.

The user switches on the car by pressing a push button. As soon as the button is pressed the car comes into the active state. At this moment, the user is asked to feed the desired destition where he / she wishes to go. After the address of the place is entered. The GPS immediately tracks the present location of the car and the Google Map Geocoding API sends the instructions to the microcontroller. As per the directions given by the Geocoding API, the car orients itself in that particular direction using the magnetometer. After this the motors starts the car motion. With advancement in the position of the car the sensor are simultaneously detection the obstacles in front of the car. As soon as an obstacle is detected the car advances in the direction of no obstacle. Also with every advancement, the gps is tracking the latitude and longitude changes of the smart car and the geocoding API is fetching the html instructions for further motion of the car.

In this way the car reaches it’s final destination.

**4.5 Layered Structure of the System**

The layered structure of the system is given in fig 4.3. This diagram shows the flow of data and instructions between the layers of the system. It has four layers. First layer is instruction given by the user(switch on the car ,feeding destination address etc). Second is the communication link which is a GPS module and GOOGLE MAPS GEOCODING. Third is the microcontroller unit which acts as central control station and receives all the control instructions and also controls all other components based on this data. Last layer is the movement of motors and sensing by sensors.



**4.5 Summary**

In this chapter the design process and approaches of an embedded system are discussed. Section 4.2 gives the details about the requirements and specifications for the project. Section 4.3 explains the hardware and software architecture of the project along with bock diagrams. Section 4.4 explains the layered structure of the project. The remaining two building blocks of the design process i.e. design of hardware and software components, their integration and testing is explained in the following chapter

**CHAPTER 6 : Result and Analysis**

A Smart Car (self driving car) prototype is developed and experiments have been conducted. This system is designed with the objective of creating a driverless car kit through which we can make the existing cars work autonomously. The car will be using the concept of internet of things (Iot) by which it will interact with the google map geocoding API to get path from the current location(which will be determined by the GPS) to the final destination given by the user.The car will follow the path prescribed by the google maps and would be able to avoid obstacles in it’s path.

****

**6.1 Google Maps Geocoding API**

Fig 1 shows the output of the geocode . The user has to feed the final destination address and the geocode will return the path to be followed by the car.

**Fig 1 – Output Geocode**

.

**6.2 Obstacle Avoidance and Motor control**

Fig .2 shows the output of the obstacle avoidance and motor control code. The ultrasonic and IR sensors are used for obstacle detection and as per the sensor outputs the motion of the motors are controlled by coding.

FIG 2 – Obstacle Avoidance Output.

**6.3 GPS**

Fig 3 shows the output being fetched by the GPS. The gps code is returning the latitude and longitude of the car’s present location.

**Fig 3 – GPS Output**

**6.4 Summary**

In this chapter, we have noted the output of the various code / components which are being used in the project.