GPS based Passive Vehicle Tracking Using Arduino

Swarna Bai Arniker
Research Centre Imarat
16 PUBLICATIONS 99 CITATIONS

GPS Based Passive Vehicle Tracking Using Arduino

S.Madhuri[1], K.Bhuvana Jyothi[1], Ch.Indraja[1]
[1]Vignana Bharathi Institute of Technology (VBIT)
Ghatkesar, Ranga Reddy District, Telangana state, India

Swarna Bai Arniker[2], K.Sita Rama Rao[2]
[2]Directorate of Reliability & Quality Assurance
Research Centre Imarat, Vignyanakancha-500069
Hyderabad, Telangana State, India
arniswarna@rcilab.in

***Abstract*—In any organization the facilities manager (FM) maintains the log book of the vehicles being used for transport of men and machinery. The goal of FM is not only to provide logistic support but also to minimize the misuse of vehicles by the concerned persons. Presently logbooks are being used to track the usage of vehicles. In this paper we present the hardware and test results of a Passive Vehicle Tracking system based on GPS technology, Arduino embedded system, Arduino IDE open source software, SD card, Matlab and Google Maps API. This vehicle tracking system may be installed in all official vehicles. This has applications in cutting down fuel expenses and to reduce misuse of official vehicles. Further this system also offers easy availability, compact, low power consumption, and cost effective solution.**

Keywords-Vehicle tracking; Arduino microcontroller; GPS module; Matlab; Google map.

I. INTRODUCTION (HEADING 1)

A Vehicle tracking system (VTS) integrates the use of automatic vehicle location in each vehicle with software that provide vehicle location based on GPS technology. VTS is widely used for urban public transport, logistics and transportation industry, hospitality industry, food delivery vans, and car rental companies. GPS based automatic vehicle tracking using RFID is discussed in [1]. Paper [2] presents on vehicle tracking and locking system based on GSM and GPS. Vehicle location and tracking system using GSM and GPS is discussed in [3]. Intelligent Anti-theft and tracking system for automobiles is presented in [4] using GPS and GSM. Advanced vehicle tracking system on Google erath using GPS and GSM is put forward in [5]. Ref. [1-5] discuss about active and real time Automatic vehicle location based on GPS and GSM technologies. This fact was the motivation to write this paper to use only GPS and low cost devices for passive vehicle tracking which would aid the facilities manager of a company to track the path travelled by the vehicle, reduce fuel cost, optimize misuse of official vehicles.

The focus of this paper is hardware realization interfacing, programming and demonstration of passive vehicle tracking system using GPS technology; Arduino embedded system [6] and open source software Arduino IDE. This is a undergraduate student project based on Arduino Uno Atmega 328 based board as shown in Fig.1 and its interfacing with GPS module UR-109 and SD card LC Studio. Arduino is single-board microcontroller, of credit card size intended to make the application of synergistic objects or domain more accessible. Arduino is an open-source easy-to-use hardware board along with flexible software designed around a 32-bit Atmel AVR Processor. Pre-programmed into the on-board microcontroller chip is a boot loader that allows uploading programs into the microcontroller memory from PC without needing a chip (device) programmer using USB cable.

GPS module provides data which includes time, date, latitude, longitude, altitude, speed etc. and this GPS data will be read by the Arduino board and stored in a SD card interfaced to Arduino board in a particular file format. Later in a PC the GPS data recorded is opened in the designed and developed Matlab based GUI for processing and display in the form of maps using Google Maps API. In this system vehicle path is tracked using GPS technology in real time and this saved GPS data is processed in Matlab offline and displayed in the form of Google maps using API also in offline. This passive vehicle tracking system may be installed in all official vehicles in an organization to track the route followed by the drivers and it acts like a visual logbook for vehicles to aid facility manager to monitor proper utilization of official vehicles and estimating the payment to cab services. Section 1 gives a introduction to vehicle tracking system and related work. Section 2 describes the proposed system and its building blocks. Hardware realization and interfacing is discussed in Section 3. Proposed system arrangement and System modules are in section 4 & 5 and Software part is presented in section 6 and test results are discussed in section 7 followed by conclusions.

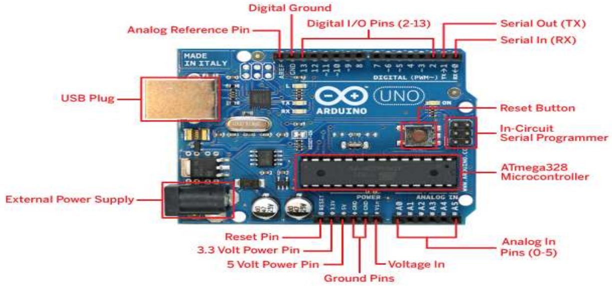


Figure 1. Arduino Uno Board [7]

II. PROPOSED SYSTEM

A. Major building blocks to realize this GPS based Passive Vehicle Tracking System

The various hardware components required to realize this system are Arduino Uno, GPS module, SD card, and USB-cable as shown in Fig. 2 and their description are presented and discussed in the subsequent sections.

B. Arduino Uno

The Arduino Uno uses a low power 8-bit AVR microcontroller board based on the ATmega328 with flash memory. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an In-Circuit Serial Programming (ICSP) header, and a reset button. It contains all the features needed to support the microcontroller; simply plug it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to start working.

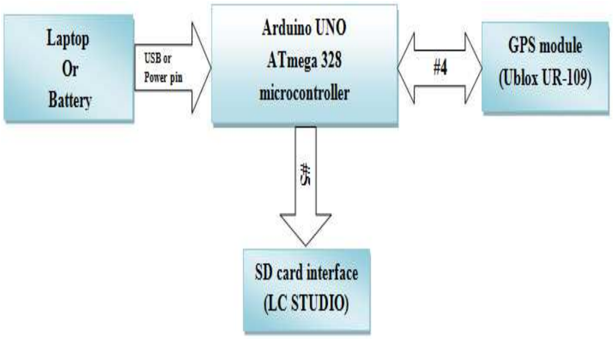


Figure 2. Block Diagram of GPS based Passive Vehicle Tracking uisng Arduino (Proposed)

C. GPS Module

GPS module used in this project is Ublox UR-109 module shown in Fig. 3. It is made of high performance and lower power chipset consisting of 50 channels. Its dedicated acquisition engine enables it to find and locate the satellites instantly. It has ultra high sensitivity of -160dBm. High sensitivity GPS receivers indicate the ability of the receiver to track the GPS signals quickly. For that it uses large bank of correlators and digital signal processing .

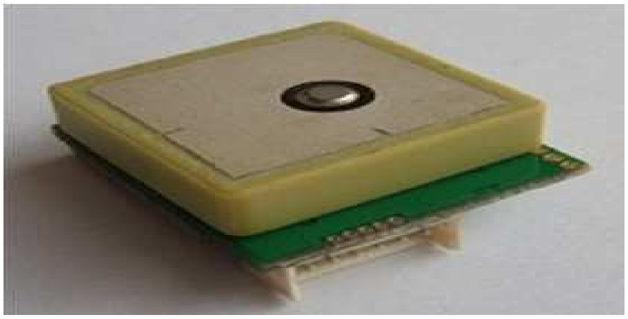


Figure 3. GPS module Ublox UR-109 [8]

It uses an NMEA protocol to track the GPS data. This is an ASCII based protocol which usually starts with the "$" symbol with carriage return/line feed [8].

D. SD Card

GPS data collected is stored on to the SD card shown in Fig. 4. So SD card is placed into the SD slot and the data received by the GPS receiver is stored onto the SD card by Arduino Uno board..

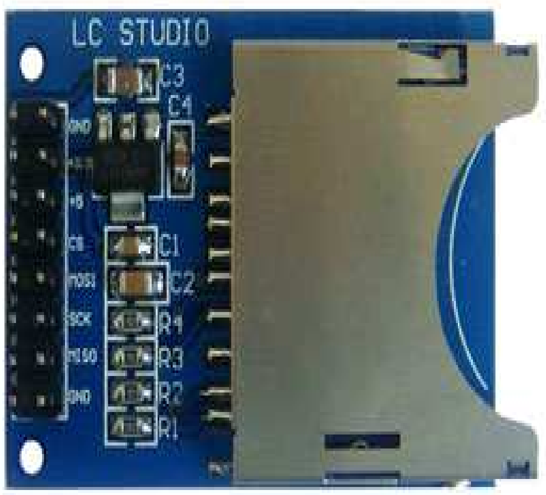


Figure 4. SD card LC Studio [9]

E. USB Cable

This is a Standard USB 2.0 cable which is used to connect Arduino Uno or any board with the USB female-A port of the computer. Programs can be uploaded into the microcontroller memory from PC via USB cable. During testing, Arduino will get power from PC through USB. During actual use, it can get from 9V Battery or an adaptor.

III. HARDWARE REALIZATION AND INTERFACING

The Hardware realization of our proposed comprises of interfacing Arduino Uno with GPS module followed by interfacing DS card with Arduino Uno.

A. Interfacing Arduino Uno with GPS Module

The receive/transmit pins of GPS module are connected to the transmit/receive pins of Arduino Uno for the transmission and reception to take place followed by Vcc (3.3V) and Gnd pins as shown in Fig. 5.

B. Interfacing Arduino Uno with Sd Card

The Hardware realization of our system continued by using SPI interface in which 10,11,12,13 pins of Arduino Uno board are connected to the pins CS, MOSI, MISO and SCK pins of SD interface as shown in Fig. 5.

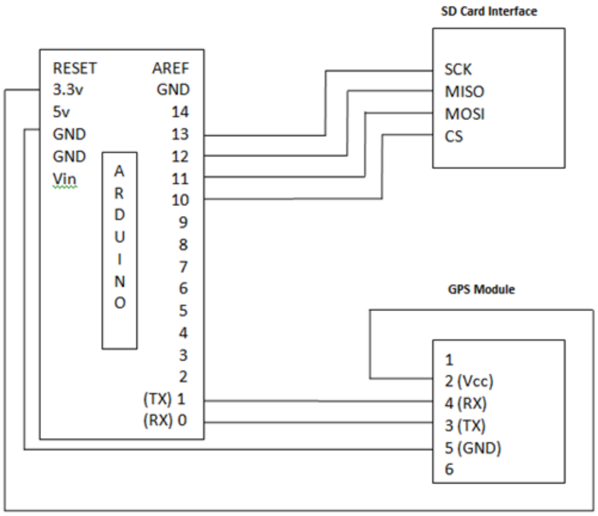


Figure 5. Interfacing arduino Uno with GPS module and SD card

IV. PROPOSED SYSTEM ARRANAGEMENT

The complete setup consists of three components- Arduino UNO, GPS module and SD interface. The main aim of this whole set up is to collect the GPS data which includes time, date, latitude, longitude, altitude, speed etc and store the obtained data in the SD card placed in the SD card slot. Initially the GPS module receives the data from the satellites based on 3D trilateration principle to locate a point on the earth in NMEA protocol.

The received data from the GPS receiver is transmitted to Arduino UNO board consisting of ATmega328 controller. This data is stored into the SD card in a .CSV file format. The SD card is removed from the SD card interface and the GPS data recorded is opened in the Matlab which is processed and is displayed in the form of maps using Google Maps API.

To make it user friendly Matlab GUI is designed in which the user can browse the required file and extract the required data and map it on maps using coordinates. Using latitude and longitude values, a point is located on earth and then latitude and longitude values of the complete route are plotted using Google Maps API.

V. SYSTEM MODULES

The complete work is divided into three modules

A. Data Collection

The GPS data of the path travelled by the vehicle is collected using a GPS receiver. The receiver collects the data from the satellites. Here it uses the principle of trilateration and locates the point on the earth. For every 3 seconds, it records the data from the satellites whose fix is obtained [10].The data collected follows NMEA protocol.

B. Data Storage

The GPS data collected is stored onto an SD card. A CSV file is generated which stores the data in rows and column format. Each of the data separated by commas is stored in a particular format in cells. It can be used for the future analysis of data.

C. Displaying GPS Data on Google Maps using Matlab

The data stored on the SD data is later transferred to the computer. It is then processed using matlab and then in Matlab using Google maps Api the data is viewed on maps. To make it more user friendly, Matlab uses GUI to display the path travelled by the vehicle by just a click on the mouse

VI. SOFTWARE IMPLEMENTATION

GPS based passive vehicle tracking using Arduino is realized by us is shown in Fig. 6. The open-source Arduino environment [12] makes it flexible to write code and upload it to the i/o board. In this application Arduino IDE software version-1.0.5 is used.

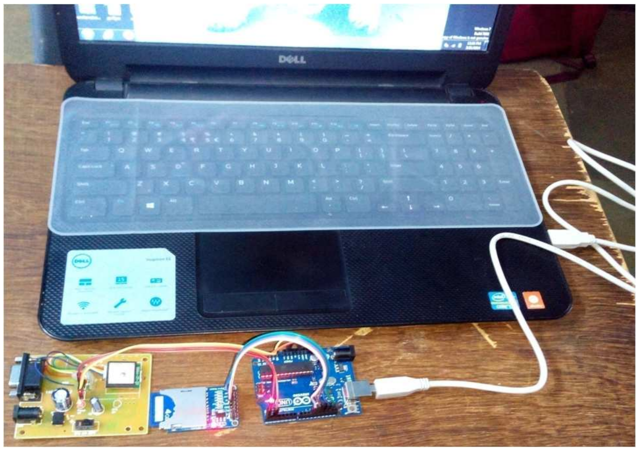


Figure 6. GPS based Passive Vehicle Tracking using Arduino (Realized)

A. Library Functions used

Arduino uses library functions similar to those used in other programming languages. There are few predefined functions which are available directly with the Arduino IDE and few can be downloaded online [11].

This module used TinyGPS++, MemoryFree, SDFat libraries.

TinyGPS++ is a library function which is mainly required to extract the GPS data which includes position, date, time, altitude, speed, course etc from GPS receivers.It provides easy-to-use method to extract the data. It is mainly required for the syntactical analysis of NMEA data streams [12].

MemoryFree library enables to know and keep the track of the amount of memory available at run time. This is mainly required while parsing codes and to solve memory related problems at run time [13].

SDFat library is mainly used to read and write data to SD cards. It supports creation, deletion, truncation of files and also supports access to subdirectories. It supports FAT16 and FAT32 file systems on SD cards [14].

B. System Flow Diagram

Fig.7 shows the process of the complete module immediately after the power is supplied to it.

When Arduino Uno board is powered on with 9 volts battery, the SD card and GPS module gets initialized. If there is any initialization error then a message “SD initialization error” is displayed on the serial monitor window of Arduino IDE. The GPS module starts locating the satellites, it takes some time to obtain a fix point.If the characters processed are less than 10, then no data is received by the GPS receiver and it display “No data” on the serial monitor window. If it is more than that then data is being received. But based on the flash timer count, it is possible to say if fix is obtained. If the flash timer count is greater than 30000, then fix is obtained. After that, the .csv file is opened and the data is recorded into it. Here based on comma, the data is segregated into cells and is displayed into corresponding cells. The process is ended here. The SD card is then removed from the SD slot.

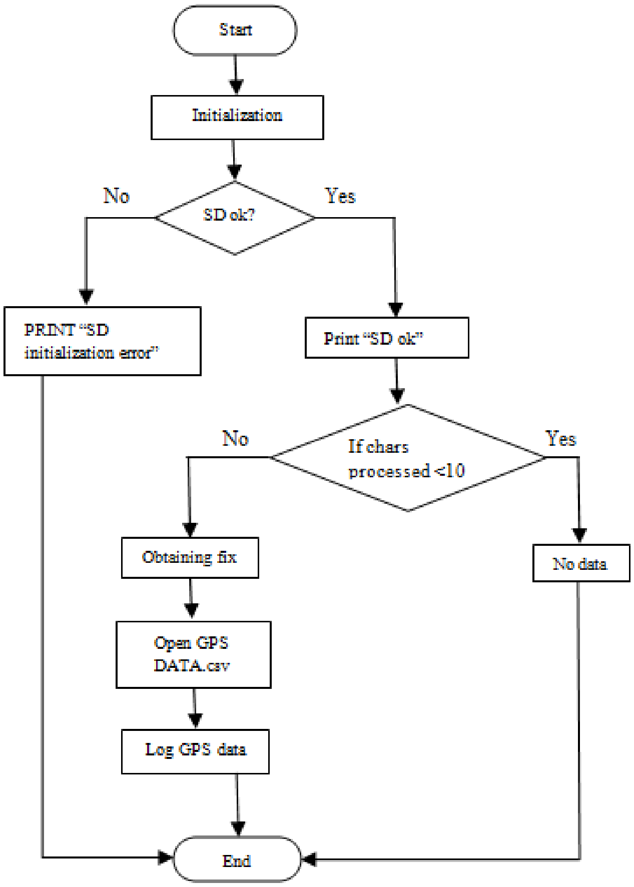


Figure 7. Flowchart of the complete process

Fig. 8 shows the process after the GPS data is recorded and stored in the system. Internet should be connected only then the Google maps view can be seen in Matlab. Matlab software is opened and in the command window the file name of the designed GUI is given. The designed GUI opens which contains a bar to browse the GPS data and space to display the map as shown in Fig. 9. The .csv file is browsed and the route is thus displayed as on Google maps with latitude and longitude coordinates

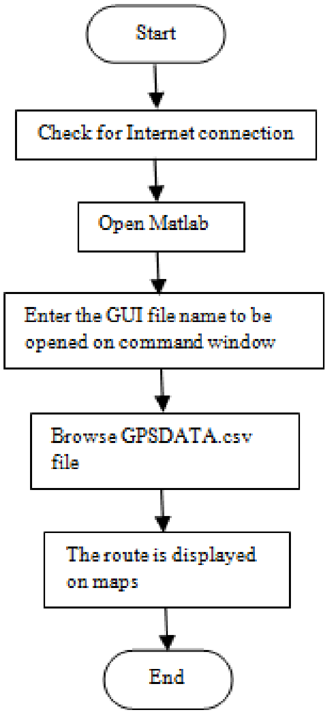
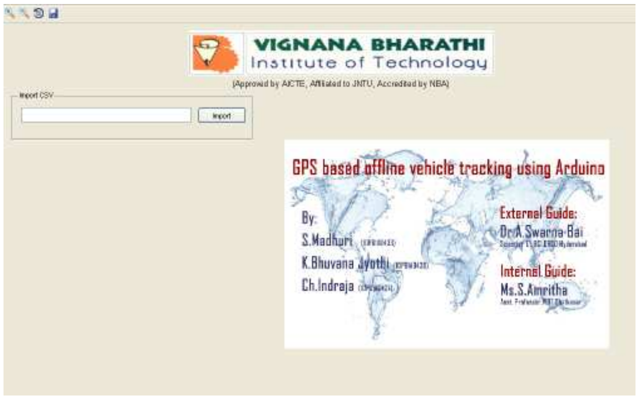

Start
Check for Internet connection
Open Matlab
Enter the GUI file name to be opened on command window
Browse GPSDATA.csv file
The route is displayed on maps
End

Figure 8. Flowchart of the process in Matlab



VIGNANA BHARATHI
Institute of Technology
(Approved by AICTE, Affiliated to JNTU, Accredited by NBA)

GPS based offline vehicle tracking using Arduino

By:
S.Madhuri
K.Bhuvana Jyothi
Ch.Indraja

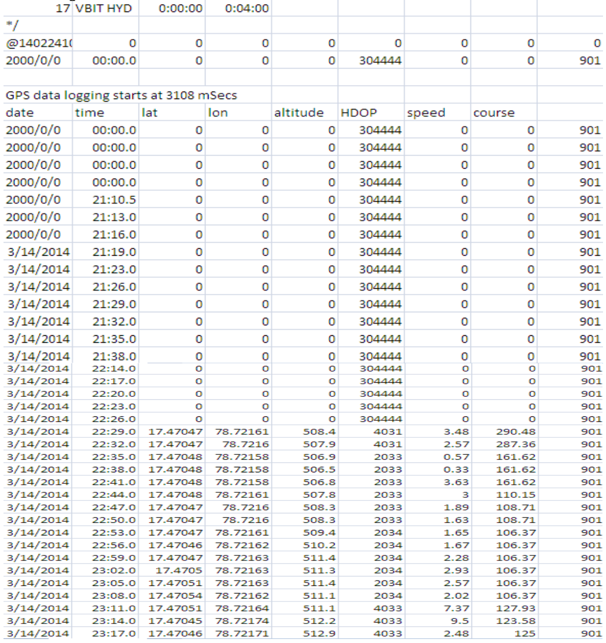
External Guide:
Dr.A.Swarna Bai

Internal Guide:
Ms.S.Amritha

Figure 9. Designed GUI

VII. Test Results

The experiment is performed in a college and the path travelled by the vehicle is recorded and the latitude and longitude values are plotted as shown in the figures below. Fig.10 shows the recorded GPS data in Excel sheet. Here the data is recorded every 3 seconds. Further Google map views comprises of Satellite view and Road map view of the route travelled. Fig.11, Fig.13 give Road map view and Fig.12, Fig. 14 give Satellite view of two different routes.



17	VBIT HYD	0:00:00	0:04:00					
*/								
@1402241(	0	0	0	0	0	0	0	0
2000/0/0	00:00.0	0	0	0	304444	0	0	901
GPS data logging starts at 3108 mSecs								
date	time	lat	lon	altitude	HDOP	speed	course	
2000/0/0	00:00.0	0	0	0	304444	0	0	901
2000/0/0	00:00.0	0	0	0	304444	0	0	901
2000/0/0	00:00.0	0	0	0	304444	0	0	901
2000/0/0	00:00.0	0	0	0	304444	0	0	901
2000/0/0	21:10.5	0	0	0	304444	0	0	901
2000/0/0	21:13.0	0	0	0	304444	0	0	901
2000/0/0	21:16.0	0	0	0	304444	0	0	901
3/14/2014	21:19.0	0	0	0	304444	0	0	901
3/14/2014	21:23.0	0	0	0	304444	0	0	901
3/14/2014	21:26.0	0	0	0	304444	0	0	901
3/14/2014	21:29.0	0	0	0	304444	0	0	901
3/14/2014	21:32.0	0	0	0	304444	0	0	901
3/14/2014	21:35.0	0	0	0	304444	0	0	901
3/14/2014	21:38.0	0	0	0	304444	0	0	901
3/14/2014	22:14.0	0	0	0	304444	0	0	901
3/14/2014	22:17.0	0	0	0	304444	0	0	901
3/14/2014	22:20.0	0	0	0	304444	0	0	901
3/14/2014	22:23.0	0	0	0	304444	0	0	901
3/14/2014	22:26.0	0	0	0	304444	0	0	901
3/14/2014	22:29.0	17.47047	78.72161	508.4	4031	3.48	290.48	901
3/14/2014	22:32.0	17.47047	78.7216	507.9	4031	2.57	287.36	901
3/14/2014	22:35.0	17.47048	78.72158	506.9	2033	0.57	161.62	901
3/14/2014	22:38.0	17.47048	78.72158	506.5	2033	0.33	161.62	901
3/14/2014	22:41.0	17.47048	78.72158	506.8	2033	3.63	161.62	901
3/14/2014	22:44.0	17.47048	78.72161	507.8	2033	3	110.15	901
3/14/2014	22:47.0	17.47047	78.7216	508.3	2033	1.89	108.71	901
3/14/2014	22:50.0	17.47047	78.7216	508.3	2033	1.63	108.71	901
3/14/2014	22:53.0	17.47047	78.72161	509.4	2034	1.65	106.37	901
3/14/2014	22:56.0	17.47046	78.72162	510.2	2034	1.67	106.37	901
3/14/2014	22:59.0	17.47047	78.72163	511.4	2034	2.28	106.37	901
3/14/2014	23:02.0	17.4705	78.72163	511.3	2034	2.93	106.37	901
3/14/2014	23:05.0	17.47051	78.72163	511.4	2034	2.57	106.37	901
3/14/2014	23:08.0	17.47054	78.72162	511.1	2034	2.02	106.37	901
3/14/2014	23:11.0	17.47051	78.72164	511.1	4033	7.37	127.93	901
3/14/2014	23:14.0	17.47045	78.72174	512.2	4033	9.5	123.58	901
3/14/2014	23:17.0	17.47046	78.72171	512.9	4033	2.48	125	901

Figure 10. GPS data recorded in Excel Sheet

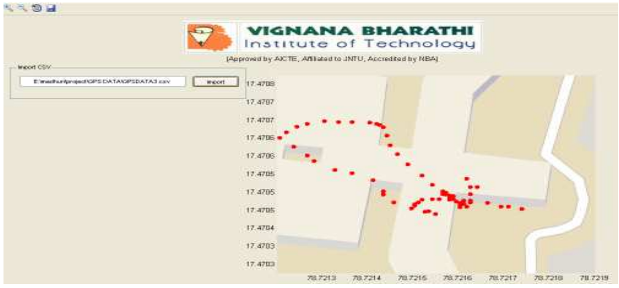


Figure 11. Map1 in roadmap view

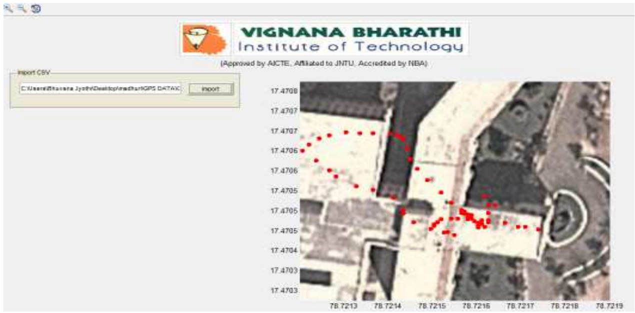


Figure 12. Map1 in satellite view

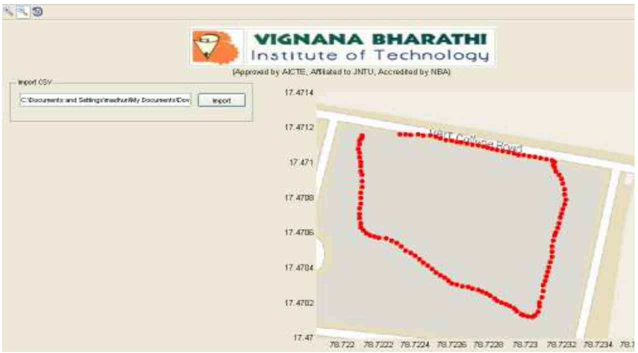


Figure 13. Map2 in roadmap view

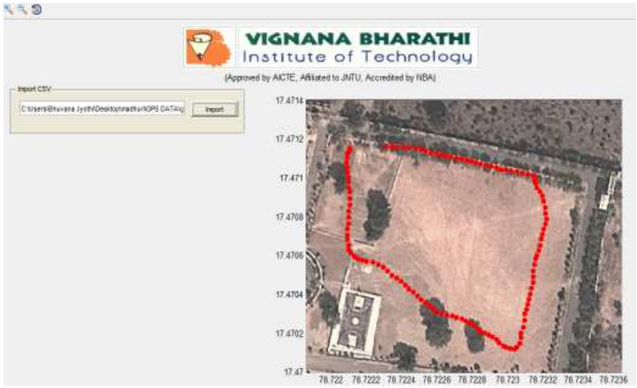


Figure 14. Map2 in satellite view

CONCLUSIONS

GPS based passive vehicle tracking using Arduino is developed, tested and demonstrated and it also displays the path traversed by the vehicle. The salient features of this product are less hardware components, less power consumption, and low cost. This system is mainly helpful for facility manager in an organization to monitor route travelled by the vehicle, reduce fuel cost, minimize misuse of official vehicles and it acts as a e-logbook. This system enables the user to be vigilant on his vehicles and also in commercial world the data obtained after tracking can be even used for analyzing the most opted routes. In military vehicles this play a major role as the vehicles which are used on specific mission can be tracked.

REFERENCES

[1] D. Bajaj and N. Gupta, "GPS Based Automatic Vehicle Tracking using RFID", *Interational Jornal of Engineering and Innovative Technology*, pp 31-35,Jan 2012.

[2] R.Ramani, S.Valarmathy, N.S.Vanitha, S.Selvaraju, M.Thiruppathi, R.Thangam,"Vehicle Tracking and Locking System based on GSM and GPS", *I.J. Intelligent Systems and applications*, pp. 86-93, August 2013.

[3] B.Kodavati, V.K.Raju, S.S.Rao, A.V.Prabhu, T.Appa Rao, Y.V.Narayana, "GSM and GPS based Vehicle Location and Tracking

System”, *International Journal of Engineering Research and Application*, pp. 616-625.

[4] M.N.Ramadan, M.A. Al-Khedher, S.A. Al-Kheder, “Intelligent Anti-Theft and Tracking System for Automobiles,” International Journal of Machine Learning and Computing, pp. 88-92, 2012.

[5] S.Kotte, H.M. Yanamadala, “Advanced Vehicle Tracking System on Google Erath uisng GPS and GSM”, *International Journal of Computer Trends and Technology*, pp. 130-133, 2013.

[6] http://arduino.cc

[7] http://arduinoarts.com/2011/08/the-arduino-uno-anatomy/

[8] GPS Engine board user manual

[9] http://apcmag.com/arduino-project-5-digital-audio-player.htm

[10] ”Trilateration”, [Online:] http://electronics.howstuffworks.com/gadgets/travel/gps2.htm

[11] “Arduino Microcontroller”,[Online:] http://www.arduino.cc/

[12] TinyGPS plus library, http://arduiniana.org/libraries/tinygpsplus/

[13] http://playground.arduino.cc/Code/AvailableMemory

[14] ”SDFat library”, [Online:] https://code.google.com/p/sdfatlib/