Volume: 2, Issue: 1
Page: 1-16
2019

Journal of Scientific Reports

# Embedded system for monitoring and controlling traffic accident in public transport

**Mr. Temesgen Mengistu Helana**

## Abstract

This project discusses and Implements in detail the purpose, designing and functioning of a automatic vehicle tracking and traffic accident monitoring system in public transport. The number of vehicles on the road globally is expected to increase rapidly. Therefore, the development of vehicle tracking system using the Global Positioning System (GPS) and Global System for Mobile Communications (GSM) modem is undertaken with the aim of enabling to minimize traffic accident and users to locate their vehicles and to keep the users from driving over speed with ease and in a convenient manner. The system will provide to the administrator with the capability to track vehicle remotely through the mobile network and at what speed the vehicle is running. Specifically, implementation of this project more focus on the server based vehicle tracking in public transport and company vehicles in case of traffic accident prevention. The system will generate the location of the vehicle and at what speed the vehicle is moving. The system will utilize GPS to obtain a vehicle's coordinate and the time and transmit it using GSM modem to the central server through the mobile network by SMS. The main hardware components of the system are Robomart GPS receiver module, Robomart GSM module and AVR Atmega16 microcontroller. The developed vehicle tracking system demonstrates the feasibility of near real-time tracking of vehicles and improved customizability, global operability and costly when compared to existing solutions.

**JSR**
Accepted 03 March 2020
Published 26 March 2020
DOI: 10.5281/zenodo.3727640

**Keywords**: traffic accident, vehicle tracking, microcontroller, GPS, GSM, Monitoring.

About Author

**Mr. Temesgen Mengistu Helana**, Department of Computer Science, Wolaita Sodo University, WSU, Wolaita Sodo, Ethiopia.

## 1. Introduction

The global number of vehicles is expected to increase as ownership becomes more affordable due to the growing economies of countries such as China, India and some African counties. However, the adoption of vehicle tracking system is still very much lacking. Such a system can be used for many applications including security of personal vehicle, public transportation systems, fleet management and others. Vehicle tracking systems have been available in the market for some time but they are application specific, region specific and are costly. Therefore, a system designed for car security will not be suitable for fleet management. It is envisioned that the proposed system will be easily customizable for various applications. The proposed system can be used nationally and is expected to be cheaper. On the other hand, Global Positioning System (GPS) navigation system is widely adopted in vehicles today. It is primarily used to assist the driver in navigating to their destinations with turn-by-turn instructions. It can also be used in tracking the distance traveled on a trip, vehicle mileage, and speed. It can keep the record of driving activity including address of each destination, names of streets traveled, and how long the vehicle remained at each location. These functions allow users to monitor the usage of their vehicles. However, these systems will not be able to track the vehicle remotely. Including all the above mentioned features and the system will store every serious incident of the vehicle on the central server when and it lets the monitors know the situations happened.

GPS is a real-time satellite navigation system for three dimensional position determinations. It was developed by several U.S government organizations, including the Department of Defense (DOD), the National Aeronautic and Space Administration (NASA), and the Department of Transportation (DOT). GPS has three segments: satellite constellation, ground-control/monitoring network, and user receiving equipment. The satellite constellation is the set of satellites in orbit that provide the ranging signals and data messages to the user equipment. The control segment oversees and maintains the space segment, which is the satellite constellation in space. The user segment, or the user receiver equipment, receives the signal from the space segment and computes the navigation, timing and other functions. [4]

## 2. Architecture of the Proposed system

The main purpose behind this project is “controlling the drivers from causing accident”. Now a days, many accidents are happening because of the drivers mistake of the driving over speed or drunk driving and disturbed driving or and others. Thus over speed driving is a major reason of accidents in almost all countries all over the world. Speed and location tracker in this system designed for the safety of the people seating inside and outside the car. This project should be installed inside the vehicle and at the same time it will be fixed on the central server.

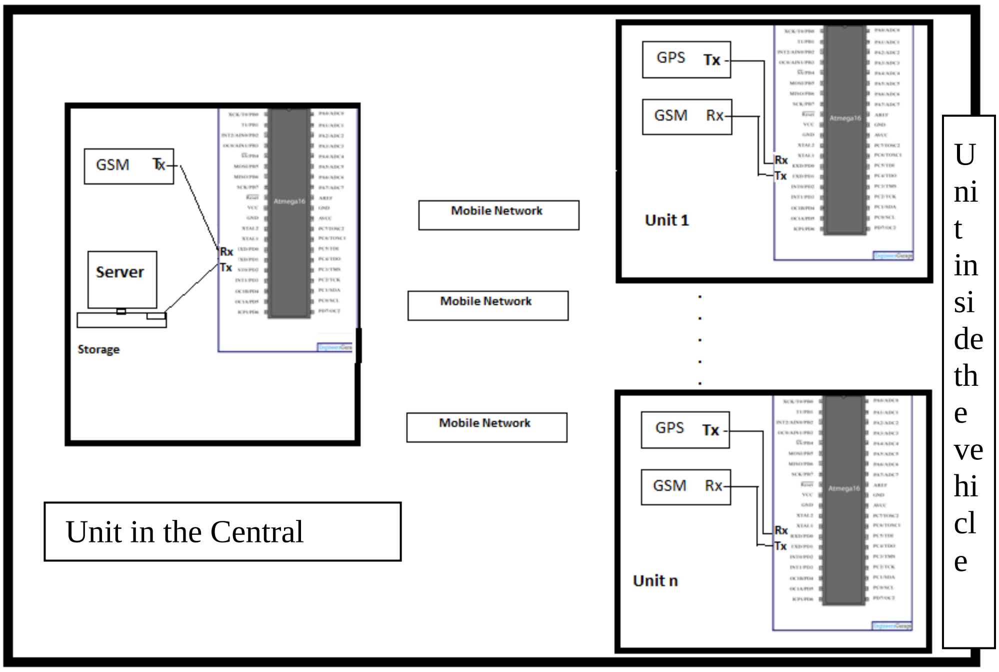


Fig 3: The architectural diagram of Tracking System

## 3. Circuit Diagram

The circuit diagram below depicts how each internal modules were interconnected in the circuitry and the simulation of the system. Based on the diagram we can easily understand structure of the system and visualize it.

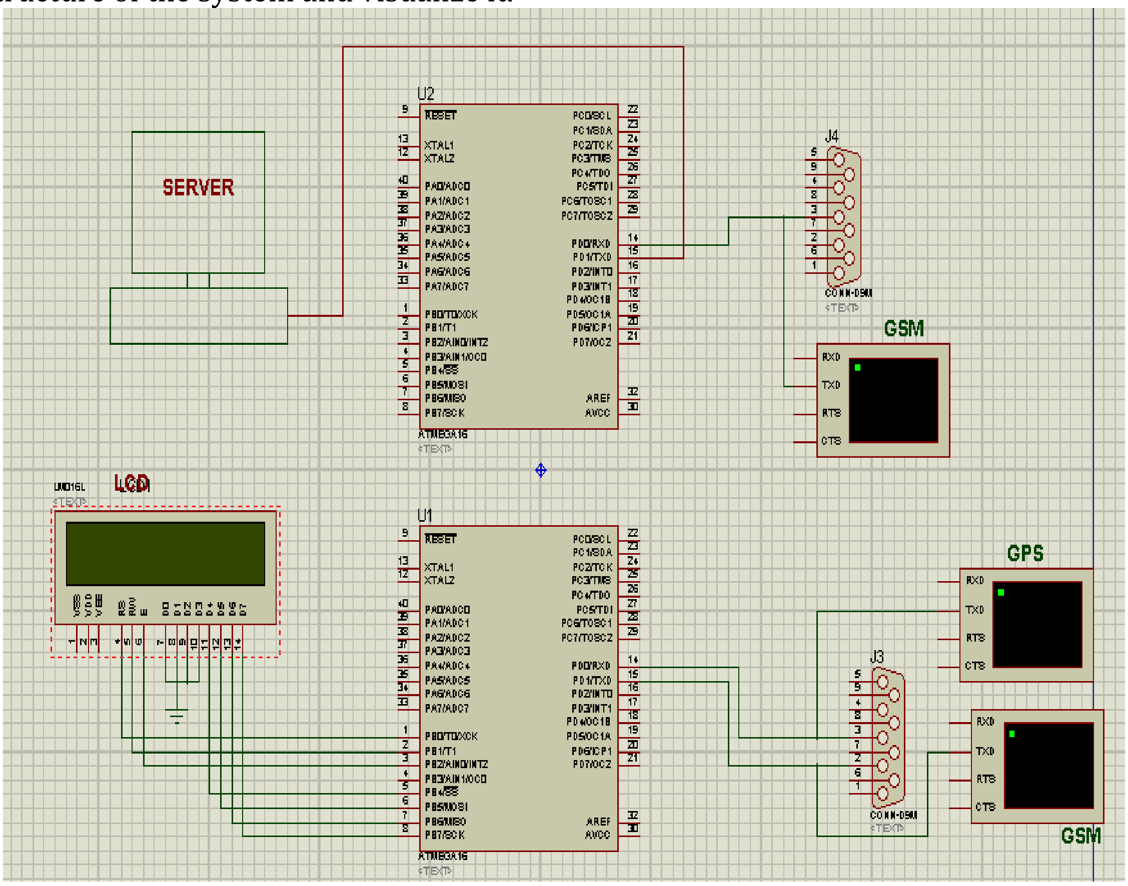


**Fig 4:** Circuit Diagram of vehicle tracking system

### 3.1 Description of Circuit diagram

**a) Atmega16**

Physically, a microcontroller is an integrated circuit with pins along each side. The pins presented by a microcontroller are used for power, ground, oscillator, I/O ports, interrupt request signals, reset and control. In contrast, the pins exposed by a microprocessor are most often memory bus signals (rather than I/O ports). In this project, the microcontroller performs the extraction of location information from GPS receiver string and sending the coordinates through the GSM module in the demand to the central server by SMS. The Microcontroller I used is ATMega16 from AVR Family.

**b) GSM**

The Global System for Mobile Communications (GSM) is the second-generation digital cellular mobile network. It is widely deployed around the world. Although improvements to GSM such as the next generation systems have been rolled out to cater for faster data centric traffic, backward compatibility to GSM is still maintained. Due to its wide availability, it is chosen as the medium for transfer of location information. The simple and inexpensive Short Message Service (SMS) allows users to send up to 160 characters. For the purpose of this project, the SMS is more than sufficient for sending the location information.[2]

Fig 1: GSM module

**c) GPS**

The main components of the vehicle tracking system are the GPS module, which is used to obtain the vehicle's coordinate and the GSM modem, which is used to transmit the location to the central server or it is also possible to user's phone through the mobile network. These components' hardware are the state-of-the-art Robomart GPS receiver module and Robomart AADD-SIMCOM-1.0 GSM module. A microcontroller, the AVR ATmega16 is also employed to control both modules and to provide an easily customizable platform for any required application. Generally, the system will provide the user with the vehicle's location coordinate upon request. It can also be programmed to send the location information periodically if required. These options allow the user flexibility of both on-demand or continuous tracking of the vehicle.

Fig 2: GPS Module

# 4. Experimental Environment

## 4.1 Screen shout of the platform

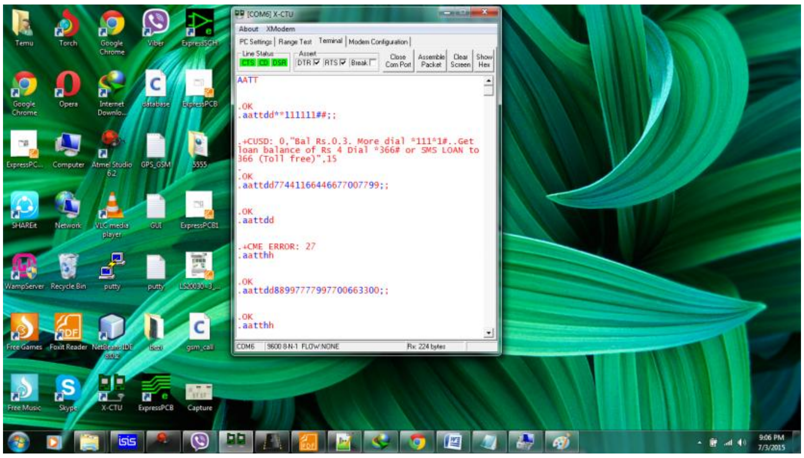

Figure 1: Testing GSM module terminal software

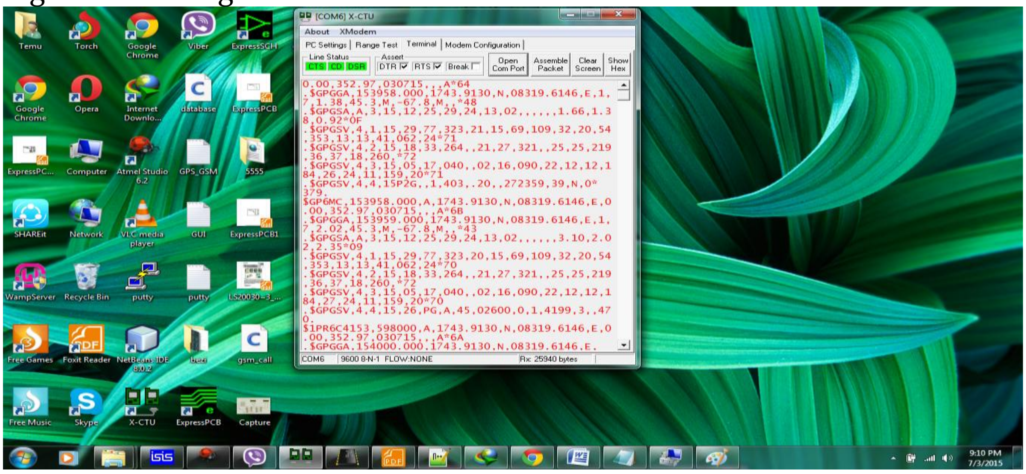

**Figure 2: The implementation of GPS on the breadboard.**

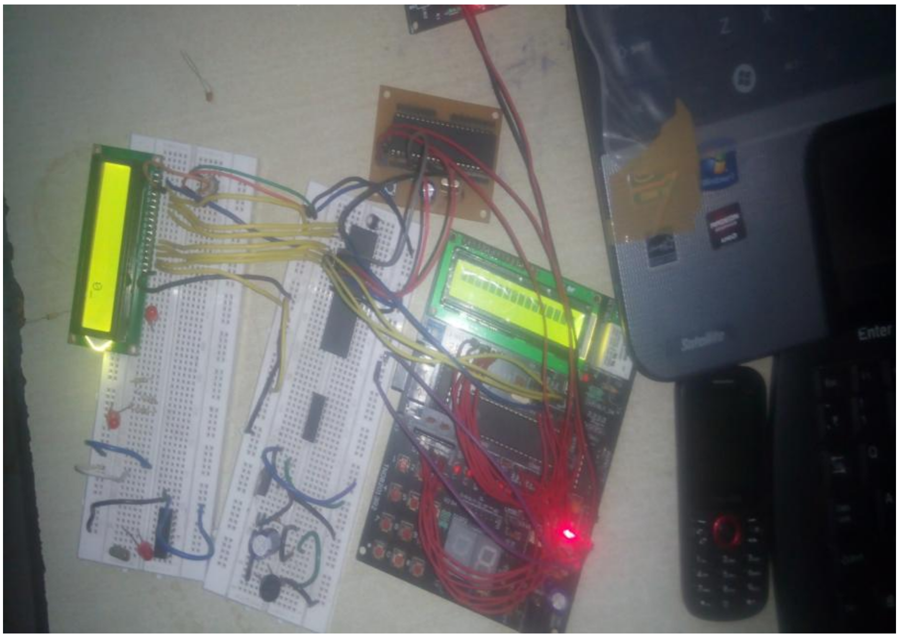

**Figure 3: Implementat ion of AVR atmega16 MCU**

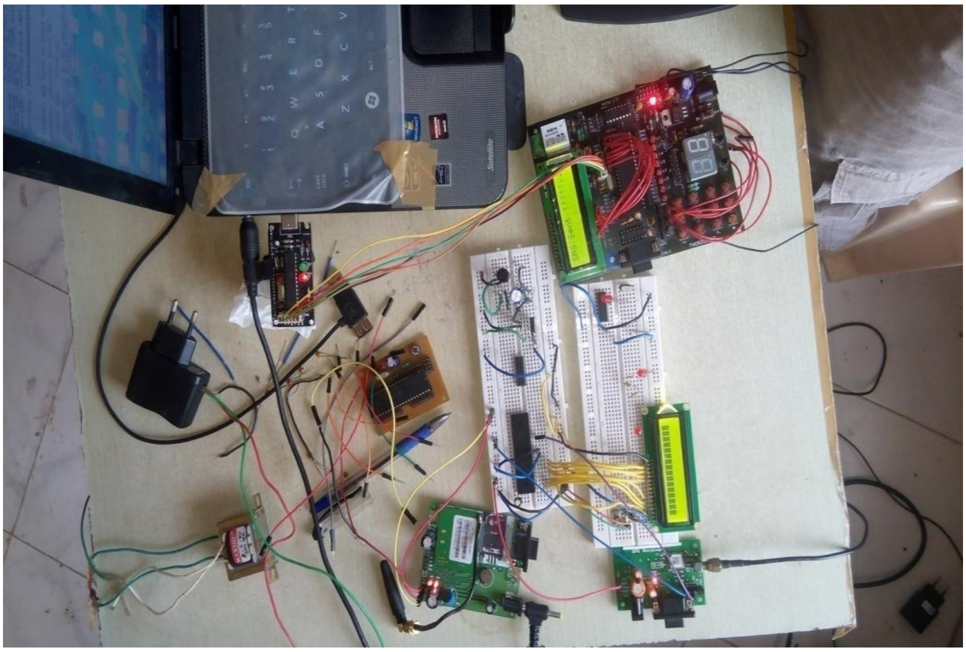

Figure 4: Circuit diagram the system connection practical

### 4.2 Software Implementation

In this project Atmel studio 6.2 for Embedded C programming for the microcontroller to handle the tasks performed inside the vehicle; Netbeans for programming java to develop database connection and user interface design on the server side. Again the java code receives the data from the serial port and stores on database. I used oracle database to store the information

#### 4.2.1 Reading Latitude, Longitude, and GMT time

The Latitude, Longitude and GMT time can be read from GPS module. The data can be communicated through LCD.

**Algorithm:**

Step 1: start
Step 2: declare three array variables lat[], logi[], tim[], lat1[], logi1[], tims1[], as character
Step 3: Read the GPS string and process it to get latitude, longitude and GMT
Step 4: Display the value on LCD.
Step 5: Read GPS data again after 2 seconds and put it in three different variables
Step 6: Calculate Distance between to locations
Step 7: Calculate the time difference between t1 and t2
Step 8: Find speed from distance/time formula
Step 5: if speed > max_allowed_speed
: Send the value to GSM to SMS it.
: Display warning message on LCD and the values
Else if
: Display the Location and time data on LCD
Step 10: Goto Step 3.

**Flowchart**

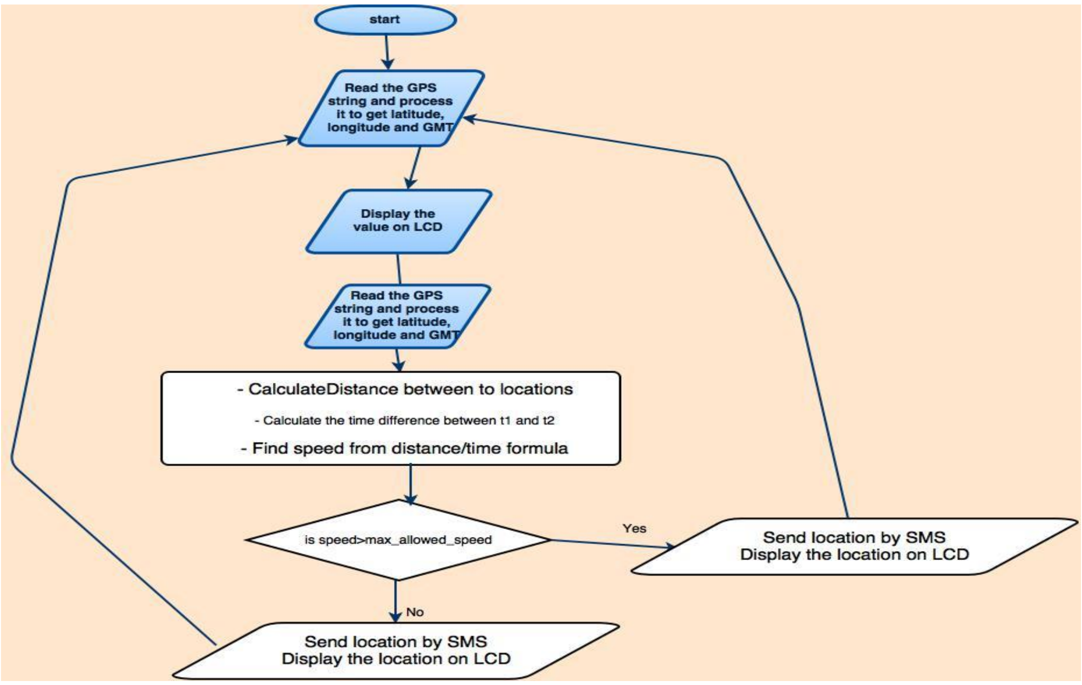


**Figure 5: Reading the location data from GPS and sending SMS**

**Coding:**

```
#define F_CPU 8000000ul
#include<avr/io.h>
#include"lcd.h"
#include<util/delay.h>
#define baud 9600ul
#define baud_prescale ((F_CPU/(16*baud))-1)
#define RS 0
#define EN 2
#define RW 1
#define lcd PORTB


void gsm_char(char data) {            //For sms
        while(!(UCSRA & (1<<UDRE)));
        UDR=data;
        _delay_ms(1);
}

void gsm_string_tx(char *str1) {            //For sms
        int i=0;
        while(str1[i]!='\0')   {
                while(!(UCSRA & (1<<UDRE)));
                UDR=str1[i];
                i++;
                _delay_ms(1);
        }   }
```

```
void usart_init() {
        UBRRL=baud_prescale;
        UBRRH=(baud_prescale>>8);
        UCSRC=(1<<URSEL)|(1<<UCSZ1)|(1<<UCSZ0);
        UCSRB=(1<<RXEN)|(1<<TXEN);
}
void usart_tx(char data) {
        while(!(UCSRA&(1<<UDRE)));
        UDR=data;
}
unsigned char usart_rx() {
        while(!(UCSRA&(1<<RXC)));
        return UDR;
}
void main () {
        DDRB=0xff;//lcd port
        DDRA=0xf0;
        DDRC=0xff;
        lcd_init();
        usart_init();
        dis_data(48);
        char x;
        char lat[10];
        char longi[10];
        int i=0, count=0;;
        while(1)        {
                x=usart_rx();
                if(x=='G')              {
                        x=usart_rx();
                        if(x=='G'){
                                x=usart_rx();
                                if(x=='A'){
                                        x=usart_rx();//for comma
                                        x=usart_rx();//for 0
                                        while(x!=','){
                                        x=usart_rx(); //First digit of latitude
                                        }
                                        x=usart_rx();
                                        while(x!=','){
                                                lat[i]=x;
                                                x=usart_rx();
                                                i++;
                                        }
                                        lat[i]='\0';
                                        x=usart_rx();//N
                                        x=usart_rx();//,
                                        x=usart_rx();//First char of latilitude
```

```
i=0;
while(x!=','){
        longi[i]=x;
        x=usart_rx();
        i++;
}
longi[i]='\0';
dis_cmd(0x80);
lcd_string(lat);
dis_cmd(0xc0);
lcd_string(longi);
i=0;
count++;
if(count==100)      {
        gsm_string_tx("AT");
        gsm_char(0x0D);
        gsm_string_tx("AT+CMGF=1");
        gsm_char(0x0D);
        gsm_string_tx("AT+CMGS=");
        gsm_char(0x22);
        gsm_string_tx("+251913350027");
        gsm_char(0x22);
        gsm_char(0x0D);
        gsm_string_tx(lat);
        gsm_string_tx("   ");
        gsm_string_tx(longi);
        _delay_ms(5);
        gsm_char(0x1A);
        PORTC=0xff;
        count=0;
}       }       }       }}}
```

**4.2.2 Receiving SMS and storing on Server**

Description

When SMS sent from remote vehicle server receives SMS from GSM module connected to the server and processes decision and stores on the database.

**Algorithm**:

Step 1: start
Step 2: wait (incoming SMS)
Step 3: Receive SMS to server
Step 4: check the ID number
Step 5: Store record to the server
Step 6: View location and the speed on server Interface to the Administrator
Step 7: go to step 2

**Flowchart**

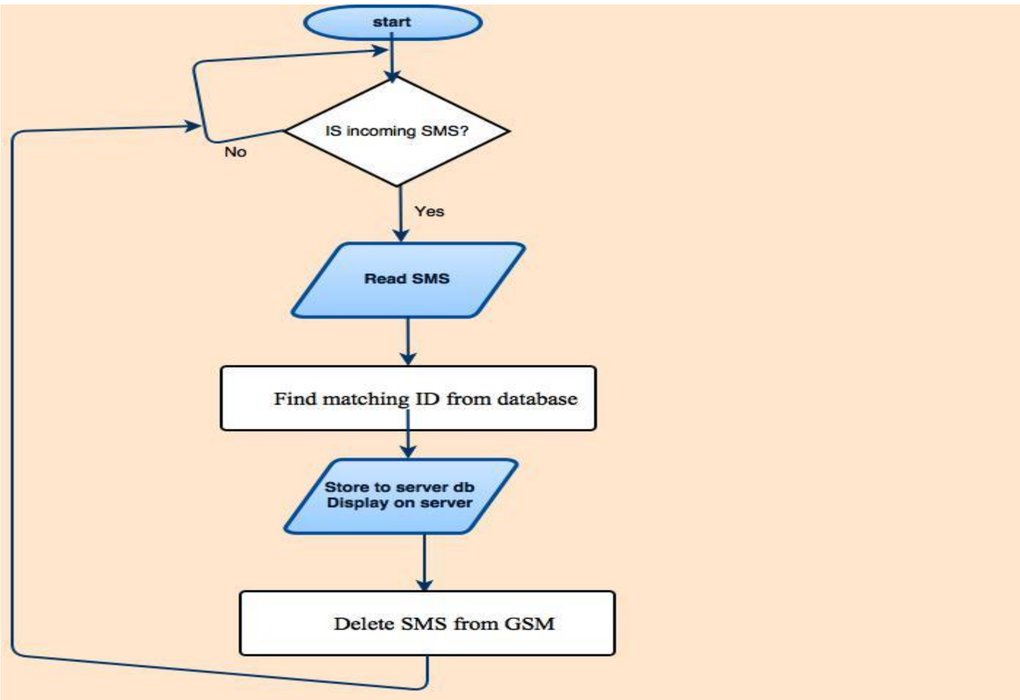


**Figure 6: Flowchart for receiving SMS**

**Coding**

```
char tt[100];
void sms_rx() {
	int i=0;
	gsm_string_tx("AT");
	_delay_ms(5);
	gsm_char(0x0D);
	_delay_ms(5);
	gsm_string_tx("AT+CMGF=1");
	_delay_ms(5);
	gsm_char(0x0D);
	_delay_ms(5);
	gsm_string_tx("AT+CMGR=1");
	_delay_ms(5);
	gsm_char(0x0D);
	_delay_ms(5);
	char val=usart_getch();
	tt[0]=val;
	i=1;
	while(val!='\0'){
		val = usart_getch();
		tt[i]=val;
		i++;
		dis_data('k');
	}
	tt[i]='\0';
```

```
        lcd_string(tt);
}
void main(){
        char value=0,str[15];//,lati_dir=0,longi_dir=0;;
        int i=0;
        DDRA=0xf0;
        DDRB=0xfF;
        lcd_init();
        usart_initial();
        while(1)        {
                        sms_rx();
                        dis_cmd(0x80);
                lcd_string("Receiving .... ");//message sending");
                        }
        }
}
```

**4.2.3 Database and Interfacing implementation**

**Sample Code:**

```
public class Db_Connector
{
        static Connection con;
        public static Connection getconnection()
        {
       try
                {
Class.forName("com.mysql.jdbc.Driver");
                        con =DriverManager.getConnection
("jdbc:mysql://localhost:3306/data_logger","root","");


                }
                catch(Exception e)
                {
                        System.out.println("class error    : "+ e);
                }
                return con;
        }
  }
```

**4.2.4 Data Viewer sample code**

```
  try {
         Connection con = new Db_Connector().getconnection();
        String sql = "select  * from trip_logger";
        ResultSet rs = con.createStatement().executeQuery(sql);
        ResultSet rs1;
         String sql1;
```

```
        int i=0;
        while(rs.next())        {
  sql1 = "select  * from owner_detail where owner_ID='" + rs.getString(8)+"'";
          rs1 = con.createStatement().executeQuery(sql1);
          rs1.next();
          jTable1.setValueAt(rs.getString(1),i,0 );
          jTable1.setValueAt(rs.getString(2),i,1 );
          jTable1.setValueAt(rs.getString(3),i,2 );
          jTable1.setValueAt(rs.getString(4),i,3 );
          jTable1.setValueAt(rs.getString(5),i,4 );
          jTable1.setValueAt(rs1.getString(2),i,6 );
          jTable1.setValueAt(rs.getString(7),i,5 );
          jTable1.setValueAt(rs.getString(1),i,7 );
          jTable1.setValueAt(rs.getString(1),i,8 );
          i++;
        }
        }catch (SQLException ex) {
        System.out.println("Error : " +ex);
      }
  }
```

**4.2.5 Data Receiving and sending in server side with java**

```
import java.io.IOException;
import java.io.InputStream;
 import java.io.OutputStream;
import gnu.io.*;

public class TwoWaySerialComm {
void connect( String portName ) throws Exception {
CommPortIdentifier portIdentifier = CommPortIdentifier.getPortIdentifier( portName );
if( portIdentifier.isCurrentlyOwned() ) {
System.out.println( "Error: Port is currently in use" );
} else {
int timeout = 2000;
CommPort commPort = portIdentifier.open( this.getClass().getName(), timeout );
if( commPort instanceof SerialPort ) {
SerialPort serialPort = ( SerialPort )commPort;
serialPort.setSerialPortParams( 57600,
SerialPort.DATABITS_8,
SerialPort.STOPBITS_1,
SerialPort.PARITY_NONE );
InputStream in = serialPort.getInputStream();
OutputStream out = serialPort.getOutputStream();
( new Thread( new SerialReader( in ) ) ).start();
( new Thread( new SerialWriter( out ) ) ).start();
} else {
System.out.println( "Error: Only serial ports are handled by this example." );
}  } }
public static class SerialReader implements Runnable {
```

```
InputStream in;
public SerialReader( InputStream in ) { this.in = in;}
public void run() {
byte[] buffer = new byte[ 1024 ];
int len = -1;
try {
while( ( len = this.in.read( buffer ) ) > -1 ) { System.out.print( new String( buffer, 0, len ) ); }
} catch( IOException e ) { e.printStackTrace(); }
  } }
public static class SerialWriter implements Runnable {  OutputStream out; }
public SerialWriter( OutputStream out ) {   this.out = out; }
public void run() {
try {
int c = 0;
while( ( c = System.in.read() ) > -1 ) { this.out.write( c ); }
} catch( IOException e ) {
e.printStackTrace();
} } }
public static void main( String[] args ) {
try {
( new TwoWaySerialComm() ).connect( "COM5" );
} catch( Exception e ) {
e.printStackTrace();
}  }}
```

### 4.2.6 The results on the server Side

These screenshots are taken from the server side .

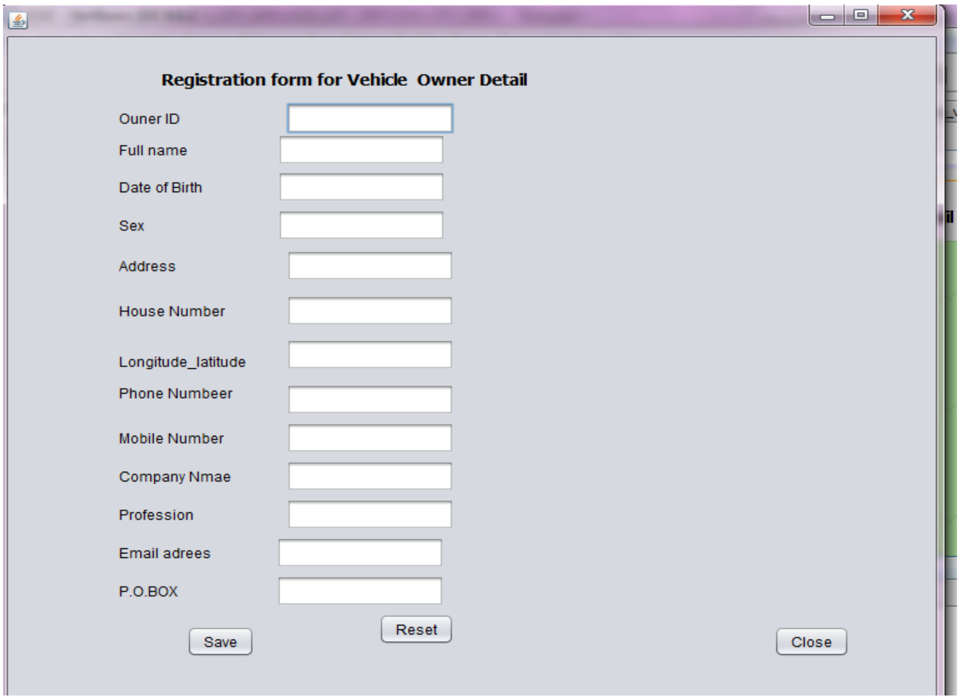


**Figure 7: The vehicle Owner registration form**

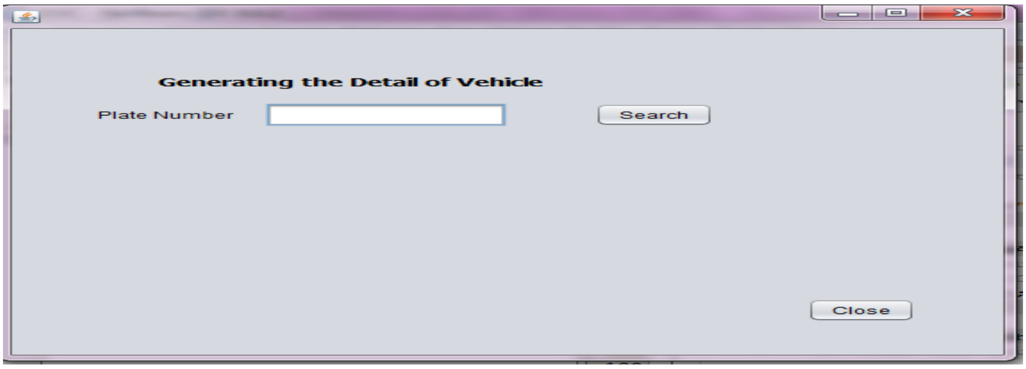


**Figure 8: Vehicle detail searching**

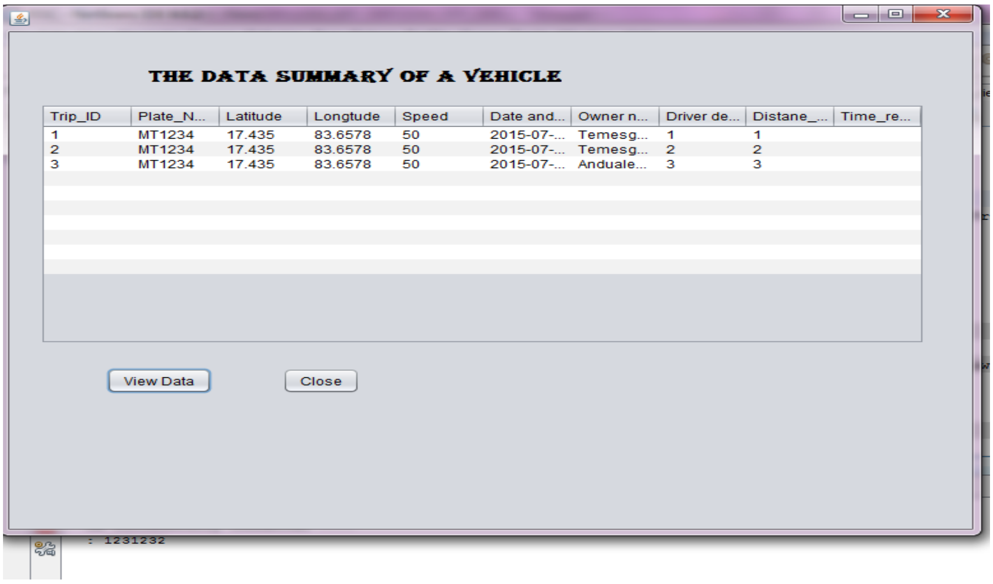


**Figure 9: Summary viewing Interface**

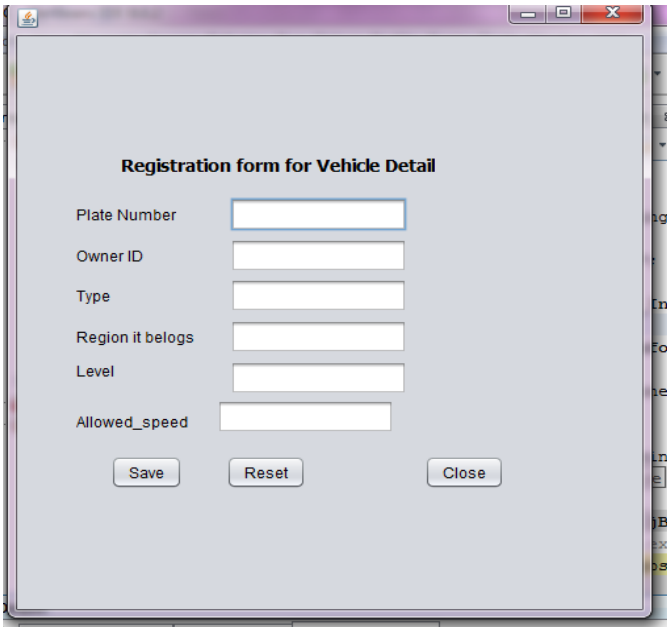


**Figure 10: The registration form for vehicle Detail**

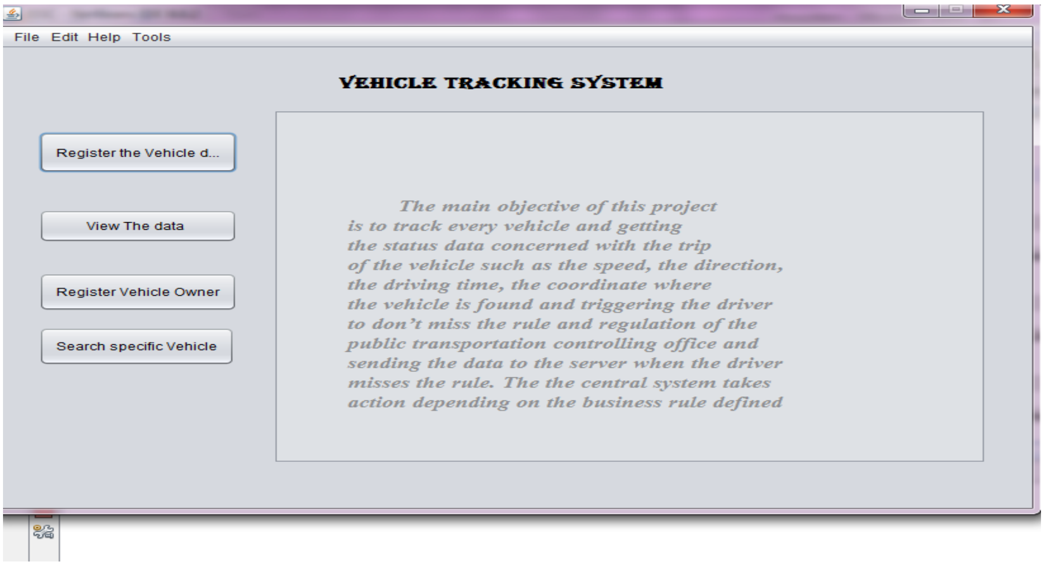


Figure 11: Fig: The main Interface to operate data on server

## Conclusion

The development of a vehicle tracking system's hardware unit has been presented and implemented in this project. The system is able to obtain a vehicle's GPS coordinate, the GMT and Direction and transmit it using the GSM modem to the central server. The developed vehicle tracking system demonstrates the feasibility of near real-time tracking of vehicles, which can be used for preventing accident in public transportation systems. It also used to track company vehicles for whether are driving as rule or not, fleet management and many other applications. The system can provide improved customizability, global operability and cost when compared to existing solutions. For future work, further design on multilayer PCB can be undertaken to reduce the wires on the unit module. The reliability of the system can be improved and additional features can also be added. Once the system is complete, the vehicle tracking system has the potential to be commercialized as a standalone product for tracking and processing data since its utility is quite popular.

## Reference

[1] D. Sperling and D. Gordon, Two Billion Cars: Driving Towards Sustainability, New York, NY: Oxford University Press, 2009.

[2] Youjing Cui; Shuzhi Sam Ge, "Autonomous vehicle positioning with GPS in urban canyon environments," IEEE Transactions on Robotics and Automation, vo1.l9, no.l, pp.15-25, Feb 2003.

[3] Rocky Mountain Tracking. (2013). Rocky Mountain Tracking: Main Page [Online]. Available http://www.rmtracking.coml

[4] LoJack. (2013). LoJack: Main Page [Online]. Available http: //www.lojack.coml

[5] H. Song, S. Zhu and G. Cao, "SVATS: A Sensor-Network-Based Vehicle Anti-Theft System," INFOCOM 2008. The 27th IEEE Conference on Computer Communications, pp. 13-18, April 2008.

[6] A. EI-Rabbany, Introduction to GPS: The Global Positioning System, Norwood, MA: Artech House, 2006.

[7] D. K. Elliott and H. Christopher, Understanding GPS Principles and Application, Norwood, MA: Artech House, 2006.

[8] G. Gu and G. Peng, "The survey of GSM wireless communication system," in Proc. International Conference on Computer and Information Application (ICCIA), pp.121,124, Dec. 2010.
[9] AT Commands Examples: Examples for Robomart GSM Wireless Modules, Robomart GSM, 2011.
[10]. Atmega16 microcontroller Manual or Datasheet, 8209DS–AVR–11/10
[11]. Car-Animal Accidents Raise Concern By Constance 1.Hays. Published: January 02, 1993. Available: http://www.nytimes.comlI993/01/02/nyregion/car-animal-accidents-raise-concern.html

1. A. EI-Rabbany, Introduction to GPS: The Global Positioning System, Norwood, MA: Artech House, 2006.

2. **Internet sources**

a) Car-Animal Accidents Raise Concern By Constance 1. Hays. Published: January 02, 1993. Available: http://www.nytimes.comlI993/0 1/02/nyregion/car-animal-accidents-raise-concern.html
b) Missing children in India, "Child protection and child rights," Available: http://www.childlineindia.org.in/missing-children-india.htm
c) Http://www.adafruit.com/arduino
d) http://www.atmel.com
e) http://www.engineersgarage.com
f) http://www.wiring.org.co/
g) http://www.cmtinc.com/gpsbook/
h) http://www.circuitstoday.com/lcd-interfacing-with-avr
i) http://www.developershome.com/sms/sms tutorial

**Cite this article:**

**Mr. Temesgen Mengistu Helana (2020).** Embedded system for monitoring and controlling traffic accident in public transport. ***Journal of Scientific Reports,*** 2(1), 1-16. doi: https://doi.org/10.5281/zenodo.3727640

Retrieved from http://ijsab.com/wp-content/uploads/1006.pdf

# Published by

IJSAB International