GPS-GPRS Control and Tracking System for Fuel Trucks via Processing of Travelling Information

Dr. Bayan M. Sabbar[1], Ayad I. Ali[2]

Abstract— A huge number of petrol institutions are suffering from thefts to their trucks or even cargo of petrol carried during transportations; these thefts were reported every year. Consequently, an efficient control and tracking system is needed to improve the effectiveness for these institutions. The objective of this project is to combine Global Positioning System (GPS) with Global System for Mobile Communication (GSM) technologies and to transfer the information via General Packet Radio Service (GPRS) supported by GSM; to construct an integrated control and tracking system for vehicles. This paper addresses the design and implementation of an integrated real-time and cost effective GPS-GPRS vehicle control and tracking system, this system is divided into three parts, which are the vehicle tracking part, fuel monitoring part, and control part. For vehicle tracking part, the system is equipped with Arduino UNO R3 microcontroller and SIM808 module that embeds both GPS receiver and GSM modem for navigation purpose. For fuel monitoring part, the system is equipped with fuel variable-resistor sensor installed within the vehicle tank to monitor and detect any sudden change of the fuel level which may indicate to a fuel theft occurred. The third part provides control procedures applied remotely on vehicles as remote stop/start of vehicle's engine by using fuel pump cut-off relay interfaced with the microcontroller. GPS receiver is utilized to determine vehicle present location via the GPS coordinates received from GPS satellites such as longitude, latitude and heading of movement with accuracy range from 2.5-20 meters, all data is transmitted to a remote server via TCP connection through GPRS service and subsequently transmitted to clients via Internet, and then becomes available to be displayed on Google Map embedded in the web application. The system prototype that updates GPS coordinates every 1 second and based on angle change of movement; was successfully developed.

Index Terms— GPS, GPRS, Vehicle Tracking, Controlling, Arduino UNO R3, SIM808, Variable Resistor Fuel Sensor, Relay.

———————————— ◆ ————————————

1 INTRODUCTION

THE GPS-GPRS tracking system is the system which makes use of GPS technology to determine the accurate geographical locations of vehicles at any time with the distance travelled from the Global Positioning Satellites through the GPS receiver device that is attached to the vehicle.

GPRS and GPS tracking system will provide real-time and efficient vehicle location information [1-2].

This system utilizes a module consists of GPS receiver and GSM modem interfaced to the Microcontroller; the device attached to the vehicle to be tracked. In order to monitor the vehicle movement; Google Maps is used for mapping the locations. The GSM modem fetches the GPS location and sends it to the server using GPRS. The system is intended for serving enterprises with huge number of vehicles and complicated requirements of usage; it is shows how such innovative technologies of GPS and GSM-GPRS that can be integrated interactively for addressing a real global problem [3]. The integration of GPS and GSM technologies was first founded via utilizing SMS as a technique of transmitting the GPS coordinates [4]. The enhancement of GPRS technology to GSM for transmitting location coordinates to a remote server enables the remote tracking of objects via using any computer connected to the Internet.

2 GLOBAL POSITIONNING SYSTEM

GPS is space-based satellite navigation system which provides geographical location and time information at all weather circumstances, anywhere over or nearby the Earth in case of the existence of an unobstructed line of sight to three or more GPS satellites.

GPS technology is described in terms of three segments:

1. Space Segment: Comprises of twenty-eight GPS satellites orbiting the globe on six orbital planes at height of 20,180 Km above the earth.
2. Control Segment: Comprises of 5 ground stations distributed around the world used to manage the operational health of GPS satellites via transmitting clock updates and orbital corrections.
3. User Segment: Comprises of different types of GPS receivers that can differ in sophistication and complexity.

GPS receiver is capable to identify its location when three GPS satellites triangulate, and to compute the distance to the GPS satellites. A fourth GPS satellite computes the time to the receiver. The information given from all four GPS satellites is compiled for determining the geographical position. The GPS receivers' sophistication impacts the accuracy and reliability of GPS data received [5]

3 GENERAL PACKET RADIO SERVICE

General Packet Radio Service GPRS technology is a packet-switched data service based on GSM infrastructure that is extensively used for voice services, GSM is not suitable for non-voice services due to its circuit-switched capability. GPRS is a 2.5 G cellular network. GPRS provides fast and affordable Internet connections to the GPRS users. This allows the users to still using GSM network to utilize voice services, but if data

transmission is needed; this will be passed via GPRS system [6]. Billing and charging is depends on the amount of data transmitted instead of the connection time. This is accomplished through allocating network's resources radio timeslot and bandwidth to GPRS users only when they require transmitting data, where the physical channel is established dynamically, this makes a considerably an efficient utilization of the available radio bandwidth.

GPRS may provide data rates up to 171.2 kbps. Whenever GPRS packets leave the GSM-GPRS network; the packets are transferred to TCP-IP network such as Internet or X.25, therefore, GPRS offers new signaling and transmission procedures in addition to new protocols, to interwork with the IP world and other standard packet networks [7].

4 VEHICLE CONTROL AND TRACKING SYSTEM

The control and tracking system is a web-based monitoring system designed for tracking any type of vehicles (i.e., fuel trucks). A combination of multiple modern communications and information technologies is utilized to design and implement the system. The system consists of a vehicle-mounted control and tracking device, a central web-based application server and database system, and a fuel sensor installed within the truck's fuel tank in addition to cut-off fuel pump relay installed within the power section of vehicle engine. Users can monitor the truck's locations graphically and fuel level information, and to interact with truck's engine remotely. The web-based tracking system empowers clients to monitor and browse the tracked location via web-application developed that embeds Google Maps and interacts with central database server to get the tracked vehicles' details. The utilizing the web-based tracking system facilitates the clients, with various operating system platforms, to effectively reach the requested details via the Internet access existence. Fig. 1 illustrates an overview of a typical web-based vehicle tracking system. The geographic locations are acquisitioned from GPS satellites utilizing GPS receiver as a part of the vehicle to be tracked and controlled, location coordinates are transmitted via GPRS service through utilizing an integrated GSM-GPRS module, GPRS technology is 2.5G that has its gateway to the Internet, and it is utilized based on 2G GSM networks, the locational information is transmitted to a specific destination server via TCP (Transmission Control Protocol) connection. And through the internet, any client can track the targeted trucks on Google Maps supported by a purpose-built web-application over a website.
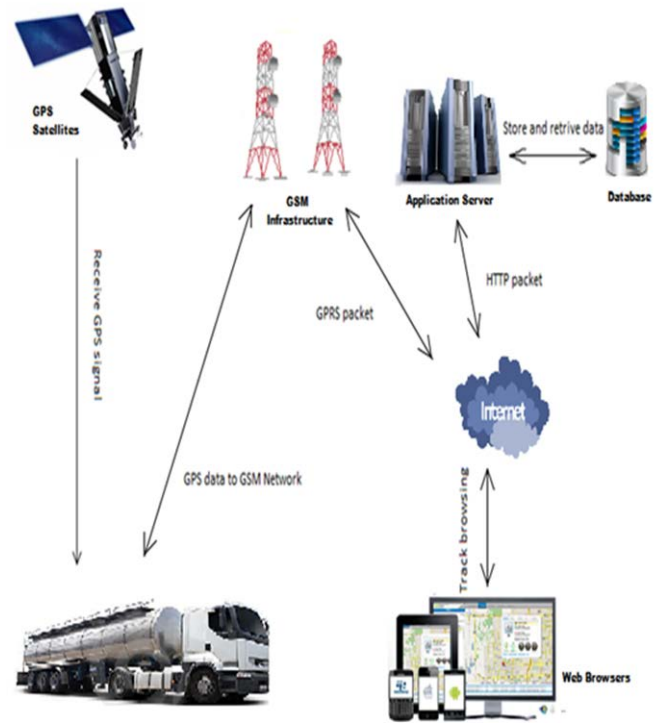

Fig. 1. Vehicle Control and Tracking System Overview

5 SYSTEM COMPONENTS AND RELASHIONSHIPS

The overall designed system functionalities outcome from the interaction and communications between the system components that are:

1. SIM808 GPRS-GPS shield module integrated with the Arduino microcontroller board and communicate via UART serial interface.
2. Arduino microcontroller board.
3. Arduino IDE software interfaced on Laptop or Desktop to communicate with the Arduino microcontroller and SIM808 GPRS-GPS shield module for the configuration operations via AT commands on USB cable interface.
4. Fuel pump cut-off relay that interfaced with the Arduino microcontroller via serial interface to perform remote control functions on the vehicle engine.
5. Fuel sensor installed within the vehicle's cargo tank that is interfaced and supported by the Arduino microcontroller board via serial interface.
6. Web\mobile application, purpose-built database and web server.

The interactions and communications between the system's components are shown in Fig. 2.
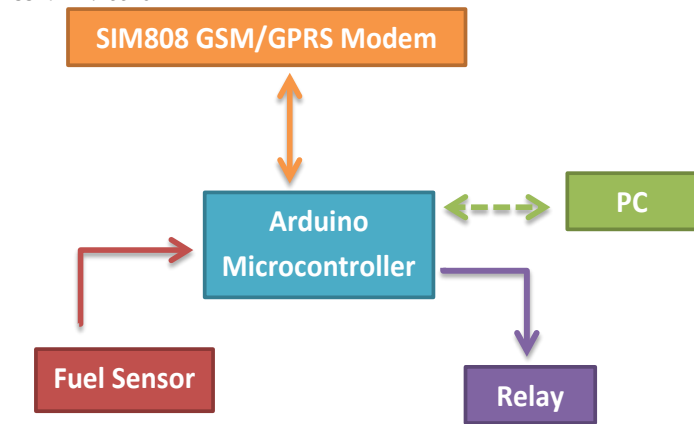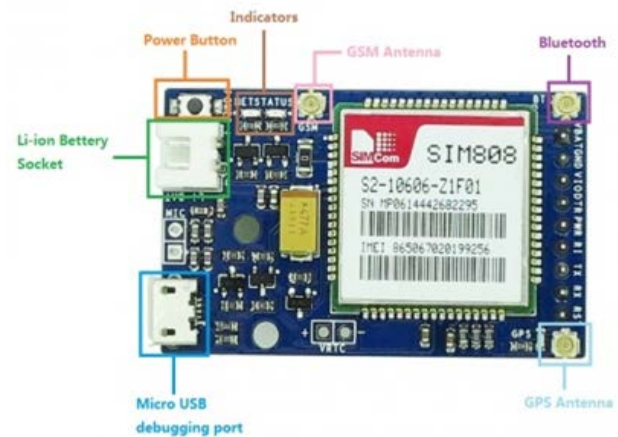
Fig. 2. System Components and Relationships

Fig. 3. SIM808 GPS/GSM module Board

6   GPS/GSM MODULE

SIM808 module is 850/900/1800/1900MHz Quad-Band GSM/GPRS module that integrates GPS technology for GPS satellite navigation. This module has a SIM application toolkit on where the SIM card is inserted. It is a compact design that integrates GPS and GPRS in one package that considerably saves both cost and time for one for developing GPS enabled applications. A GSM/GPRS modem with SIM808 module permits for creating data connections over GSM network via a standard USB interface. Cellular modems, specifically USB-stick ones, are currently at very inexpensive prices [8] [9]. SIM808 is designed upon power saving technique; therefore the current's consumption is low as 1.0mA in the sleep mode (GPS engine is powered down). It includes two variant serial ports on the board, one used for the module's cellular section and the other for the module's GPS section. The cellular serial port allows for full management of SIM808 module, so that it can be utilized to configure and communicate with the GPS receiver, with the purpose of calling for data about geographical positioning and satellite status and for transferring it to the Uno microcontroller. This is the methodology that is followed in this project design. All GPS functions are controlled by AT commands via serial port. This module utilizes AT commands for executing users' preferred functions. SIM808 transmits GPS NMEA information and communicates with Arduino microcontroller that is integrated to, with various optional baud rates that are set via the Arduino IDE software. Fig. 3 illustrates SIM808 module with its interfaces [8] [9].

7   ARDUINO UNO R3

The Microcontroller that is utilized for the project is Arduino Uno R3 as shown in fig. 4. R3 is the 3rd and latest revision of Arduino Uno. The Arduino Uno is a microcontroller board that is based on ATmega328. The ATmega328 is a single microchip controller that has 32 Kbytes (with 0.5 Kbyte which is occupied through the boot loader). Moreover, it comprises of SRAM and EEPROM that can be written and read with the EEPROM library, I/O pins, AVR microcontroller chip, a power jack, a USB connection, ICSP (In-Circuit System Programming header), and the reset button. Simply, it is connected to a computer via USB cable. The Clock speed of the Arduino is 16 MHz, thus it performs a specific task quicker than other processors or controllers. AVR chip is continuously clocking on 16 MHz regardless to what a code is performing, it never halts; thus its current's consumption is essentially independent from the code that is executed. The code execution is halted only if putting the AVR chip into its sleep mode and severely cut the current consumption for the AVR chip [10].
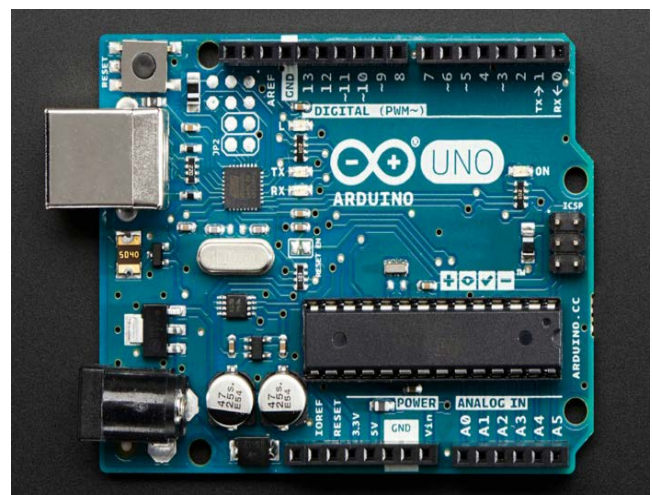

Fig. 4. Arduino Uno R3 Microcontroller

8   FUEL PUMP CUT-OFF RELAY

The system is utilized for the control purposes as the procedures described below:

1. A single pole double terminal (SPDT) relay coil pins are connected to a driving circuit which is connected to a specified Arduino microcontroller digital output pin, the driving circuit is used to boost the output current, because the power that outcome from the Arduino microcontroller is low and cannot operate the relay and cut off the engine power. A LED lamp light is installed to the relay to light up when the relay is operated and the engine is remotely stopped. Fig. 5 illustrates the wiring diagram that is utilized to perform the remote start and stop engine while Fig. 6 shows an example of a simple driving circuit.

2. The relay pole pin is connected to the vehicle power source.

3. The relay normally closed (NC) pin is connected to the fuel pump power supply wire, replacing the original wire connection.

4. The relay normally open (NO) pin is connected to a lamp, to indicate an engine cut event occurrence, this way the driver will be aware of the remotely executed engine cut command issued by the fleet management server.
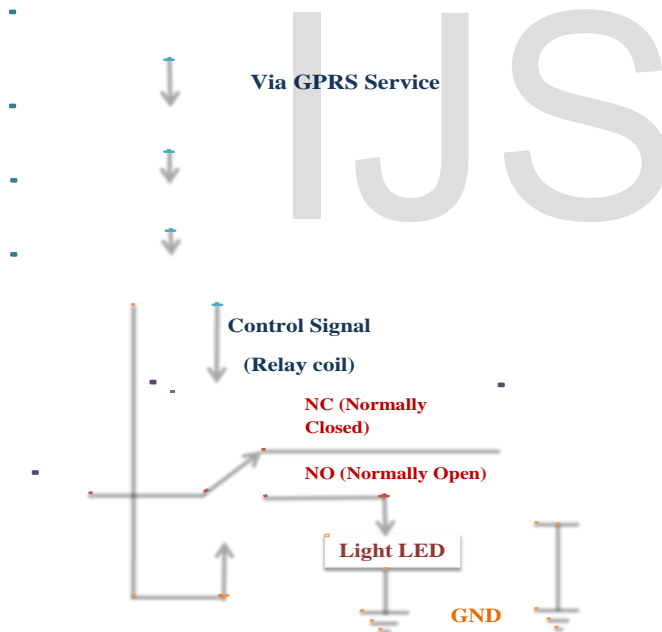

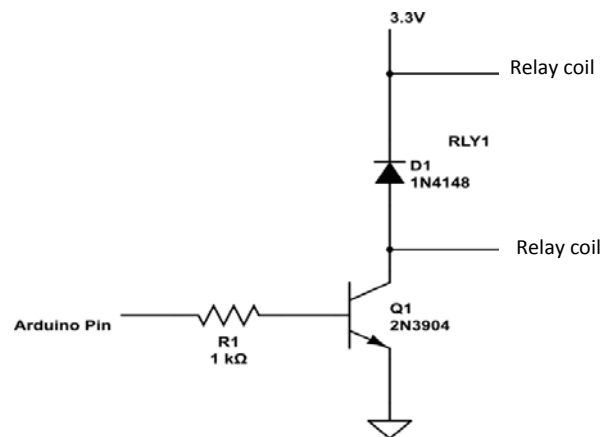Fig. 5. Wiring Diagram of the Remote Start and Stop Engine

Fig. 6. Driving Circuit

9   Variable RESISTANCE FUEL LEVEL SENSOR

Variable resistance sensor is utilized to convert the fuel quantity in Milliliter into the electrical domain in Millivolt. The calibration operation must be performed in order to eliminate any unwanted influences of the environmental conditions. It can be used to perform the tasks as the follows:

Remote fuel level monitoring with high precision measurement in external tank used for fuel cargo carriage with present and past situations.

It provides detailed information about fuel draining and fueling in cargo tank, which is fixed with time and location. This can detect any illegal fuel drain operation, which can prevent the fuel fraud and manipulation.

Installing it inside the internal tank (for consuming function) is very dangerous and may lead to an explosion while the fuel consumption is being in process because it needs to make holes and welding works in the fuel tank of vehicle. Fig. 7 shows the resistance sensor with its components.
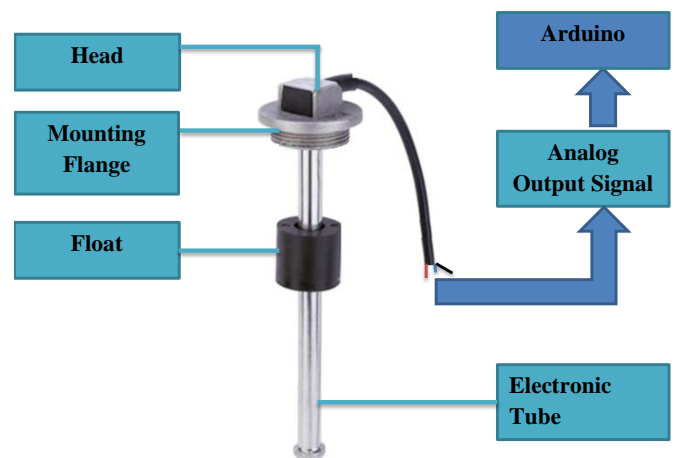

Fig. 7. Resistance Sensor

Arduino reads the sensor's values via using one analog input of the Arduino board. The resistance sensor's analog values are read as a voltage because this is how the analog inputs work. Three wires are connected to the Arduino board.

The first wire is connected to ground from one of the outer pins of the variable resistance sensor. The second one goes from 5 volts to the other outer pin of the sensor. The third one goes from the analog input 0 to the variable resistance sensor's middle pin. Fig. 8 shows the wiring interfaces between the Arduino Uno R3 and the sensor.
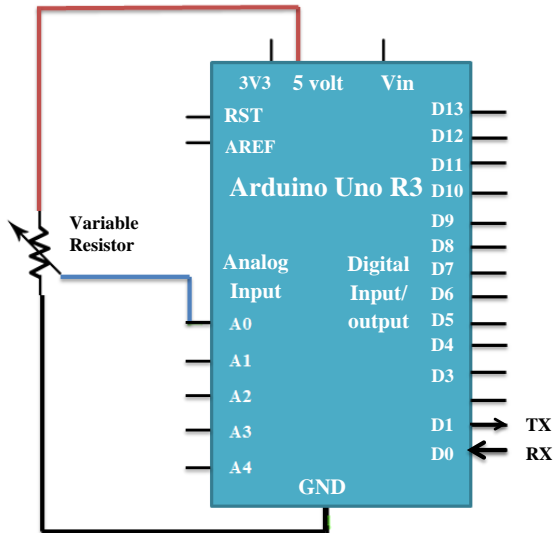

Fig. 8. Arduino to Variable Resistor Sensor Interface

Software library and sketches must be loaded to the Arduino from the Arduino IDE software on PC. The variable sensor pin is set to analog pin 0, where the sensor is attached. Another variable is created which is the sensor value for storing the values read from the sensor. The analogRead ( ) command converts the input voltage range, 0 to 5 volts, to a digital value between 0 and 1023. This is performed via a circuit within the Arduino microcontroller called ADC (Analog-to-Digital Converter). By turning up and down of the float of the variable resistance sensor duo to the change in the fuel level within the tank, the amount of resistance is changed on either side of the middle pin of the sensor. The changes of the relative resistances between the middle pin and the other two outside pins, gives a different voltage at the analog input of the Arduino board.

10 DATA GATHERING, TRANSMISSION, RECEPTION

First, initializing the serial port UART of Arduino with SIM808 must be performed on baud rate 9600. Then, Initializing the Arduino fuel pin sensor as an input to the Arduino microcontroller for analog signals that come from the fuel sensor, and initializing the engine cut pin as an output from Arduino to perform remote start and stop on vehicle's engine. Various AT commands are utilized for reading GPS location data, and transmitting the GPS location and fuel information to remote web server via GPRS connection over a predetermined periodical interval (every one second).

Then SIM808 module is Powered-On and the GPS mode is set up first for opening connection with a baud rate of GPS is set to 9600 bps, the overall procedures to set up the GPS and GSM mode for SIM808 are performed via sending AT commands through the Arduino IDE on PC towards the Arduino microcontroller which integrated with the SIM808; the procedures to set up the GPS mode are describes as the follows:

1. Turn ON GPS power supply (AT+CGPSPWR=1), if it is needed to turn it off; value zero is used instead of value 1.
2. Reset GPS in Cold start mode via (AT+CGPSRST=0), During the Cold Start, ephemeris and almanac GPS data are downloaded first to the SIM 808 from the GPS satellites before a GPS fix data has been acquired.
3. Reset GPS in Hot start (autonomy) mode via (AT+CGPSRST=1), the autonomy mode depends on the GPS satellites only for retrieving location information of GPS signals. This mode need line of sight connection to the GPS satellites, and cannot be utilized indoors and environments that include physical obstructions, it enables the GPS receiver to completely synchronize with four or more GPS satellites for the first GPS fix, that may take few minutes.
4. Acquire and read the present GPS information of the geographical location via (AT+CGPSINF=0). Moreover, the Arduino microcontroller reads the Fuel level.
5. Information that is come as analog signals from the fuel sensor installed within the vehicle's tank to the Arduino which converts in to digital signals.
6. Encapsulate the GPS data received to $GPRMC NMEA sentence via utilizing (AT+CGPSOUT=32) command, after these successful settings; the NMEA sentence will be ready for transmission.

The procedures to set up the GSM mode are describes as the follows:

1. Set up the Access Point Name (APN), user name and password for the PDP context via the command (AT+CSTT="goco","",""). APN is the name of a gateway between GPRS and the Internet and the APN that is used in this project is "goco" supported by ASIACELL Telecommunication Company in Iraq. Specifically, it identifies the PDN (packet data network) that the SIM808 communicates with it for getting GPRS service. Moreover, APN is used to define the service types.
2. Bring up the GPRS wireless connection via (AT+CIICR) command based on the previous configuration that is set via the (AT+CSTT) command.
3. Get the local IP address via (AT+CIFSR) command. It is essential that the PDP context must be activated before IP address is gotten.
4. Config.s the SIM808 module for a single IP connection via (AT+CIPMUX=0) command, and it must be executed before starting the TCP connection.
5. Startup TCP connection via using (AT+CIPSTART="tcp", "76.76.18.171", "80"). '76.76.18.171' indicates to the remote server's IP, and '80' refers to the remote server's port.
6. Set the packets length that will be transmitted via (AT+CIPSEND=length), whenever the response is OK;

the data has been sent via executing (AT+CIPSEND) command.

After sending the position and fuel information, SIM808 acquires the new location data after one second passed, then it will check if the angle of movement of vehicle is changed or not, if the angle still without any change; then an incremental counter is set by AT command via the Arduino microcontroller that counts for every one seconds till reaching the limit which is set to 15 seconds, if this limit is reached, then the counter is reset and SIM808 restarts again to check the availability of the GSM network and to acquire another new GPS data, this is a loop of procedures that is performed periodically to detect any change of the vehicle location. In the case of non-reaching the limit of 15 seconds; the Arduino will check for receiving a command to set the engine cut o/p pin of Arduino microcontroller either On or Off to perform remote start and stop for vehicle engine via the fuel pump relay attached to the power section of the vehicle engine. The procedures sequence of designing the control and tracking system via AT commands is illustrated in fig. 9.

$GPRMC sentence includes information of device ID, latitude, longitude, Angle, Fix Time, Fuel level, Cell ID, MNC (Multinational Corporation), MCC (Mobile Country Code), LAC (Location Area Code). Packets which include the $GPRMC sentences and fuel status is transmitted to the remote server using TCP connection via GPRS service. If GSM network isn't available, subsequently the data are stored on the internal memory of the SIM808 module, and the network is periodically checked.

Fig. 9. Design Flowchart

11 LOCATION DATA ANALYSIS

The data transmitted from the designed hardware and gathered at server side includes Latitude, Longitude, speed of vehicle, the angle of movement on the road, the time of receiving GPS data from GPS satellites on the designed hardware tracker and the time of receiving GPS data on the server side from the designed hardware tracker. The delay between these two times of data reception at GPS receiver and server side is determined. The data is transmitted from the GPS receiver to the server every single second wherever there is a change in the movement angle, if there is no change in the angle it will transmit the data to the server for every 15 second, Thus any transmission delay is determined by subtraction operation on the two reception times of GPS receiver and the server side. Moreover, the angle and speed of movement variations are

analyzed and determined. The time on the server is UTC/GMT used to unify the time for all clients all over the world to support their time zones in various countries; to monitor the vehicles information in the exact times, whereas the time provided on the web application for this project is on the Iraq time.

Latitude and longitude values are converted into military grid reference system (MGRS) which is the geo-coordinate standard used by NATO militaries for locating points on the earth. The MGRS is derived from the Universal Transverse Mercator (UTM) grid system and the universal polar stereographic (UPS) grid system, but uses a different labeling convention. The UTM zone of Iraq is 38S, the longitude and latitude values are converted to x and y which are, successively the Easting and Northing, The MGRS based Mercator projection is a map projection, the following equations used to map the longitude and latitude into x and y on the map for this projection standard.

$$x = \left\lfloor \frac{lon + 180}{360} \right\rfloor \cdot 2^z \tag{1}$$

$$y = \left\lfloor \left(1 - \frac{\ln\left(\tan\left(lat \cdot \frac{\pi}{180}\right) + \frac{1}{\cos\left(lat \cdot \frac{\pi}{180}\right)}\right)}{\pi}\right) \right\rfloor \cdot 2^{z-1} \tag{2}$$

$$lon = \frac{x}{2^z} \cdot 360 - 180 \tag{3}$$

$$lat = \arctan\left(\sinh\left(\pi - \frac{y}{2^z} \cdot 2\pi\right)\right) \cdot \frac{180}{\pi} \tag{4}$$

Where z represents the zoom level parameter which is an integer value in between 0 (zoomed out) and 18 (zoomed in).

By mapping values of x-axis which represents longitude and y-axis which represents latitude to each other; it will create the route of travel, as a study case used in this project that the route started in Baghdad-Iraq, from Al-Salihia zone to the University of Al-Nahrain by passing Al-Sinak Bridge to the Abu-Nuass Street to reach the destination in the University of Al-Nahrain, the data of the movement was on 28/3/2016 started from 12:12 pm to 12:30 pm as shown in fig. 10.
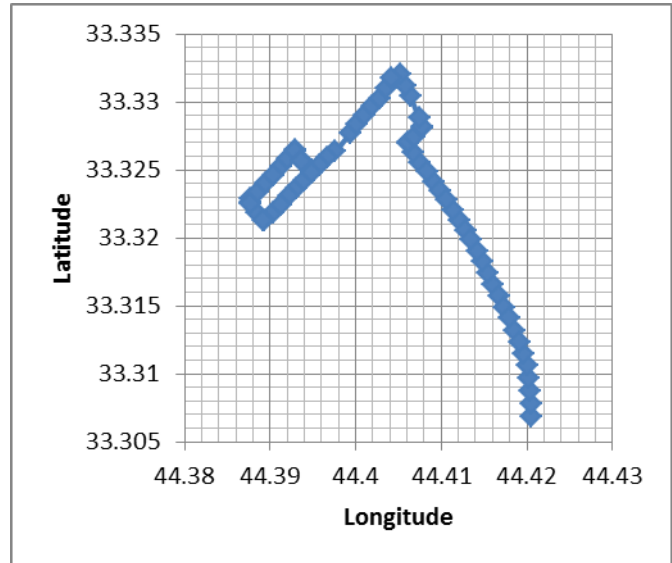

Fig. 10. Designed route on Ground

Fig. 11 displays the Designed Hardware Route on Google Map embedded in the web application. The route expressed in blue route upon the MGRS Projection via the equations (1), (2), (3), (4), and (5).
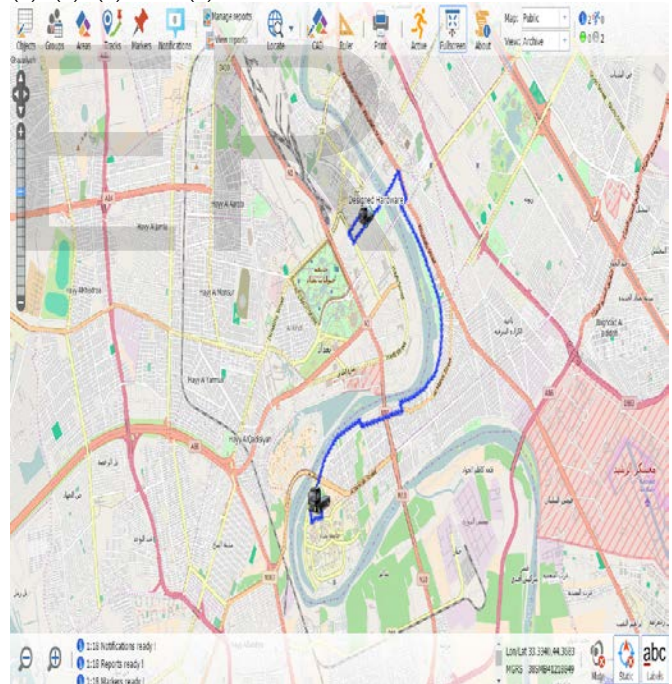

Fig.11. Designed Hardware Tracker Route on Google Map

12 STARFINDER TRACKER

It is ready-made GPS receiver device which its model is the Canadian StarFinder Lipac, it is used separately and in parallel with the designed hardware tracker (Arduino and SIM 808) in order to make comparisons in the efficiency and effectiveness concerning both GPS receiver devices. When installing StarFinder tracker and designed hardware tracker within one vehicle, a study case was created with the same date, time and route in Baghdad to analyze the data for each

one and determine the efficiency and performance for each tracker.

As a comparison between these two trackers with the ideal route extracted from the web application which supports Google Map and Google Earth, the designed hardware tracker shows a better performance to determine the locations by sending more location data than the StarFinder tracker with the same period of time and same route, the designed hardware sent 70 reports of location data within 17 minutes, where StarFinder sent 50 data report within the same 17 minutes.

The designed hardware tracker shows a better performance than StarFinder tracker for expressing the precision of measurements because the designed hardware tracker is programmed to take into the consideration the change in angle of movement on the road plus time intervals limits where the StarFinder tracker takes into the consideration the time interval limits. The 'Blue' route expresses the tracked locations determined by the designed hardware tracker and the 'Orange' route expresses the tracked locations determined by the StarFinder tracker. And the 'Green' route expresses the ideal route. StarFinder tracker shows inaccuracy to determine the angle of movement whereas the designed hardware has solved this issue which makes the designed hardware tracker measurements is close to the ideal route and better than StarFinder tracker's route which displays the readings of locations in some places over buildings or non-street areas as shown in fig. 12.

13 FUEL SENSOR MEASUREMENTS

The Arduino board contains a 6 channels (8 channels on the Mini and Nano, 16 on the Mega), 10-bit analog to digital converter. This means; it will map input voltages between 0 and 5 volts into integer values between 0 and 1023. This yields a resolution between readings of: 5 volts / 1024 units or, 0.0048828125 volts (4.8828125 mV) per unit.

The fuel tank is used to test the resistor sensor is cylinder with a height of 30 cm and diameter of 4 cm that has a total fuel volume with 1000 Milliliter. The tank has an irregular concavity in the middle of the cylinder in order to show any irregular change for the fuel levels that maybe caused by the geometry of the tank's shape. values in millivolt that expressed the size of fuel in milliliter, the expected values that is gained via converting the millivolt inputted data to the Arduino into digital numbers, the expected values are a number with fraction, thus these fraction parts will be eliminated; this elimination will cause errors for the measurements. The fuel is poured in to the cylinder tank gradually started with a maximum tank size is 0 liter towards the minimum tank size which is 1 liter. All digital data in the Arduino is appended to the GPS data within the NMEA sentences and passed to SIM808 to transmit it to the server side via GPRS service. The tank has a concavity in the middle, thus it will lead to an exponential growth for the measurements in millivolt plus the digital measurements at the Arduino, it presents an error summation of 11.2912 and Average error of 0.537676 as shown in the chart illustrated in fig. 13 that shows the growth of measured values in mV when pouring the liquid into the tank the increased for every 50 mL. Equivalent, Digital Value and Error Value Growth are shown in fig. 14.

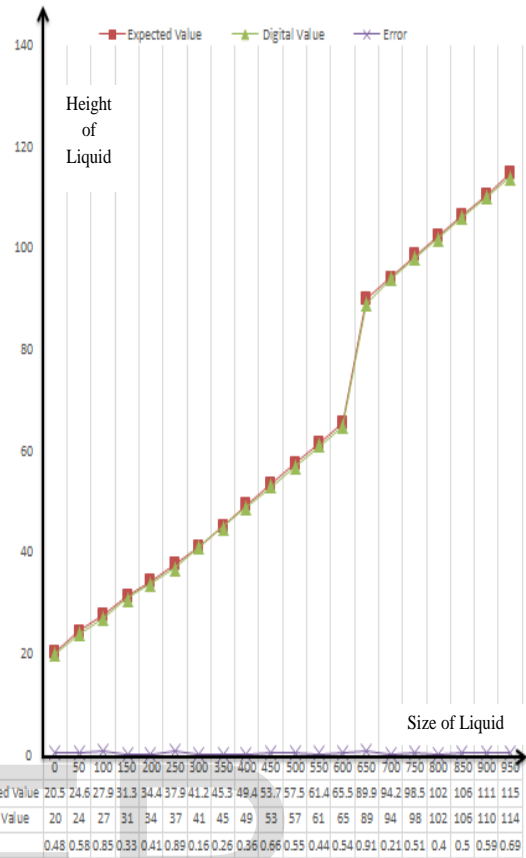Fig.12. Ideal, Designed Hardware and StarFinder Routes

Fig. 13. Measured Values Growth

Fig. 14. Equivalent, Digital Value and Error Value Growth

14 CONCLUSION

1. A cost effective vehicle control and tracking system was presented; the system utilized the well-known GPS and GPRS technologies, appropriate for inclusive range of applications everywhere throughout the world. The real-time tracking is provided via the integration of GPS and GPRS technologies. The transmission data's cost is enormously is reduced through utilizing GPRS service rather than utilizing SMS service. Google Map is a free service and the use of TCP protocol as the data transmission method. The inclusive implementation of the proposed system considerably superseded the traditional and expensive SMS based tracking system.

2. The designed system shows locations of the tracked vehicle in real-time on Google Map with an error in between 2.5 and 35 meter. The system accuracy is extremely dependent on the GPS coordinates received from GPS satellites and the GPS tracker device used, whereas usability and reliability are dependent on the reliability of the GSM-GPRS communication network. The designed system is highly effective in zones wherever there is wide GSM-GPRS network coverage. Client application is a PC-based web browser application that enables the client to login to the web application to check the tracked vehicle locations and

       movement actions anywhere and everywhere all over the world ever the Internet is accessible.

3. There is a security gap for the traditional system that utilized only tracking system without the integration with control system, where the GPS/GPRS technologies cannot prevent a vehicle from the actual theft operation; they can subsequently assist to recover it by knowing and tracking the vehicle location after the theft. GPS/GPRS technologies cannot enable the vehicle owner to communicate with the vehicle engine online, even if the vehicle owner is certain that his vehicle was stolen and can displays its current locations on Google Map. The designed system proposed security to perform control operation as the remote stop/start for the vehicle engine, if it is stolen and to track its movement online for retrieval purpose. The proposed system is an integration of various modern embedded and communication technologies.

4. The designed system helps to establish substantial solutions for the fuel theft problem plus the vehicle theft problem. The proposed system provides periodic details about the fuel level status within the external tank that is used to transport the fuel cargo. Hence any unauthorized operations to steal the fuel or vehicle; will be detected in real-time, the system provides complete protection via the remote control on the vehicle to stop its engine immediately and the support of alerting system in the remote controlling and monitoring side.

REFERENCES

[1] A. Ajagbe et al, "The Use of Global System of Mobile Communication (GSM) Among University Students in Malaysia," International Journal of Innovation, Management and Technology (IJIMT), Vol. 2, No. 6, pp. 512-518, Dec. 2011.

[2] S. Ahmed et al, "Real-Time Vehicle Tracking System," M.S. thesis, Dept. Elect. and Electron. Eng., BRAC Univ., Dhaka, Bangladesh, 2015.

[3] M. Medagama et al., "GIS/GPS/GPRS and Web – based Framework for Fleet Tracking," in Nat. Conf. Geoinformatics Applications, Sri Lanka, 2008.

[4] N. Chadil et al., "Real-Time Tracking Management System Using GPS, GPRS and Google Earth," in Proc. 5th Int. Conf. Electrical/Engineering /Electronics, Computer, Telecommunications and Information Technology (ECTI), 2008 © IEEE Int., Krabi, Thailand, May 2008, pp. 393-396. doi: 10.1109/ECTICON.2008.4600454

[5] A. El-Rabbany, "Introduction to GPS: The Global Positioning System," Boston, USA: Artech House, 2002.

[6] J. Bannister et al., "Convergence Technologies for 3G Networks IP, UMTS, EGPRS and ATM," England: John Wiley & Sons Ltd., 2004.

[7] "GPRS General Packet Radio Service," Usha Communication Technology Ltd., London, UK, June 2000.

[8] L. Fried, "Adafruit FONA 808 Cellular + GPS Breakout," Adafruit Industries Co., New York, USA, Feb. 2016.

[9] "SIM808 Hardware Design V1.00," Shanghai SIMCom Wireless Solution Ltd., Version No. 1.00, Shanghai, China, Mar. 27, 2014.

[10] L. Fried, "Arduino Tips, Tricks, and Techniques," Adafruit Industries Co., New York, USA, Jan. 2014.