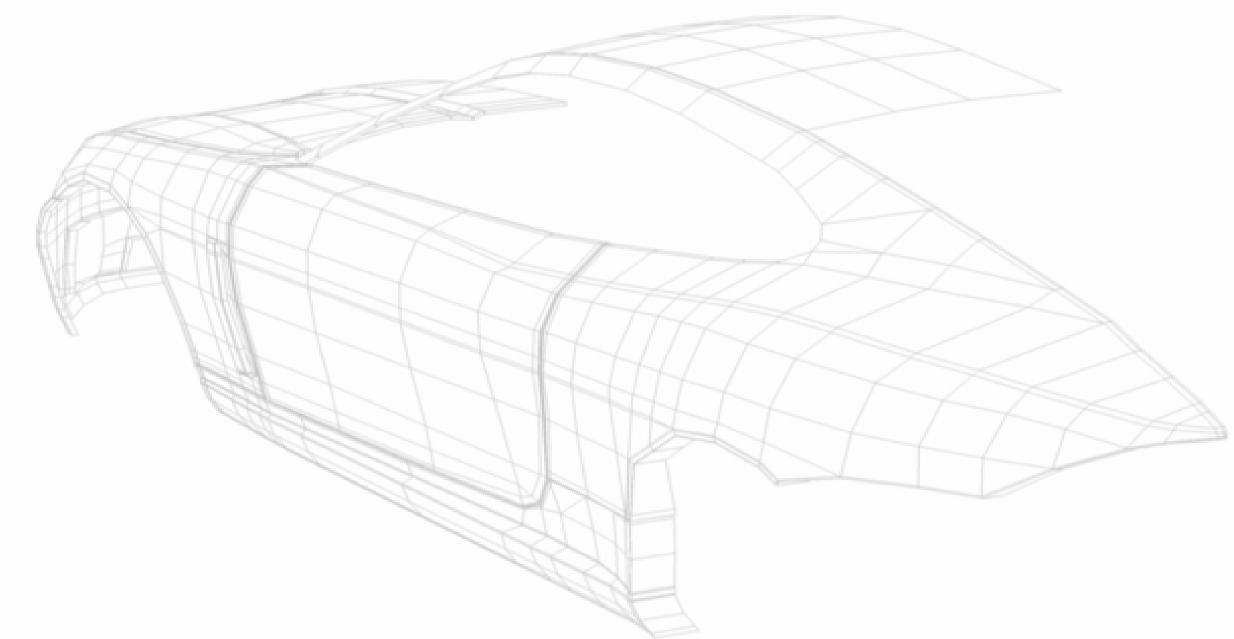
2nd Generation Car Security System (2GCSS)

Embedded Car Security System



A Thesis Report

Submitted to:

Department of Electrical and Electronic Engineering

Submitted by:

Md. Mashiur Rahman	09221090
Md. Kudrat Ullah Nahid	09221063

Thesis Supervisor

Dr. Khalilur Rhaman
Assistant Professor

Department of Computer Science and Engineering

BRAC University
Dhaka, Bangladesh

Declaration

We declare that the thesis report is based entirely upon the research conducted by the members of the group under the guidance and supervision of our thesis supervisor. The report focuses on the theoretical aspects of developing an embedded car Security System and also illustrates the mechanism of how we have implemented the theoretical aspects of the car security system and it illustrates certain necessary steps required to implement the theories in order to develop an actual prototype of the "2nd Generation Car Security System" in reality. All the reference documents have been stated in the reference section and all the extracted information and interpretation have been done through our own effort

Date: 15th December 2012

Md. Mashiur Rahman

Md. Kudrat Ullah Nahid

Dr. Khalilur Rhaman
Assistant Professor
Department of Computer Science and Engineering
BRAC University

Table of Contents

Abstract:

In this era of technology security and maintenance has become a big question in all over the world. Wherever a developed technology is used, the question of its security and maintenance also arises. If any precious thing is lost the first thing comes in our mind is where it's right now. The solution can be given by a Global positioning system (GPS).Another thing which comes in our mind is the maintenance of that system. We can use sonar, alarms, and switches in various ways so that we can protect our valuable system. The world has so far experienced many kinds of modules for security and maintenance. As such we have implemented modern technologies in this security system and added all the features in a module which is more reliable and flexible while use. We have implemented the following functions in our thesis.

- Real time Vehicle tracking using GPS.
- Engine On/off via SMS using the GSM technology
- Robust Jumper alert and warning alarm
- Blind point detection using SONAR.
- Parking Assistant
- Passenger counter and data log

Acknowledgements

We would like to take this opportunity to express our heartiest gratitude to our thesis supervisor Dr. Khalilur Rhaman who has always provided every possible help and support over this time that we have utilized to develop this project of ours We have tried to develop a project that can easily be used and can be related to by general people and we had a vision in our mind to implement such a project and our batch mates have also helped us to work as a source of inspiration through ideas about components wherever we struggled.

Introduction:

In this era of technology security has become a big question in all over the world. And having a security system for automobile is almost default as vehicles in considered being a valuable possession. With the development in technology and enhanced security features, the car thieves has also become technologically upgraded enough to disable those features. Hence keeping this in mind we have integrated the existing different security system units in one single unit with some improvisations so it becomes impossible for the thieves to disable the system. This thesis project proposes the incorporation of different car security mechanisms. Microcontroller PIC18F2550, sonar module, Gyro Sensor, Transmitter Receiver (RX TX), and SIMCOM586 (GPS and GSM module) have been used as the hardware part. Software part is done using C# (.Net framework) is the programming language. The GSM module at the user end is connected to the user's PC though a serial port communication. Google API is used as the user interface to view the Google map and wireless communication is used for the auto locking system of the car doors. The interconnection of all the features provides flexibility to the user and is more reliable ensuring higher safety and protection against car theft. Moreover this system offers a cost effective solution to users than buying the different security devices individually. Saving the car from accident is also considered as a part of security of the passengers and the car itself. Thus we have incorporated the technique "blind point detection" with the rest of the security features which subsides the risk of accident mainly caused during switching lanes or overtaking. A number of literatures [2][3][4][5][6][7][8] [9] and [10] are reviewed to develop each part of this security device.

Existing System

A few of the leading manufacturing countries of car security system includes China, USA, Italy and India. Some of the highly demanded car security systems are listed below:

- **Alarm Systems** – Practically, noise influences on thieves quickly and effectively. It also acts as a messenger to let the people nearby know about the illegal entry in a car. That is why there is wide range of alarm system sensors: impact, motion, door-open, etc. Most alarm systems ensure that their alarm noise is loud enough to grab people's attention.
- **Remote Keyless Entry System** – This is a system which locks and unlocks cars remotely. This is achieved by sending pulses on a particular frequency. Advanced mechanism uses encrypted pulse transmissions that ensure higher safety.

- **Steering Wheel Locks** – For amateur and unprofessional thieves, this system is very effective. This is achieved since the steering gets firmly fitted that allows theft to be practically impossible.
- **Immobilizers** – This special type of security cuts off the fuel or the ignition to stop the car from being started. However, even with this method enabled, a thief can break into a car, but stealing gets quite impossible as long as the immobilizer is hidden from the thief; in other words, as long as the immobilizer is not disabled by the thief.
- **VIN Etching** – Vehicle Identification Number (VIN) is etched onto many parts of a car. This specifically serves the security purpose of prevention of stealing cars with an intention to sell its parts. This means, even if a vehicle is stolen, the presence of VIN on various important parts of the car will makes its identification easy.
- **Central Locking system** – The main idea of this security is pretty simple. Through this, locking or unlocking of only the driver's door will act as a similar action to the other doors of the car.
- **GPS Vehicle Tracking system -** With the GPS tracking device installed in the car, the owner is certain that his car won't ride away too far a distance without it's knowledge. The minute it is suspected that the car has been stolen, the owner can inform the authority and they will immediately tap into the GPS system so that they can instantly check where the car is at that very instant. Skilled car thieves can override the car's alarm system and get away with the car, but with the GPS tracking system active in the car, hiding is never an option.

Unfortunately, even with so many advanced technologies to guard our cars for safety, car theft has not been stopped. Rather in some countries, car theft has increased than the past years. A possible answer to this mystery might be, with the advancement of car security, car theft has upgraded itself as well with the use of higher advanced technology and tactics

Statistical Analysis

United Nations Office on drugs and crimes give a report for Car theft in Bangladesh. Crimes recorded in criminal (police) statistics (2005-2006):

2005	2006	Rate 2005	Rate 2006
1044	1410	0.68	1.90

Given below are some simple statistical analysis of car thefts for Malaysia and UK:

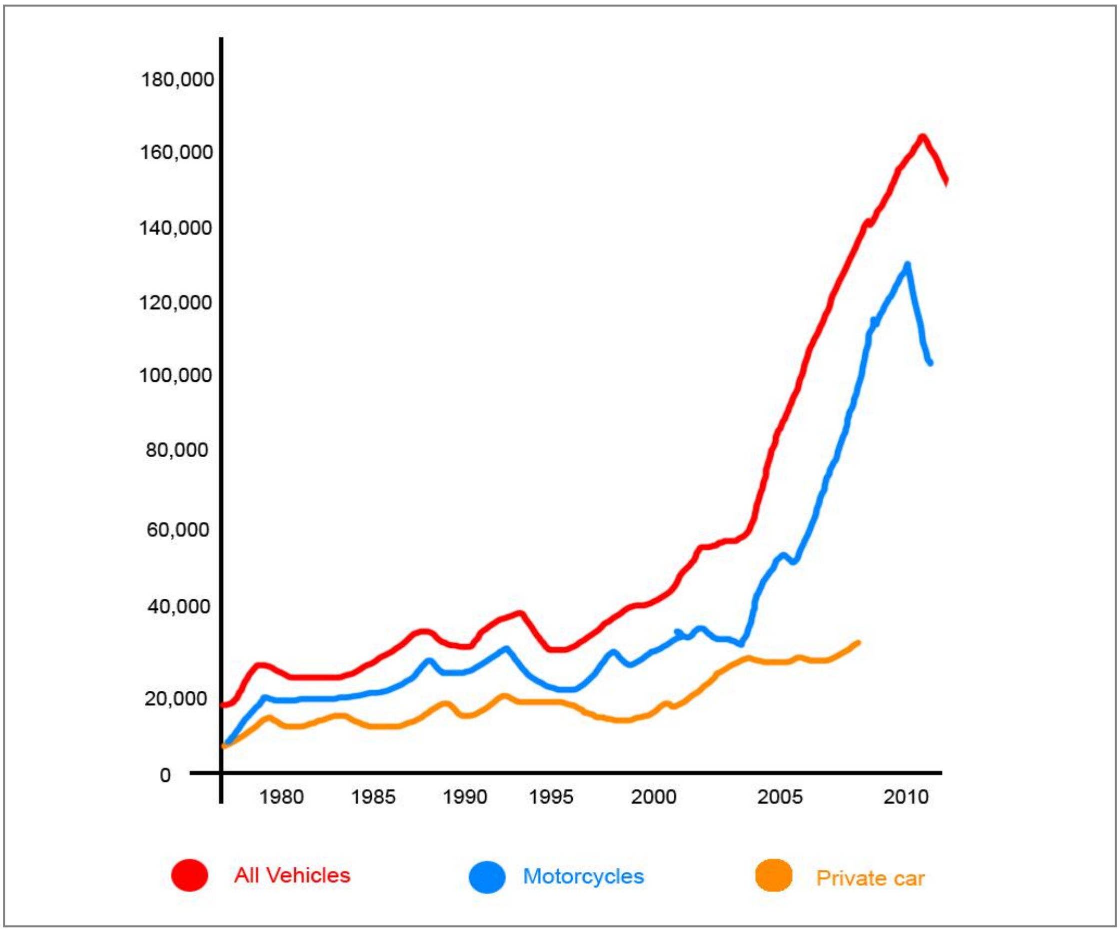


Fig: 1 car theft statistics in Malaysia from 1980 to 2010

As per this graph, in 1980, around 9000 cars were stolen. From 1990 to 1997, car theft even though decreased slightly but drastically increased in between 2000 and 2005. Still not at 2010 car theft has increased significantly compared to that in 1980. The graph says around 21000 cars have been stolen in 2010. This gives a percentage increase of around 133% compared to 1980.

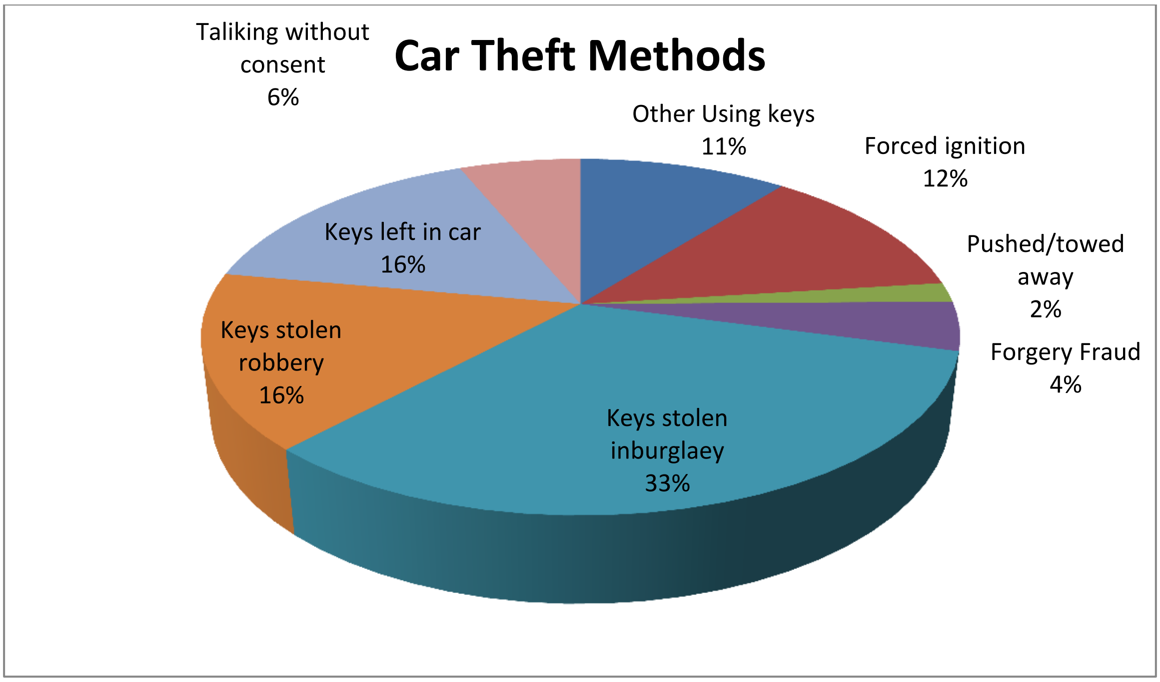


Fig: 2 car theft methods at UK in 2009

Completed Features

1. GPS Vehicle Tracking:

The GPS tracking system is the most common tool in a security device. At its simplest definition, the GPS device or tracker is the component in charge of receiving the information about the location of the vehicle and providing this data to the GPS tracking application software installed in the owner's computer through transmission system. There are two types of existing tracking system, passive tracking, real time tracking on either satellite based or cellular network based and tracking through SMS using GSM module. We have incorporated all this different techniques so that the best of all techniques are active into one single system. Our developed system is user friendly because this tracking option is totally depend on the user. This proves to be more efficient, user friendly and cost effective.

2. GSM engine ON/OFF system:

SMS used on modern handsets are originated from radio technology in radio memo pages using standardized phone protocols. Later defined as a part of the Global system for mobile communication (GSM), it is used as a means of sending messages to or from mobile handsets or GSM modules. Using this concept we have configured the engine ON/OFF system through SMS. The user can easily turn the engine on or off by sending a SMS code. The ON/OFF technique is directed by a simple command which is then imported to the microcontroller on the GSM module. From the microcontroller the output would be directed to the wire that goes to the last slot of the ignition switch. A relay is connected to this wire.

3. Blind Spot/Point Detection:

A blind spot in a vehicle is an area around the vehicle that cannot be directly observed by the driver while at the controls and also cannot be detected using side mirror. So if any car wants to overtake then it causes a serious accident. In transports, the driver visibility is the maximum distance at which the driver of a vehicle can see and identify prominent objects around the vehicle. Good driver visibility is essential to safe road traffic.

When a driver drives a car he can only see a car whose position is more than 40 degree horizontally from the driving car. To develop this system we used a SONAR module. Our system is based on microcontroller and we have used *PIC18F4550*. The sonar detects the moving vehicle either behind or beside it and alerts the driver to reduce the speed or to halt which reduces the risk of accidents during overtaking.

4. ***Parking Assistant:***

parking assistant is a system which will help the driver to park a car without any hazard. Most of the time we see that parking a car between two cars is very risky. Toyota and Lexus already give this opportunity to the customer who will buy those cars but this cars and device is very costly so that it is not available for all type of customers. We designed our System with SONAR (Ultrasonic) and the mechanism of our system is almost same as blind point detection system.

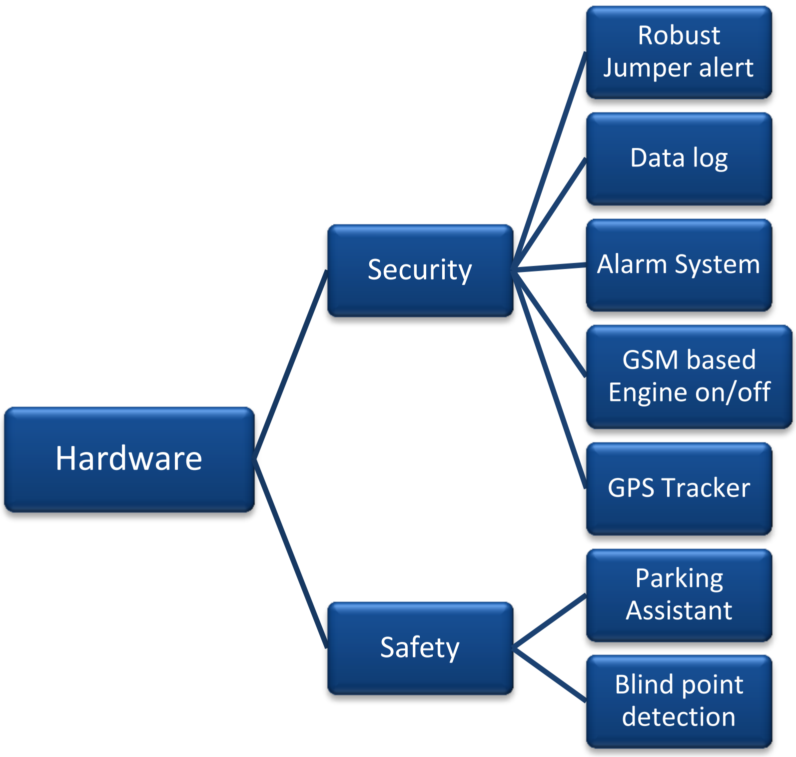
5. Robust Jumper Alert and Warning Alarm:

Whenever the car experiences a heavy shake or jump from its position at rest, this system will alert the owner that someone is trying to mess with car. For this mechanism a very small electronic device called “Gyroscope” which is sensitive to changes in gravitational acceleration ‘g’ is used. The initial position is considered when the car is at rest. With the initial position the gyroscope calculates ‘g’ and generates a corresponding voltage and sets this voltage value as the reference voltage. Due to a heavy shake when a voltage change of approximately +/- 0.5V from the reference voltage is detected the alert would be activated. With a shake the gyroscope generates various voltage readings with the respective changes in gravitational acceleration ‘g’.Thus the variation of voltage from the reference value will notify the owner with the alarm being turned on and also a SMS will be sent to the owners mobile.

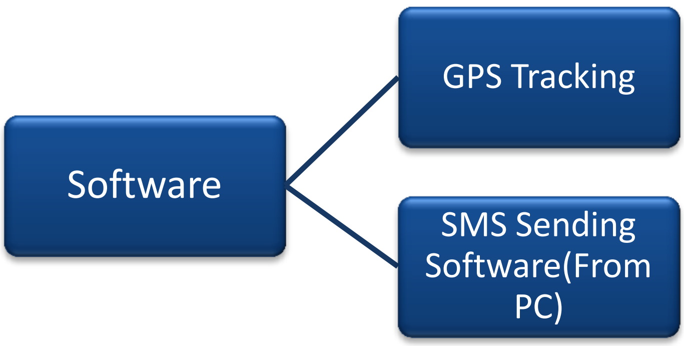
Passenger counter and data Log:

Passenger counter is very unique idea. This System will count the passenger in every time engine start and save the data into EEPROM of microcontroller. If the owner of the car wants to know that how many passengers are now in vehicle than just a small SMS is enough (Push Pop system).

System Architecture



Hardware Block Diagram



Software Block Diagram

Fig: 3 System Architecture

Previous Plan

Over the past few decades, increasing levels of income and plummeting car prices, as seen vehicle sale soar. Unfortunately the widespread popularity of cars has also increased the probability of car lost. Most of us are familiar with the normal car security system like whoops, beeps of the modern day cars. These systems are not that much effective and those which are effective are very costly. Following this scenario we decided to make a combined car security system which will be very much effective with a very low cost and we named it as second generation car security system (2GCSS).

In the first semester of our thesis we mainly focused on the GPS part and our system was successful to detect our location. Our system was using real time data from the satellite and displayed at our web based platform. We used C # to make this platform. We faced some problem due to GPS antenna but finally we overcame.

Second semester was very effective semester for our thesis. In this semester we mainly focused on Blind point detection and GSM based Engine on off system. In this system we were able to switch on/off engine through SMS. Blind point detection is another interesting part of our thesis. In general a driver can see a car behind him using rear view mirror within a particular distance. If the distance is increased he can't see the car and while he turn left or right there happens an accident. To prevent this problem we used sonar .we implemented sonar in our system and it can detect cars within 150 cm. It produces ultrasonic sound in 40 degree angle. So if any car comes in this range, it notifies the driver.

In this semester we have implemented the shake alarm system in our system. We used gyroscope in this process. The system turns on the GSM system and send message to the owner if the car faces any massive shake. And also we develop a on board passenger counter and data log. And successfully complete our full security system.

Project Progress

GPS Tracking System:

The GPS tracking system is the most common tool in a security device. At its simplest definition, the GPS device or tracker is the component in charge of receiving the information about the location of the vehicle and providing this data to the GPS tracking application software installed in the owner's computer through transmission system.

The existing tracking systems have two modes of tracking, passive tracking and real time tracking. In real time tracking there are two types, one is getting the location from satellite network and another is getting the location based on cellular network. The former system provides a minimal margin of error most of the times. This tracking mechanism is able to communicate in areas where cellular service is unavailable. Its disadvantage is that the GPS antenna has to have a clear view to the sky. For instance, if the vehicle gets into a garage, there will be no GPS location available. Whereas the latter has a bigger margin of error but has some additional benefits to it. The cellular network based tracking is cost effective and allow for the most data to be sent for the best price. As the maximum parts of the earth are now covered by cellular networks, this tracking technique is a great option. Its advantage is that it will perfectly work within the buildings making it very useful unlike satellite based tracking at such situation. Another additional benefit with cellular based tracking is that unlike satellite based tracking system they work as a two way tracking device. This means a device deployed remotely can be sent a new set of instructions to the GPS device on the vehicle to adjust the way it operates and communicates. Henceforth we designed to work with both location mechanisms in a combined form so that one's advantage can compensate the flaws of other.

Fig: 4 Satellites in space.

We included the passive tracking mechanism in our system though it is not effective like the other tracking mechanism for logging purpose. The GPS records location at regular intervals and store in its memory. The memory is then retrieved to download the log by the user for viewing. It helps the owner to keep track of locations covered by the car. It does not have any post cost such as recurring cost since monthly monitoring charges are not needed.

Mechanism:

We have used SIMCOM548C module. The compact of this module made it easy to integrate GPS/GPRS and GSM as an all in one solution and saved us significantly both time and cost for the integration of additional hardware component. GPS device (SIMCOM548C) is installed on the vehicle. The module has a total of 60 pins. The main pins of the hardware that we have used includes the power key, status key, Vcc(positive voltage supply),GPS_TXA(transmission pin) and ground. The status key shows the current condition of the communication. The power key when connected to the ground, the receiver's entire system is by default set on it. When powered on, the tracking device communicates via a cellular or satellite network and the GPS-TXA receive the data from satellite or cellular network which is then passed on to the module. The module in turn sends the data to the software on the user's computer through a microcontroller. AT command is used to convert the 8 bit data which is sent through the microcontroller into a string value which is readable by the user on the software. Through the user interface such as Google API the owner then can see the tracked location with the corresponding longitude and latitude value on the Google map. The Google map pinpoints the exact location of the vehicle. We have used uses an open source software and is built using C# (.Net Framework) is the programming language for the software. There are different protocols though which the GSM/GPS module can communicate synchronously with the satellites. We have used NEMA protocol. NEMA protocol sends 8 bit data, and 1 stop bit and a final no parity bit. The advantage of using NEMA protocol is that it automatically synchronizes with the satellites with a baud rate when in open air. The higher the baud rate is, the greater is the time for synchronization. So, NEMA protocol usually uses a 4800 baud/s for synchronizing with the three satellites.

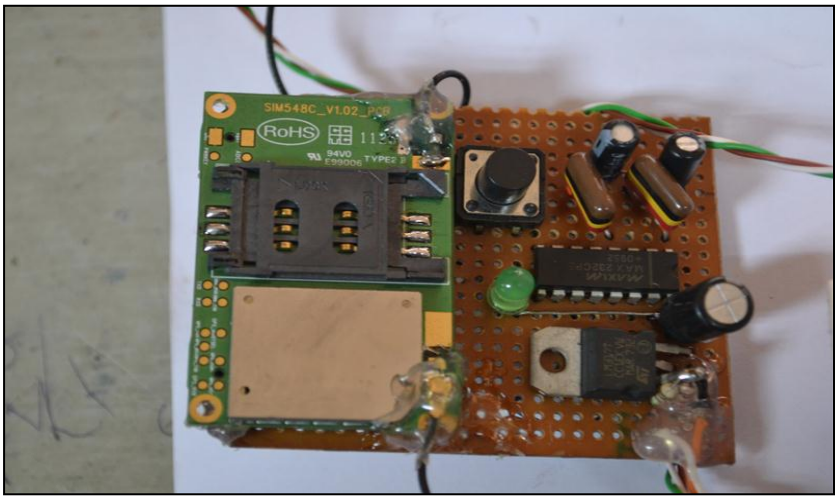


Fig: 5 SIMCOM 548c

The continuous data transmission for the GPS/GPRS tracking from the GPS device is wastage when tracking is not required because it would incur the owner the cost of monitoring charges. Hence we have incorporated the GSM module with the GPS device. This not only saves the owner the cost of buying the devices individually but also cut off the monitoring charges since this mechanism is completely user dependent. As the system runs on SIM card other than the SIM card charge in the unit, no additional costs are included.

Software for GPS

The GSM is used for sending SMS and receiving reply. We have developed a PC based software using C# (.NET Framework) is the programming language used for the software. The data transmission and reception is similar to the GPS/GPRS tracking system but this mechanism would activate only when the user sends a SMS through the GSM module connected to his/her PC through serial port, to inquire about the vehicle's location. When the user sends a SMS to the GSM device installed on the vehicle, the device sends a reply with longitude and latitude value to the user's PC software. Through Google API the user will view the Google map and the map pinpoints the exact location of the vehicle.

Our Software gets Latitude and Longitude in two ways. If COM Port of our GPS module is connected into PC than the software directly get the data from serial port and on the other hand if COM port connected into the on board controller than the software gets the value via SMS. For GPS Tracker Software development, GPS - NMEA sentence information is very important. In GPS NMEA **(National Marine Electronics Association)** Communication there are 19 Interpreted sentences.

1. $GPBOD - Bearing, origin to destination
2. $GPBWC - Bearing and distance to waypoint, great circle
3. $GPGGA - Global Positioning System Fix Data
4. $GPGLL - Geographic position, latitude / longitude
5. $GPGSA - GPS DOP and active satellites
6. $GPGSV - GPS Satellites in view
7. $GPHDT - Heading, True
8. $GPR00 - List of waypoints in currently active route
9. $GPRMA - Recommended minimum specific Loran-C data
10. $GPRMB - Recommended minimum navigation info
11. $GPRMC - Recommended minimum specific GPS/Transit data
12. $GPRTE - Routes
13. $GPTRF - Transit Fix Data
14. $GPSTN - Multiple Data ID
15. $GPVBW - Dual Ground / Water Speed
16. $GPVTG - Track made good and ground speed
17. $GPWPL - Waypoint location
18. $GPXTE - Cross-track error, Measured
19. $GPZDA - Date & Time

For Software Development we use the NEMA sentence which is **$GPGGA**. Because this NEMA is Global Positioning System Fix Data. Time, position and fix related data for a GPS receiver.

$GPGGA,hhmmss.ss,llll.ll,a,yyyyy.yy,a,x,xx,x.x,x.x,M,x.x,M,x.x,xxxx*hh

hhmmss.ss = UTC of position
llll.ll = latitude of position
a = N or S
yyyyy.yy = Longitude of position
a = E or W
x = GPS Quality indicator (0=no fix, 1=GPS fix, 2=Dif. GPS fix)
xx = number of satellites in use
x.x = horizontal dilution of precision
x.x = Antenna altitude above mean-sea-level
M = units of antenna altitude, meters

x.x = Geoidal separation
M = units of geoidal separation, meters
x.x = Age of Differential GPS data (seconds)
xxxx = Differential reference station ID

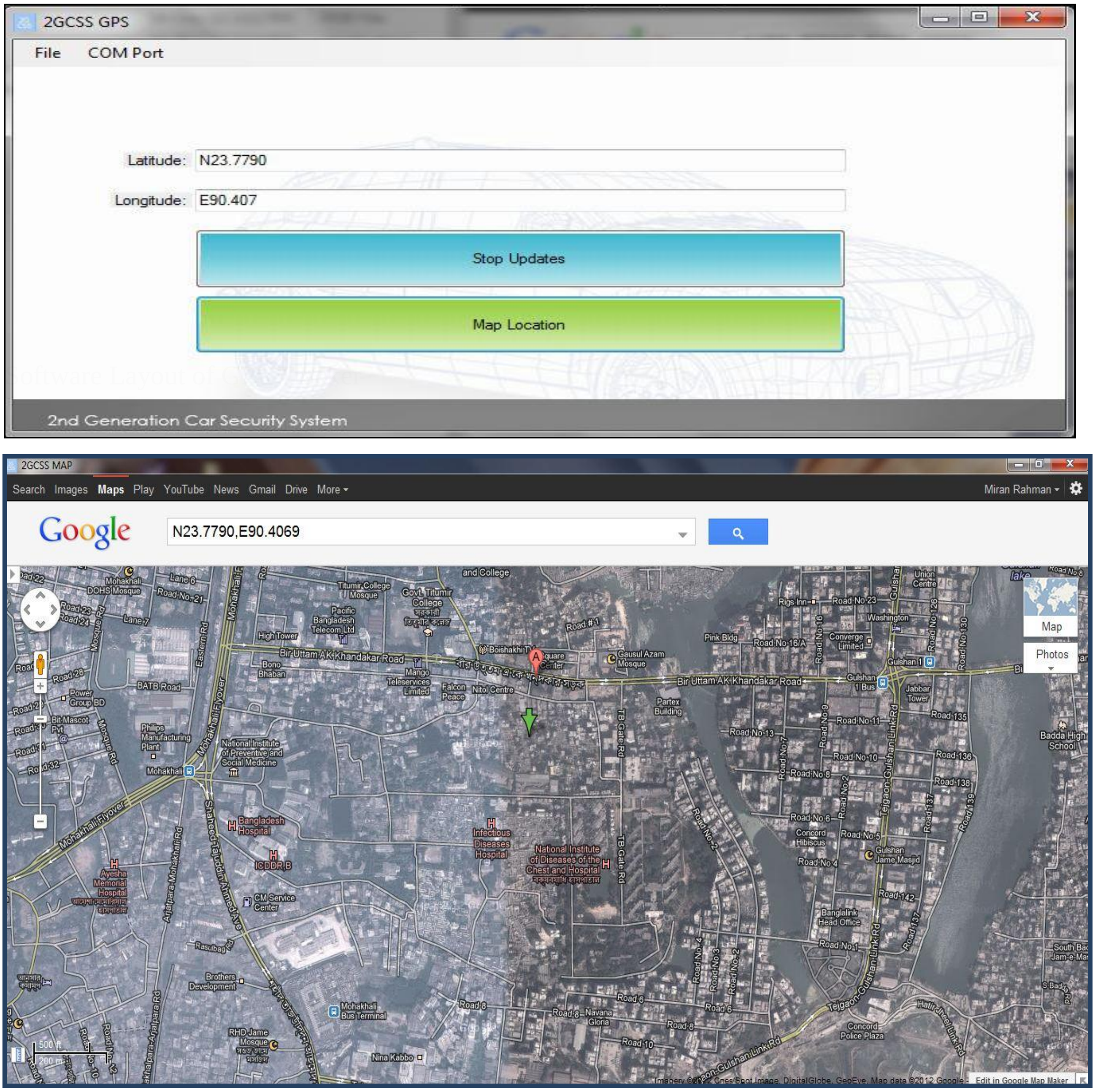


Fig: 6 Software Layouts

Latitude and Longitude calculation in software


```
if (lineArr[0] == "GPGGA")
                {
```

```
try
                              {
//Latitude
Double dLat = Convert.ToDouble(lineArr[2]);
                                  dLat = dLat / 100;
string[] lat = dLat.ToString().Split('.');
                                  Latitude = lineArr[3].ToString() +
lat[0].ToString() + "." + ((Convert.ToDouble(lat[1]) /
60)).ToString("#####");

//Longitude
Double dLon = Convert.ToDouble(lineArr[4]);
                                  dLon = dLon / 100;
string[] lon = dLon.ToString().Split('.');
                                  Longitude = lineArr[5].ToString() +
lon[0].ToString() + "." + ((Convert.ToDouble(lon[1]) /
60)).ToString("#####");

//Display
                                  txtLat.Text = Latitude;
                                  txtLong.Text = Longitude;

                                  btnMapIt.Enabled = true;
                              }
```

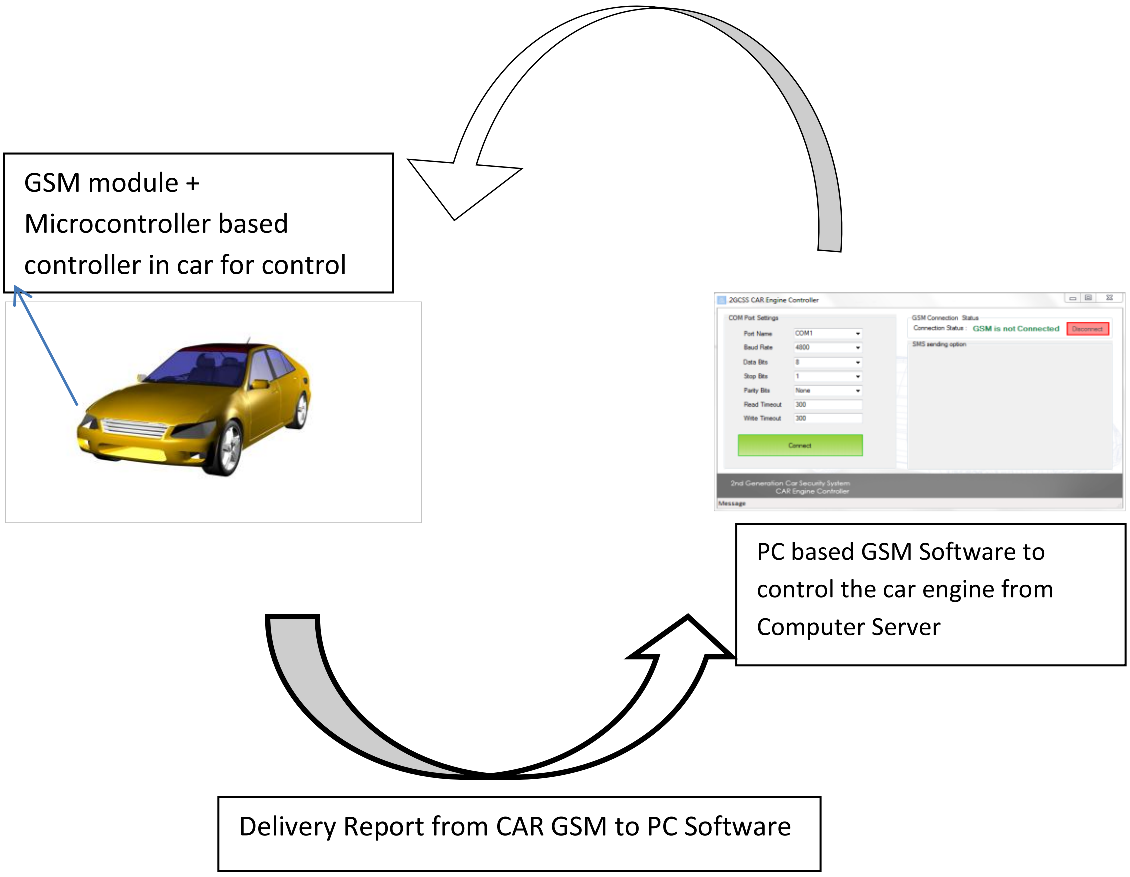
In the existing GPS/GPRS tracking devices use of smart-phones, net connection and continuous data transmission was required. Hence to make the device more efficient and to consume less power we have developed this SMS based system with the use of GSM technology so that via a push SMS the vehicles location is tracked. This technique also differs in the subsequent cost of service and is cost effective since it diminishes the unnecessary transmission of data.

GSM Based Engine On/Off System

SMS used on modern handsets are originated from radio technology in radio memo pages using standardized phone protocols. Later defined as a part of the Global system for mobile communication (GSM), it is used as a means of sending messages to or from mobile handsets or GSM modules. Using this concept we have configured the engine ON/OFF system through SMS. The user can easily turn the engine on or off by sending a SMS code. The ON/OFF technique is directed by a simple command which is then imported to the microcontroller on the GSM module. From the microcontroller the output would be directed to the wire that goes to the relay of engine which is situated in relay BOX of an Automobile (like: Car).

Basic Concept

2GCSS Car Engine Controller BLOCK Diagram



Mechanism

Our system is based on microcontroller and we are using*PIC18f2550-i/sp. For RS 232 CommunicationMAX232CP IC* (+5V-Powered, Multichannel RS-232 Drivers/Receivers).For project purposes we use LED instead of engine relay.

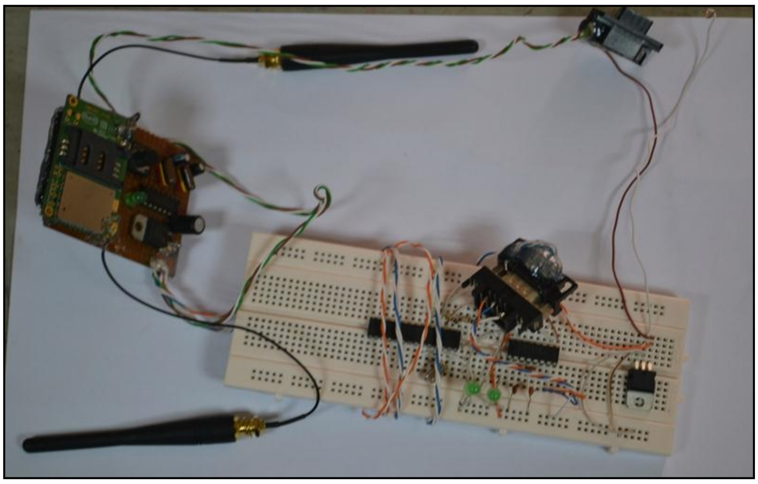


Fig: 7 Circuit of SMS based engine on /off system (Raw level design)

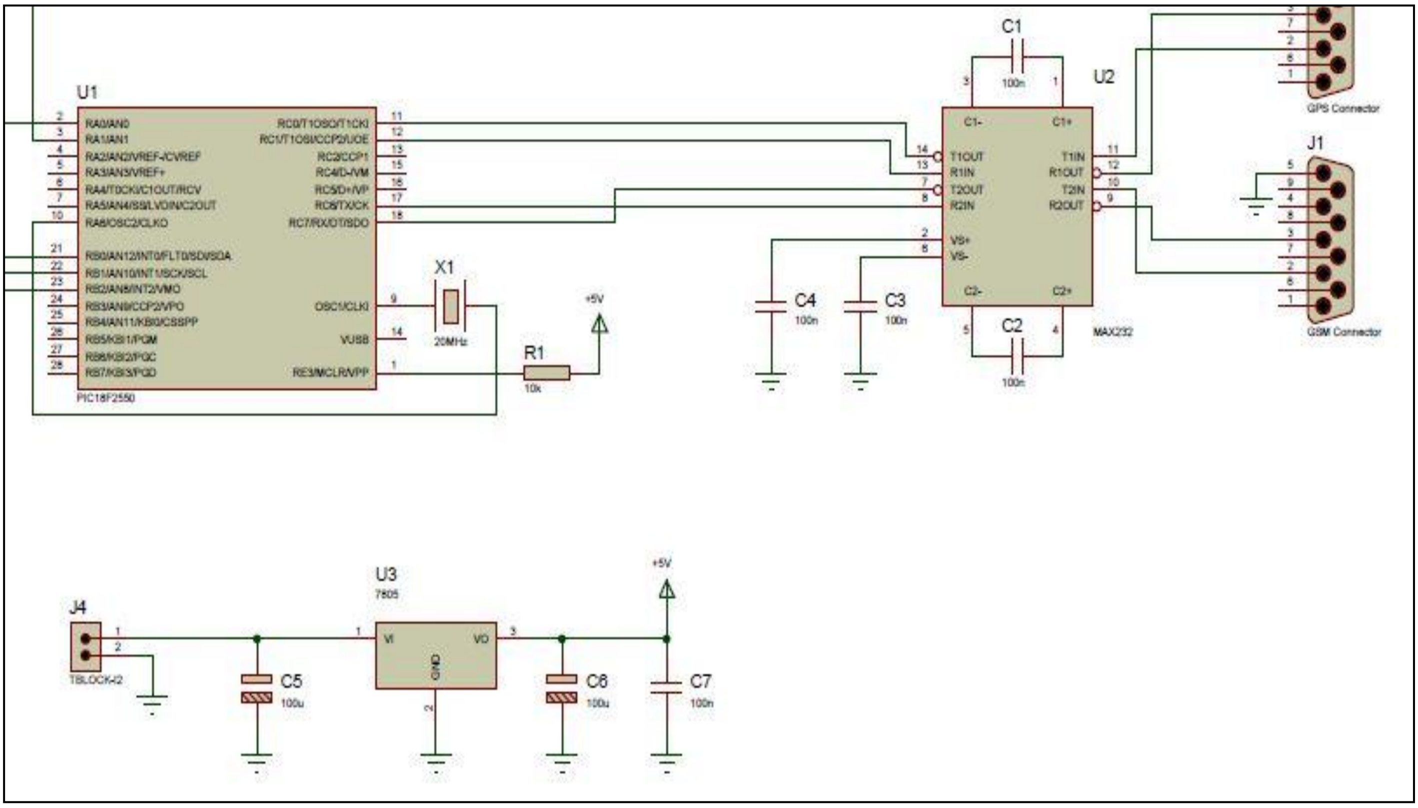


Fig: 8 Circuit diagram of SMS based engine on /off system

Software

For Sending SMS and receive the delivery report we develop a PC based software. C# .Net is the programming language of this software.

Besides installing a GPS/GSM module on the car, another GSM module is used for the PC as well's module is connected to the PC via serial port. In the GSM module a SIM card is inserted and is configured with the PC software. Without this SIM nobody can send SMS from this software. To turn OFF the engine the user sends a SMS from the computer software to the SIM card number which is set in the GSM module installed in the car. The SMS is a unique code which is only known by the user, for example "sw off". When the GSM module on the car receives this command, it is directed to the START wire of the ignition switch. The relay connected to the wire then opens its contact and hence prevents current flow causing the engine to turn OFF. Similarly to start the engine automatically the user sends a SMS with a unique code, for example "sw on". The output is directed to the START wire. With this command the relay closes its contact forming a closed loop and allows current flow causing the engine to turn ON. In each case a delivery report from the car GSM module is sent to the user's PC software to show the status of the engine. Hence this mechanism immobilizes the vehicle and prevents engine start when it has been stolen unless authorized by sending the engine the ON command.

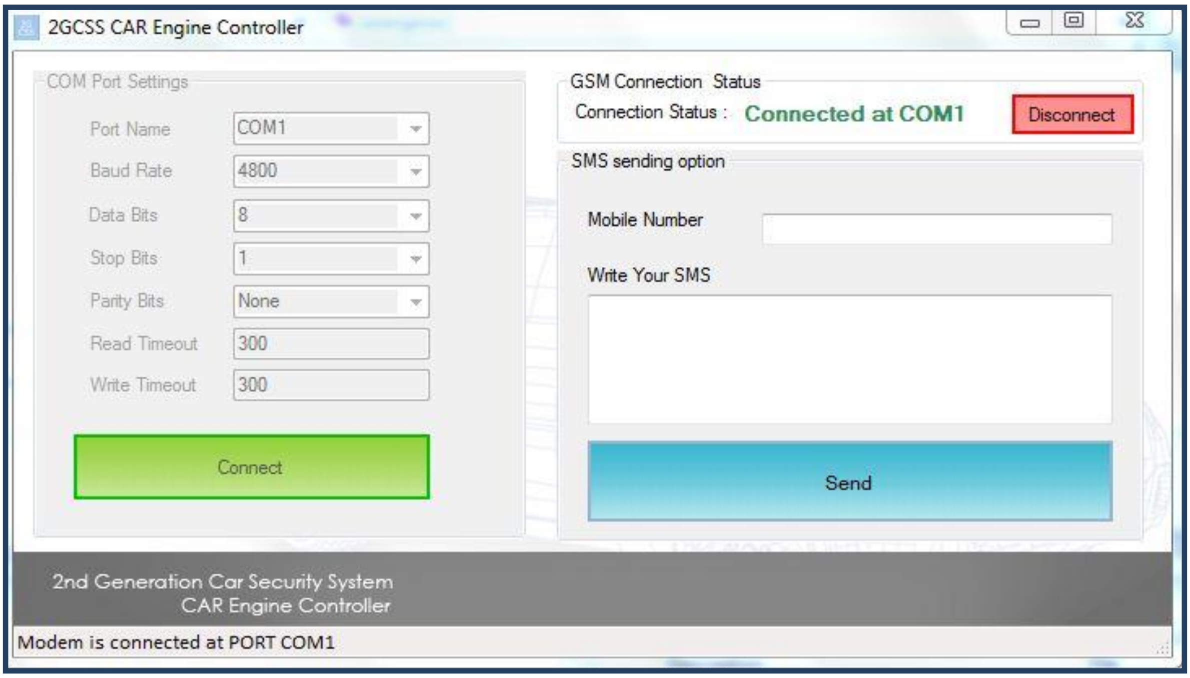


Fig: 9 Software Layout

Code for sending SMS

```
namespace _2GCSS_GSM_Car_Engine_Controller
{
    public partial class carengine : Form
    {
        #region Constructor
      public carengine()
        {
            InitializeComponent();
        }
        #endregion
      #region Private Variables
      SerialPort port = new SerialPort();
      gcss objgcss = new gcss();
     //     ShortMessageCollection    objShortMessageCollection     =     new
ShortMessageCollection();
      #endregion
      #region Private Methods

      #region Write StatusBar
      private void WriteStatusBar(string status)
      {
          try
          {
              statusBar1.Text = "Message: " + status;
          }
          catch (Exception ex) {
          }
      }
      #endregion

      #endregion

        #region Private Events
```

```
        private void SMSapplication_Load(object sender, EventArgs e)
        {
            try
            {
                #region Display all available COM Ports
                string[] ports = SerialPort.GetPortNames();


                foreach (string port in ports)
                {
                    this.cboPortName.Items.Add(port);
                }
                #endregion
                 this.btnDisconnect.Enabled = false;
    }
            catch(Exception ex)
            {
                ErrorLog(ex.Message);
            }
}

        private void btnOK_Click(object sender, EventArgs e)
        {
        try
            {

                this.port     =     objgcss.OpenPort(this.cboPortName.Text,
Convert.ToInt32(this.cboBaudRate.Text),
Convert.ToInt32(this.cboDataBits.Text),
Convert.ToInt32(this.txtReadTimeOut.Text),
Convert.ToInt32(this.txtWriteTimeOut.Text));

                else
                {
                    //MessageBox.Show("Invalid port settings");
                    this.statusBar1.Text = "Invalid port settings";
                }
            }
            catch (Exception ex)
            {
                ErrorLog(ex.Message);
            }
        }

        private void btnDisconnect_Click(object sender, EventArgs e)
        {
        try
            {
                this.gboPortSettings.Enabled = true;
                objgcss.ClosePort(this.port);
            this.lblConnectionStatus.Text = "GSM is Disconnected";
                this.btnDisconnect.Enabled = false;
                txtSIM.Visible = false;
                txtMessage.Visible = false;
                label10.Visible = false;
                label2.Visible = false;
                btnSendSMS.Visible = false;
            }
            catch (Exception ex)
            {
```

```
                ErrorLog(ex.Message);
            }

        }

        private void btnSendSMS_Click(object sender, EventArgs e)
        {
            try
            {

                if       (objgcss.sendMsg(this.port,       this.txtSIM.Text,
this.txtMessage.Text))
                {

                    this.statusBar1.Text = "SMS is successfully sent";
                }
                else
                {

                    this.statusBar1.Text = "Failed to send message";
                }

            }
            catch (Exception ex)
            {
                ErrorLog(ex.Message);
            }
        }
        #endregion
        #region Error Log
        public void ErrorLog(string Message)
        {
            StreamWriter sw = null;

            try
            {
                WriteStatusBar(Message);

                string                     sLogFormat                =
DateTime.Now.ToShortDateString().ToString()      +       "       "       +
DateTime.Now.ToLongTimeString().ToString() + " ==> ";

                string sPathName = @"2gcsserror_";

                string sYear = DateTime.Now.Year.ToString();
                string sMonth = DateTime.Now.Month.ToString();
                string sDay = DateTime.Now.Day.ToString();

                string sErrorTime = sDay + "-" + sMonth + "-" + sYear;

                sw  =  new  StreamWriter(sPathName  +  sErrorTime  +  ".txt",
true);

                sw.WriteLine(sLogFormat + Message);
                sw.Flush();


            }
            catch (Exception ex)
            {
            }
```

```
            finally
            {
                if (sw != null)
                {
                    sw.Dispose();
                    sw.Close();
                }
            }

        }
```

We use a GSM modem for this pc software's modem is connected into the PC via COM port. In GSM module we insert a SIM and without this SIM nobody can send SMS from this software.

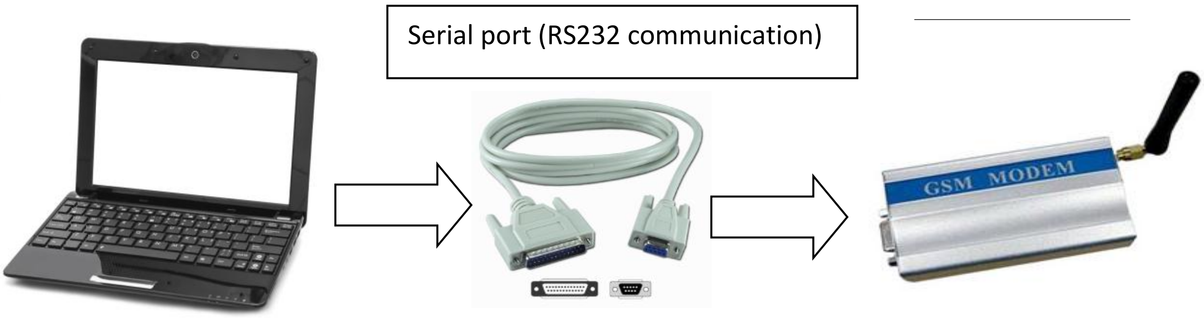


Fig: 10 connect the GSM module via Serial port (RS232 communication)

Blind Spot/Point Detection

A blind spot in a vehicle is an area around the vehicle that cannot be directly observed by the driver while at the controls and cannot be detected using side mirror. So if any car wants to overtake then it causes a serious accident. In transports, the driver visibility is the maximum distance at which the driver of a vehicle can see and identify prominent objects around the vehicle. Good driver visibility is essential to safe road traffic.

Mechanism:

When a driver drives a car he can only see a car whose position is more than 40 degree horizontally from the driving car. For this purpose we have used sonar to find whether there is any car between 0-45 degrees horizontally from the driving car at a maximum distance of approximately 150 cm. The sonar module is invisibly mounted in the corners of the rear bumper. The sonar here acts as the blind spot. For instance carA enters the blind spot of car B while the driver of carB was switching lanes, the sonar module of carB detects the moving vehicles behind or beside it, such as carB and notify the driver by illuminating a light on the rear view mirror and also sends and display the distance of the car A from it to the LCD mounted on the car's dashboard. Thus it alerts the driver to reduce the speed or to halt which reduces the risk of accidents during switching lanes or overtaking. It is basically optimized for highway driving and also for driving in free roads. It has been extensively tested and ended up in an optimum result with a low false alarm (illumination of light in the rear view mirror) rate.

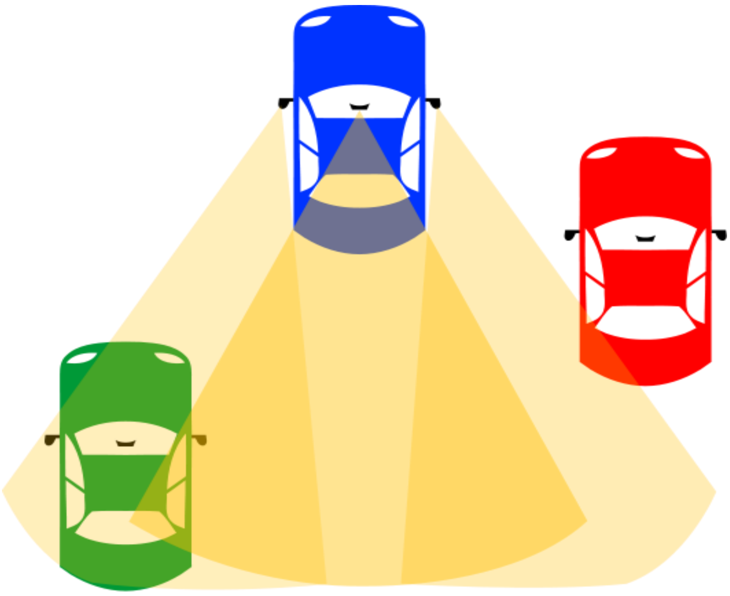


Fig: 11 Blind Point

Hardware:

To develop this system we used a SONAR module. Our system is based on microcontroller and we are using PIC18F4550 .For display we use LCD display unit.

Detection algorithm, object-to line mapping and the alerting algorithm are performed by the microcontroller in the system. When the vehicle disappears from the blind spot, the LED light turns off.

We are planning to extend this system by combining digital cameras with object detection radars which will deliver the image. This detection system alerts drivers to any object in their blind spot and they can also see it on the mini monitor inside the car. Integrating cameras with an object-detection radar system combines the best of both technologies into one active system for the drivers.

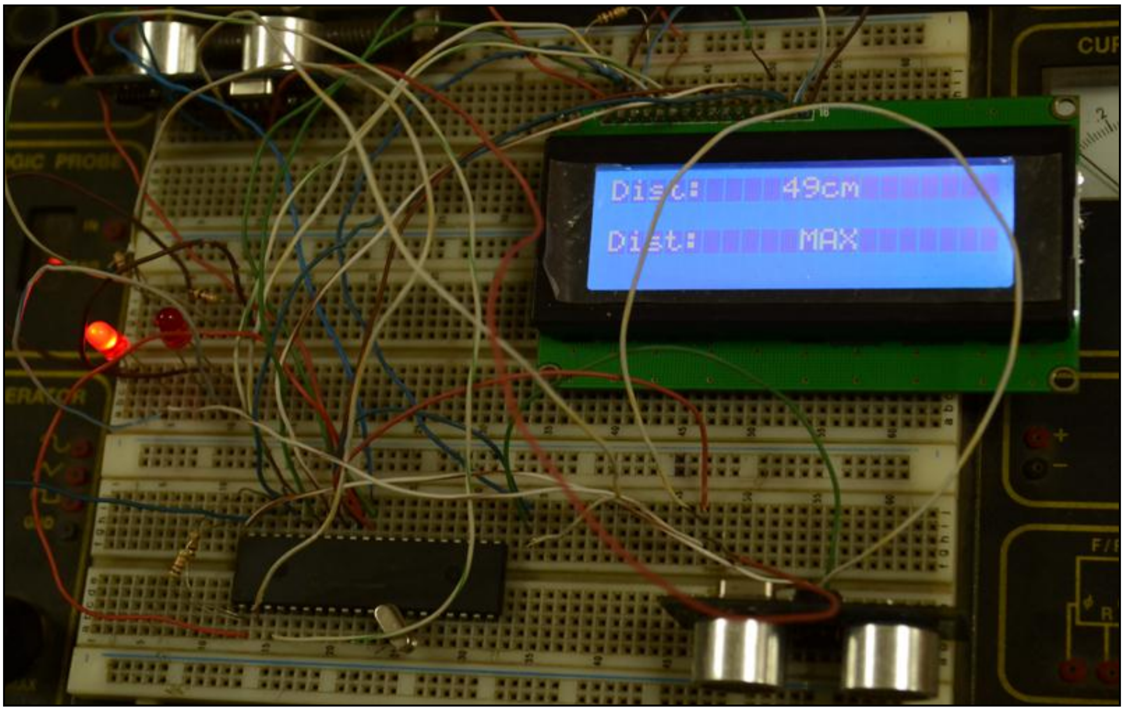


Fig: 12 Circuit of Blind Point Detection (Raw level design)

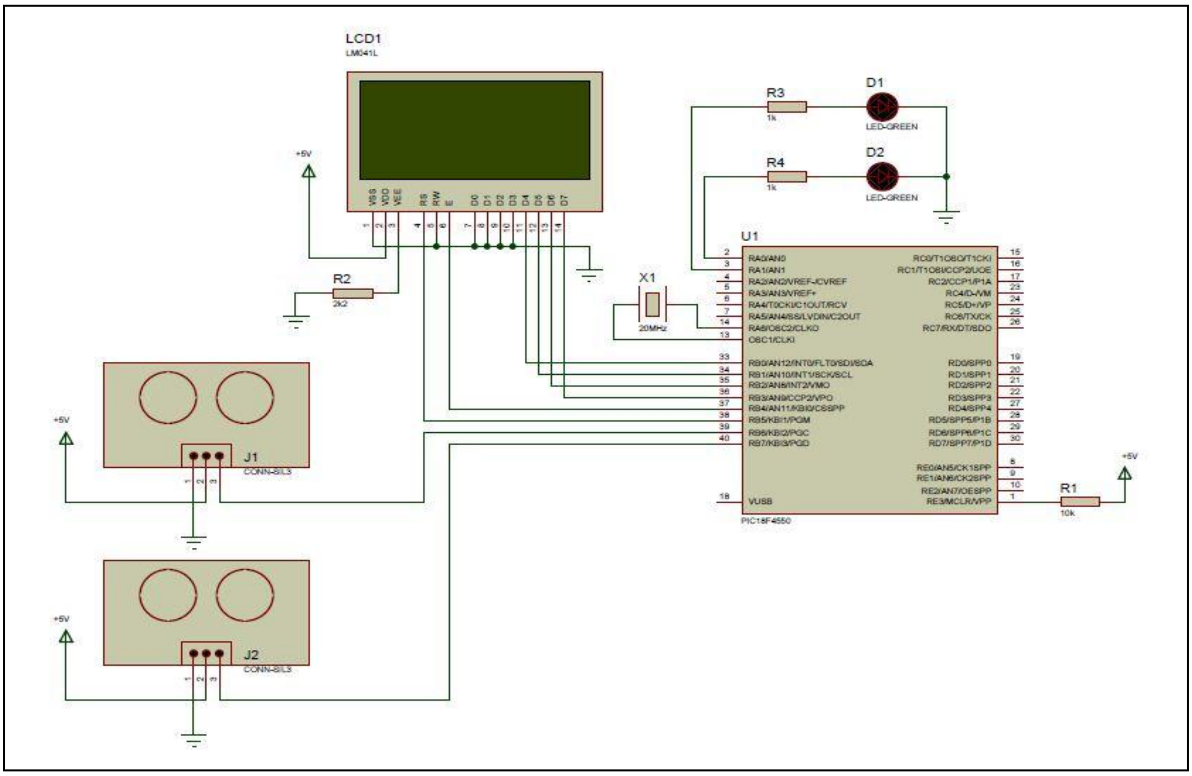


Fig: 13 Circuit diagram of Blind Pont detection

Parking Assistant:

The ultrasonic wave sensor is built into the corners of the bumper. It detects the distance to objects and notifies the driver display lamp alerts. The mechanism is almost same as the Blind point detection.

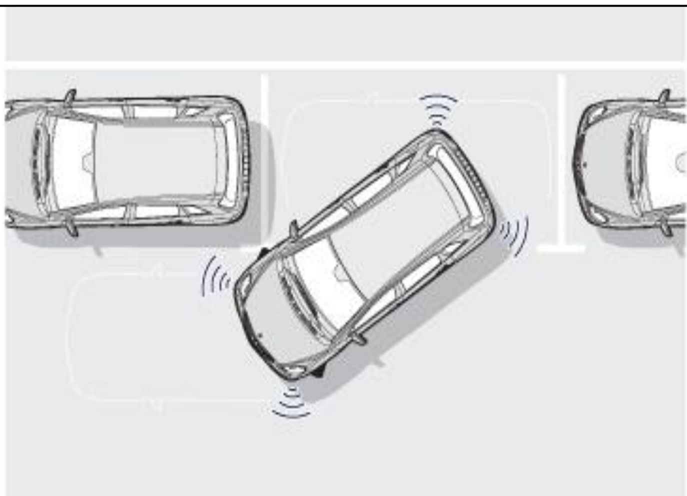


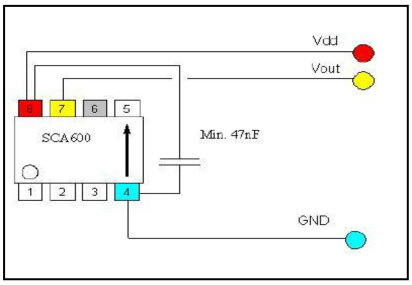
Fig:14 Parking Assistant System

Robust Jumper Alert and Warning Alarm:

Whenever the car experiences a heavy shake or jump from its position at rest, this system will alert the owner that someone is trying to mess with car. For this mechanism a very small electronic device called "Gyroscope" which is sensitive to changes in gravitational acceleration 'g' is used. The initial position is considered when the car is at rest. With the initial position the gyroscope calculates 'g' and generates a corresponding voltage and sets this voltage value as the reference voltage.

Mechanism:

During any shaking or unwanted bumps of the car when it's at rest position, the gyroscope generates various voltage readings corresponding with the changes in 'g'. Now, within the microcontroller, PIC18F2550, the code has been set such that if the generated voltage is not within ±0.5V of the reference voltage, the alarm, which is pin: 23 of the microcontroller will ring, that is, will be on. This ±0.5V has been considered as the maximum tolerance level of movement of the car. Thus the variation of voltage from the reference value will notify the owner with the alarm being turned on and also a SMS will be sent to the owners mobile.



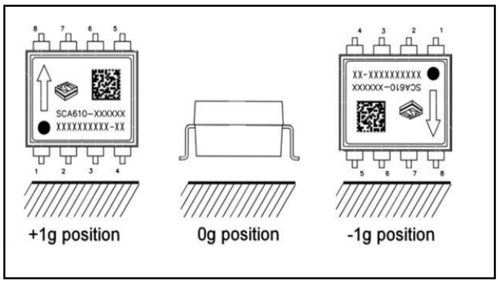


Fig: 15Pin configuration and Gyro sensor‘s positions

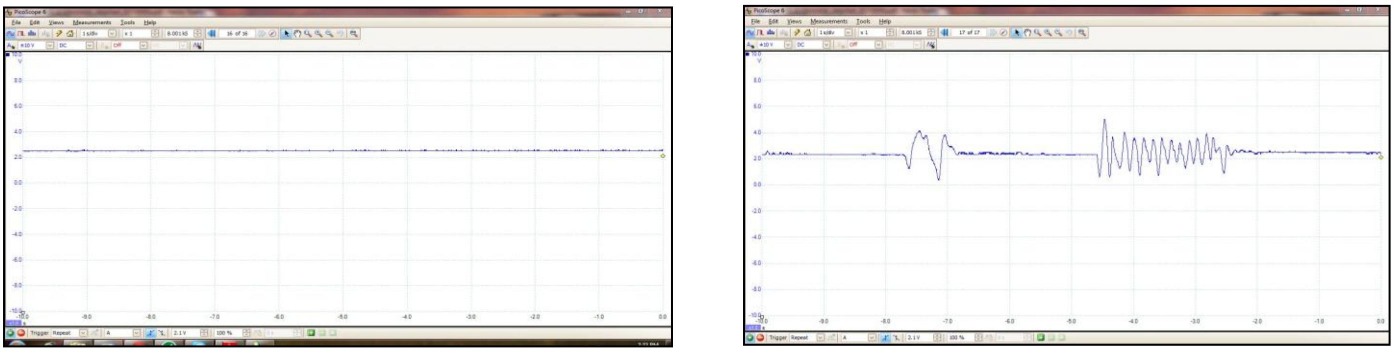


Fig: 16 Frequency change in Oscilloscope

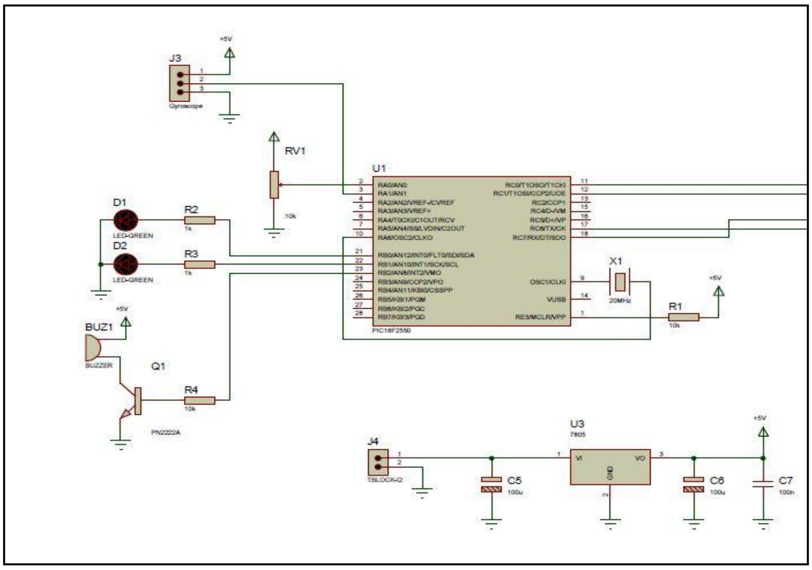


Fig: 17 Circuit diagram of Robust Jumper Alert and Warning Alarm

Passenger counter and data Log:

Passenger counter and data log system gave a different dimension in our thesis. Now our system can count the passengers and give a brief history about the passenger’s number with time. In this process we have used a switch so that whenever the driver switch on the engine all the other switches are activated which are under the seat cover. So if there is any passenger the loaded switch is pressed for single time. All this data are saved into the

EEPROM .A registered mobile number is associated with the system. We have counted it as the owner's mobile number. So whenever the owner send a message to the system writing "log", the EEPROM replies with the data and real time from GPS via SMS. Only the owner can reset the counter whenever he needs.

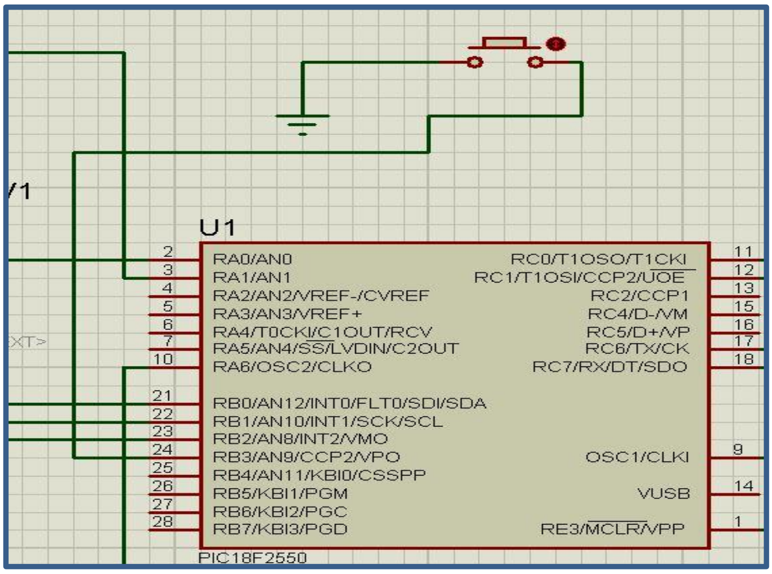


Fig: 18 Circuit diagram Passenger counter and data log

Code in microcontroller:

```
// send status to cellNumber
                    sendMessage(CellNumber, 0, NewMsgText, 0);
                    */
          }
          else if (strcmp(MsgCommand, "clear") == 0)
          {
                    EEPROM_write(0,0);
                    Delay_ms(10);
}
          else if (strcmp(MsgCommand, "log") == 0)
          {
                    // send Counter to cellNumber
                    setString(NewMsgText, "Counter: ");
                    Temp = EEPROM_Read(0);
```

```
            ByteToStr(Temp, TempStr);

            strcat(NewMsgText, TempStr);

            sendMessage(CellNumber, 0, NewMsgText, 0);

    }
```

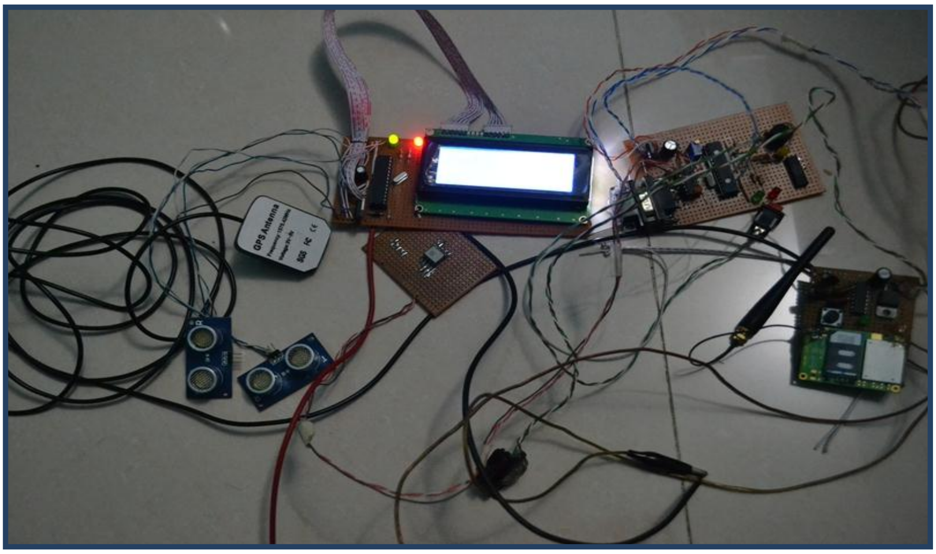


Fig: 19 Full system

Future Plan:

We have a plan to make this second generation car security system more effective, that's why we will implement more features in this system. Some of our future plans are given below. We are now planning to integrate a small digital camera inside the vehicle and interface it to the security system. The camera would be enabled and powered up, starting to obtain a video signal from the environment hopefully with the thief present in the frustum. A 10-20sec movie can be sent to the third generation mobile phone, like 3G, over the M2M link providing the rightful owner with audio and video of the scene.

Discussion:

During the design implementation process each system has been verified individually to ensure there was negligible error. After the integration of the total system the integrated unit has been tested and verified to make sure the implemented design is accurate and the system has been developed as such that it has met all the requirements defined for it.

References:

[1] http://www.rmiia.org/auto/auto_theft/statistics.asp

Last visited date: 12/12/2012

[2] Patrick Bertagna, GTX Corp. http://www.eetimes.com/design/communications-design/

Last visited date: 12/12/2012

[3] http://electronics.howstuffworks.com/question537.htm

Last visited date: 12/12/2012.

[4] Jiwa Abdullah, "The Design of Mobile Control Car Security System", IACSIT International Journal of Engineering and Technology, Vol.3, No.3, June 2011

[5]Shihab A.Hameed, Shaima Abdulla, Aisha Hassan, Othman Khalifa, Jamal I.Doud, "Effective car monitoring and tracking model", Australian Journal of basic and applied science, Vol, 6, No. 1 pp.1-8,2012

[6] Vadim Geurkov, Nasim Zoubeiri, Stephen chui and Ryan, "Vehicle security System", Case Study, Reyson University.

[7] http://www.autonavigationgps.com/how-does-a-gps-tracking-system-work/

Last visited date: 12/12/2012

[8] http://autotips.plentycar.com/car-anti-theft-devices/

Last visited date: 12/12/2012

[9] http://aprs.gids.nl/nmea/

Last visited date: 12/12/2012

[10] Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and Stefan Savage, "Tadayoshi Kohno, Experimental Security Analysis of a Modern Automobile", IEEE Symposium on Security and Privacy, 2010.

[11] http://www.rmiia.org/auto/auto_theft/statistics.asp

Appendix:

On Board Controller (Microcontroller Code)

```
#include "HardwareProfile.h"
#include "Library\AdvanceUartLibrary.h"
#include "Library\GSMModem.h"
#include "Library\MiscLibrary.h"

#define SAKE_THRSHLD 100


//BLink: "+88019900557"; GP: "+8801700000600"; Airtel: "+8801600006001"; Robi:
"+8801801000004"; TeleTalk: "+880150159999"
const char *ServiceCenterNumber = "+8801700000600";
char *UserNumber = "+8801722600610";
char MsgCommand[10];
char MsgParameter[10];
char TempStr[10];
char NewMsgText[100];

void InitGPS_Module();


void main()
{
	char *MsgTxtPtr, *StrPtr;
	unsigned char Shake_Flag = 0, Temp = 0, Gerbage;
	unsigned int G_ScopeReading, Old_G_Value = 0;
	unsigned int Timer = 0;

	//--- Configure System ---
	ConfigureSystem();
	InitializeIO();

	// set GPS port
	_O_MUX = 1;
	UART1_Init(4800);
	Delay_ms(100);
	InitGPS_Module();

	// set GSM port
	_O_MUX = 0;
	UART1_Init(9600);
	Delay_ms(100);

	// set on
	_O_STAT_LED = 1;
```

```
	// initilize the modem
	InitModem();

	// successfully initilized, so set off
	_O_STAT_LED = 0;

	// set Old_G_Value at first startup
	Old_G_Value = ADC_Read(0);

	while(1)
	{
		// read G sensore reading
		Old_G_Value = ADC_Read(0);
		G_ScopeReading = ADC_Read(1);

		if(button(_B_ING_KEY,10,0))
		{
			Temp = EEPROM_Read(0);
			Temp++;
			EEPROM_Write(0, Temp);

			// ack Status
			Delay_ms(300);
			_O_STAT_LED = 1;
			Delay_ms(300);
			_O_STAT_LED = 0;
		}

		if(!Shake_Flag)
		{
			if((G_ScopeReading > (Old_G_Value + SAKE_THRSHLD)) ||
(G_ScopeReading < (Old_G_Value - SAKE_THRSHLD)))
			{
				_O_BUZZER = 1;
				Delay_ms(100);
				_O_BUZZER = 0;
				Delay_ms(100);

				_O_BUZZER = 1;
				Delay_ms(100);
				_O_BUZZER = 0;
				Delay_ms(100);

				Shake_Flag = 1;
				Timer = 0;

				// send status to UserNumber
				sendMessage(UserNumber, 0, 0, "Dhakka khaisere, dhakka
khaise. Taratari check koren.");
			}
```

```
}
else
{
        _O_BUZZER = 1;
        Delay_ms(100);
        _O_BUZZER = 0;
        Delay_ms(100);

        _O_BUZZER = 1;
        Delay_ms(100);
        _O_BUZZER = 0;
        Delay_ms(100);
        Timer++;
}

if(Timer > 20)
{
        Shake_Flag = 0;
        Timer = 0;
}

// set new value
Old_G_Value = G_ScopeReading;

if(isNewMsgReceived())
{
        // blink status
        _O_STAT_LED = 1;
        Delay_ms(100);
        _O_STAT_LED = 0;
        Delay_ms(100);
        _O_STAT_LED = 1;
        Delay_ms(100);
        _O_STAT_LED = 0;

        // parse text from the message
        MsgTxtPtr = getMsgText();

        // extract and copy MsgCommand
        StrPtr = strtok(MsgTxtPtr, " ");
        strCpy(MsgCommand, StrPtr);

        // extract and copy MsgParameter
        StrPtr = strtok(0, " ");
        strCpy(MsgParameter, StrPtr);

        // convert to lower case
        convertToLower(MsgCommand);
        convertToLower(MsgParameter);
```

```
if(strcmp(MsgCommand, "status") == 0)
{
	// ack Status
	Delay_ms(300);
	_O_STAT_LED = 1;
	Delay_ms(300);
	_O_STAT_LED = 0;

	/*
	// set header text
	setString(NewMsgText, "Status:\n");

	// get Temperature
	StrPtr = appendString(NewMsgText, "Temp: ");
	ByteToStr(Temperature, TempStr);
	StrPtr = strcat(StrPtr, TempStr);
	StrPtr = appendString(StrPtr, " Deg\n");

	// get Power
	StrPtr = appendString(StrPtr, "Power: ");
	ByteToStr(Power, TempStr);
	StrPtr = strcat(StrPtr, TempStr);
	StrPtr = appendString(StrPtr, " Watt\n");

	// send status to cellNumber
	sendMessage(CellNumber, 0, NewMsgText, 0);
	*/
}
else if (strcmp(MsgCommand, "clear") == 0)
{
	EEPROM_write(0,0);
	Delay_ms(10);
}
else if (strcmp(MsgCommand, "log") == 0)
{
	// send Counter to cellNumber
	setString(NewMsgText, "Counter: ");
	Temp = EEPROM_Read(0);
	ByteToStr(Temp, TempStr);
	strcat(NewMsgText, TempStr);
	sendMessage(CellNumber, 0, NewMsgText, 0);
}
else if (strcmp(MsgCommand, "pos") == 0)
{
	// default NEMA text
	//
$GPGGA,123906.217,2349.0810,N,09022.1960,E,0,03,,50.8,M,-50.8,M,,0000*64

	// set GPS port
	_O_MUX = 1;
```

```
                    UART1_Init(4800);
                    Delay_ms(50);

                    // read until '$' charecter found
                    Gerbage = RCREG;
                    while(1)
                    {
                        while(!RCIF_bit) { }
                        if(RCREG == '$') { break; }
                    }
                    readUART(1000);

                    // set GSM port
                    _O_MUX = 0;
                    UART1_Init(9600);
                    Delay_ms(50);

                    // send NEMA sentence to cellNumber
                    sendMessage(CellNumber, 0, UartBuffer, 0);
                }
                else if (strcmp(MsgCommand, "sw") == 0)
                {
                    // ack Status
                    Delay_ms(300);
                    _O_STAT_LED = 1;
                    Delay_ms(300);
                    _O_STAT_LED = 0;

                    // get switch state from MsgParameter
                    if(strcmp(MsgParameter, "on") == 0 || strcmp(MsgParameter,
"1") == 0)
                    {
                        _O_OUTPUT1 = 1;
                    }
                    else if(strcmp(MsgParameter, "off") == 0 ||
strcmp(MsgParameter, "0") == 0)
                    {
                        _O_OUTPUT1 = 0;
                    }
                }
            }
        }
}


void UART_WriteText(const char *Txt)
{
        while(*Txt)
        {
                UART1_Write(*Txt);
```

```
            Txt++;
        }
}


void InitGPS_Module()
{
        // GGA: 1 sec interval
        UART_WriteText("$PSRF103,00,00,01,01*25\r\n");
        Delay_ms(50);

        // GLL: Disable
        UART_WriteText("$PSRF103,01,00,00,01*25\r\n");
        Delay_ms(50);

        // GSA: Disable
        UART_WriteText("$PSRF103,02,00,00,01*26\r\n");
        Delay_ms(50);

        // GSV: Disable
        UART_WriteText("$PSRF103,03,00,00,01*27\r\n");
        Delay_ms(50);

        // RMC: Disable
        UART_WriteText("$PSRF103,04,00,00,01*20\r\n");
        Delay_ms(50);

        // VTG: Disable
        UART_WriteText("$PSRF103,05,00,00,01*21\r\n");
        Delay_ms(50);

        // MSS: Disable
        UART_WriteText("$PSRF103,06,00,00,01*22\r\n");
        Delay_ms(50);

        // ZDA: Disable
        UART_WriteText("$PSRF103,08,00,00,01*2C\r\n");
        Delay_ms(50);

        // DEBUG: Disable
        UART_WriteText("$PSRF105,0*3F\r\n");
        Delay_ms(50);
}
```