Khulna University Studies Special Issue ICSTEM4IR: 925-935, 2022

ORIGINAL ARTICLE

SCIENCE & ENGINEERING

IoT-BASED VEHICLE TRACKING SYSTEM FOR KHULNA UNIVERSITY

Mahmudul Hasan Abid[1], Aminul Islam[2,*], Atish Dipankar Biswas[3], Iffat Ara Talin[1]

[1]*Electronics and Communication Engineering Discipline, Khulna University, Khulna 9208, Bangladesh*

[2]*Computer Science Engineering Discipline, Khulna University, Khulna 9208, Bangladesh*

[3]*ICT Cell, Khulna University, Khulna 9208, Bangladesh*

KUS: ICSTEM4IR-22/0234

Manuscript submitted: June 29, 2022 Accepted: September 28, 2022

Abstract

Right now, there is no perceptible vehicle management framework for Khulna university that provides a vehicle monitoring system, real-time fuel consumption data analysis, vehicle route optimization by calculating passenger traffic, and so on. Albeit creating that framework is the holistic objective of the total project, the significant objective of this paper is to propose a cost-effective, reliable IoT-based vehicle monitoring system for Khulna University based on the developed prototype device. The open-source controller and GPS-GPRS-based module determine the real-time position of the vehicle, and the location of the test vehicle can be communicated via GPRS, which is provided by the GSM network. This real-time location data is stored in a web-based IoT platform. Authorized users of the system can access this information via the internet. The proposed solution will facilitate all stakeholders, including teachers, students, and other Khulna University staffs, to use this information to make smarter travel selections. This will also pave the way for future research like intelligent vehicle route optimization by storing real data from different vehicles of Khulna University on the online database.

Keywords: Vehicle Tracking, Arduino, SIM800L, GPRS, IoT

Introduction

As technology advances at a rapid rate, automated vehicle monitoring systems are being employed in various approaches to track and show vehicle positions concurrently. The vehicle tracking techniques have been successfully employed in the shipping sector at first because individuals needed to know where a certain ship would be at a specific moment. Nowadays, many other fleet management systems are also implementing this technology for convenience. This study presents a vehicle tracking system for Khulna University based on GPS-GPRS technology that would help its stakeholders to find the exact position of an upcoming vehicle on its route.

*Corresponding author: <aminul@ku.ac.bd >

DOI: https://doi.org/10.53808/KUS.2022.ICSTEM4IR.0234-se

Khulna University (KU) has a pretty big campus, and it allows students to be both on and off-campus. Besides students, there are a large number of teachers and employees who stay outside of the campus area and everyday travel to the university. Like other public universities, KU offers campus transportation services to make it easier for this large group of students, teachers, and employees to move around the city. Despite a fixed schedule, buses seldom arrive on time owing to unexpected circumstances. As a result, students cannot but wait for an extended period of time at the bus stop for the arrival of the transport vehicle. The core vision of this study is to offer such an approach to distant students that can minimize bus waiting time and help them find all required information about the bus's arrival/departure time, real-time position, and projected waiting time. To this end, in this paper, a prototype of the tracking device is presented that tracks the real-time location of a vehicle and sends the location information to an IoT-based real-time database server. This stored information in the server will then be used by an android application to show the bus location on Google Maps. (The future work of this project).

One of the major tasks of this study is the transmission of location data. GSM, along with SMS technologies, have been utilized for wireless data delivery. SMS technology is able to come up with bus location details via the GSM network together with the GSM modem (Ramani et al., 2013, Wen & Hsu, 2005). The SMS technology has grown in popularity for its quick and easy-to-use approach to sending and receiving data (*Public Transportation Management Service Using GPS-GSM - PDF Free Download*, n.d.). But for this proposed literature, this method is not compatible because of the high volume of SMS messages. In the real scenario, a large number of people would try to find the location of the vehicle, which would be impracticable for that huge number of stakeholders with respect to the huge network load and SMS cost.

Some authors also have used smartphone hotspot-based WiFi technology recently to send the data to a server. In this method, a smartphone needs to be always near the vehicle, which is also not a very good solution. A GNSS-based bus monitoring system was proposed by Kannaki et al. (2007). The major purpose of their systems was to cut down the intermission period of passengers at bus stops by providing information about bus whereabouts to passengers via SMS. Manikandan & Niranjani (2014) developed a web application based on GNSS that furnishes real-time values of the speed of the bus, as well as bus position on Google Maps. Raja et al. (2014) presented a tracking system that can provide information about the number of available seats along with the position of the bus. They utilized wireless communication technologies such as GSM and GPS to convey information about the number of unoccupied seats in the bus and the real-time position of the vehicle.

Chandurkar et al. (2013) came up with a monitoring system that can provide the passengers with the present position of buses as well as their expected arrival time at several stops along their itineraries. The bus's current location and route were determined by the link updater. Their proposed system includes notifying the commuters about the predicted arrival time of the bus by updating information in the control unit through a display board installed at bus stops.

Zhian & Han (2010) designed a prompt public transportation system employing ZigBee and GSM. Their proposed model showed improved service quality, as well as operational efficiency. Additionally, their system offered the passengers to obtain information about the individual bus. However, this approach needs the use of an additional modem.

In this work, a GPRS-based data transferring system has been used, which is able to send data to the internet without using a WiFi network. To track the location's coordinates, a U-Blox NEO-6M GPS Module is used.

Methodology

In this work, a GPS-GPRS-based vehicle monitoring system is proposed for Khulna University (KU) buses that allows the stakeholders to track the vehicle's location during its journey. The present position of the vehicle has been determined by a GPS device integrated inside the target vehicle, and the location coordinates

Khulna University Studies Special Issue (ICSTEM4IR): 925-935:2022
DOI: https://doi.org/10.53808/KUS.2022.ICSTEM4IR.0234-se

can be communicated via GPRS, which is provided by the GSM network. This real-time location data is stored in a secured server (Firebase) using HTTPS protocol. The Figure 1. below depicts the procedure.

Two buck converters are used to power up the Arduino and GSM module individually from a 12-volt battery. The GPS module is powered by the Arduino board.

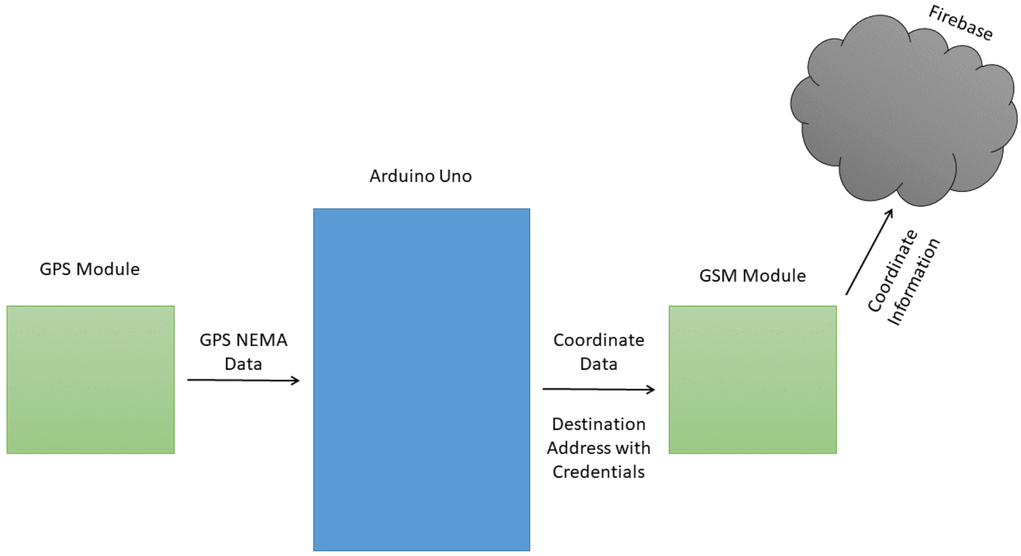

Figure 1. The system structure at a glance

Description of used components

Arduino

The brain of this project is made up of an Arduino UNO R3 microcontroller. That is a Microchip ATmega328P microcontroller-based open-source board that is designed and developed by Arduino. cc. The board has pre-installed I/O pins for both digital and analog data that may be used to connect to various circuit expansion boards. The board features six analog I/O pins plus 14 digital I/O pins, and six digital pins are capable of providing PWM output. It uses the Arduino IDE (Integrated Development Environment) through a USB cable of type B. There are two possible ways to power up the board, the first one is by a USB connection, and the second one is through an external 9-volt battery. The optimum voltage for an Arduino board is 7-12 volts. A software program is written in the C programming language to control the connected modules. After the compilation, the code is saved into the microcontroller's flash memory which starts executing every time the Arduino is powered up.

GSM/GPRS module

The GSM module connects the device that is placed inside the vehicle to a distant server to provide the current location of the vehicle through an HTTPS connection using the GSM/GPRS network. In this project, SIM 800L is utilized as a GSM module. The SIM800L is a compact cellular module that can exploit GPRS-based communication and both transmission as well as the reception of SMS, voice calls, or cellular data. It is a low-cost module aided with a small footprint and quad-band frequency handling capabilities. That makes it an ideal alternative for projects that need long-distance communication.

Figure 2. Arduino Uno.

Figure 3. SIM800L GSM module

GPS module

Trilateration is a technique that is data collected from satellites to find a pinpoint location here on Earth. Global Positioning System, also known as GPS devices, specifies a particular location using trilateration. A GPS receiver uses radio waves to measure interspace to satellites. The technique of trilateration is related to triangulation which is used to measure angles (Tim Gunther, 2020). GPS modules are infused with minuscule processors paired with antennas that receive data directly from satellites through specific RF frequencies. It will then obtain timestamps from each apparent satellite, as well as other data. The location and time can be precisely determined if the antenna of the module detects four or more satellites. The open source NEO-6M GPS module is selected for this project. It is a high-performance full GPS receiver accompanied by a built-in 25 x 25 x 4mm ceramic antenna.

Buck Converter

Buck converter is utilized to lower the output DC voltage from the provided input DC voltage in SMPS circuits. As the DC input, rectified AC supply can also be used. Buck converter is beneficial when electrical isolation between the switching circuit and the output is not required, but in the case of rectified AC input, isolating the rectifier and the AC source is important. The isolation can be given by a main isolating transformer. The LM2596 module is chosen in this project to step down the 12-volt input power to the desired voltage from the vehicle battery. This module can drive a 3-A load with an outstanding line and load control. Some of the versions of these devices come up with fixed output voltage, while some versions are facilitated with adjustable output voltage. We utilized the adjustable edition of the LM2596 module in the project as the power loss and the heating issue is less in this edition. The LM2596 module has an operating frequency of 150 kHz that allows filter components that are even smaller than switching regulators operated at a lower frequency.

Figure 3. NEO 6M GPS module.

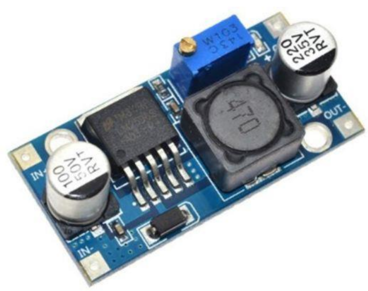

Figure 4. LM2956 adjustable buck converter.

DOI: https://doi.org/10.53808/KUS.2022.ICSTEM4IR.0234-se

Wiring of the components

The wiring of components in the proposed model is given below.

1. The VCC and the Ground pin of the GPS module should be connected to the 5V pin and any of the GND pins of the Arduino board, respectively.
2. The Rx pin of the GPS module will be connected to pin '1' (Tx pin) of the Arduino board. And The Tx pin of the GPS module will be connected to pin '0' (Rx pin) of the Arduino board. (According to the proposed system)
3. The VCC and the Ground pin of the GSM module should be connected to the +V point and -V point of the buck converter 1, respectively.
4. The Rx pin of the GSM module will be connected to the newly defined Tx pin of the Arduino board via Software Serial. And The Tx pin of the GPS module will be connected to the newly defined Rx pin of the Arduino board via Software Serial. (According to the code, Rx pin is '2' and Tx is '3'.)
5. The Vin pin and the Ground pin of the Arduino should be connected to the +V point and -V point of the buck converter 2.
6. The buck converter 1 should be tuned so that its output is 3.7-4.20 volt.
7. The buck converter 2 should be tuned so that its output is around 9 volts.
8. The +Vin and –Vin pins of two buck converters will be connected to the 12-volt battery of the vehicle.

** The SIM800L modules need almost 2-ampere of current when it is connected to the network. So, it is not worth it to power the SIM800L from the Arduino board directly.

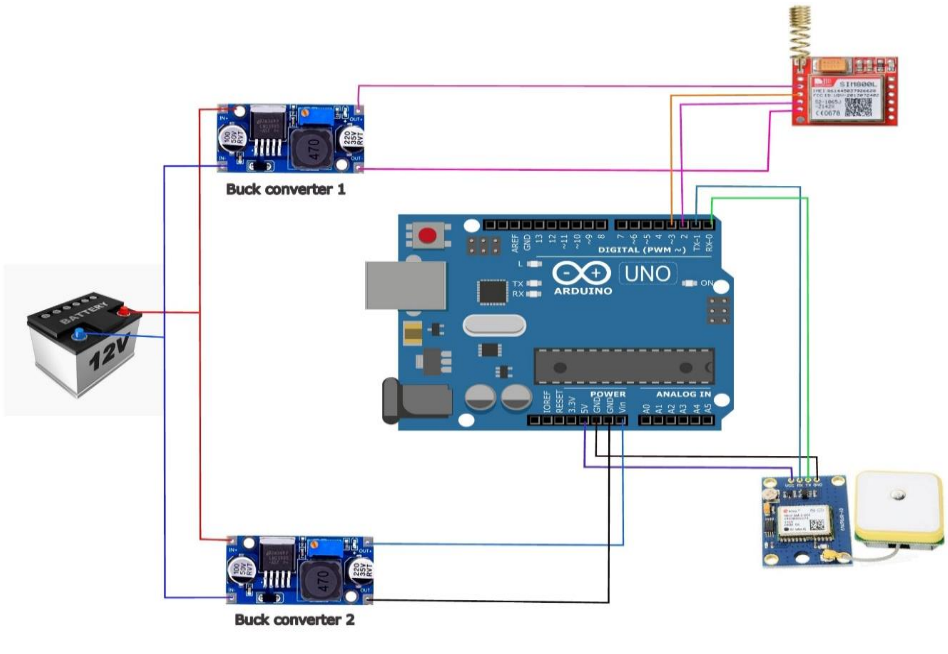

Figure 5. Wiring diagram.

In the presented model, a 12 V battery is chosen because most of the vehicle's battery is 12 V. All the GPS and GSM modules use serial communication to send and receive data from Arduino. The Arduino Uno has one serial port (pin no. '0' & '1'), which in this case is used by the GPS module. As a result, another serial port must be defined for the GSM module, which is accomplished using the Software Serial library. The 'buck converter 1' is used to step down the 12 volt supply to 4.10 V to feed the GSM module. The GSM module is not powered up from the Arduino board as it requires almost a 2-ampere current when it is connected to the network (*SIM800L Design Datasheet Pdf - Hardware Design. Equivalent, Catalog*, n.d.). The GPS module is powered up directly by the Arduino board. The 'buck converter 2' is used to step down the supply voltage to 9 volts and feed the Arduino board. Although the supported input voltage range by the Arduino Uno is 6 to 20 V, the recommended voltage range is 7 to 12 V (*Arduino Uno Rev3, n.d.*). The 5 V pins may deliver less than 5 V if the board is fed with lower than 7 V, and the voltage regulator could overheat and harm the board if more than 12 V is used to feed. However, the output voltage from a 12-V battery is roughly 13 V when it is fully charged. So, for added safety, the 'buck converter 2' is utilized, which may be skipped for a further lower-cost design, and the Arduino board can be powered straight from the 12 V battery.

The code

The Github link to this project's code is given in the reference section. *(MH-Abid, 2022/2022).* The code is written following the algorithm depicted in Figure 6.

The most challenging part of the coding is esablishing an HTTP session with the cheap SIM 800l module, which is described in the flow chart of Figure 7. The following eight AT commands need to be executed for a successful HTTP session. The corresponding AT commands, along with their functions, are given below-

- "AT+HTTPINIT"- To start HTTP connection.
- "AT+HTTPSSL=1"- To Enable SSL 1.0 as the server is an HTTPS server. For the HTTP server, the value should be '0'.
- "AT+HTTPPARA=\"CID\", 1"- To set up parameters for HTTP session.
- "AT+HTTPPARA=\"URL\","+FIREBASE_HOST+".json?auth="+FIREBASE_SECRET- To set the HTTP URL - Firebase URL and FIREBASE SECRET.
- "AT+HTTPPARA=\"REDIR\",1" – To set up redirect.
- "AT+HTTPPARA=\"CONTENT\",\"application/json\"" – To set up content type.
- "AT+HTTPDATA=" + String(data.length()) + ",10000" - To set up data length.
- "AT+HTTPACTION=1" – HTTP post request. The value should be '0' for HTTP get request.

After posting (or getting) the data, the session should be terminated. The AT command for terminating a session is- "AT+HTTPTERM". (Visit the Github link for the full code)

Steps to run the code

After downloading the code, the following steps should be followed to run the code.

1. Download and install Arduino IDE.
2. Download TinyGps (Hart, 2013/2022) and SoftwareSerial (*ArduinoCore-Avr/SoftwareSerial.h at Master · Arduino/ArduinoCore-Avr*, n.d) libraries from GitHub, which are used in the program.
3. Include the libraries in your environment. [Sketch>Include Library>Add zip library>(select the Library from the folder where it is kept).

Khulna University Studies Special Issue (ICSTEM4IR): 925-935:2022
DOI: https://doi.org/10.53808/KUS.2022.ICSTEM4IR.0234-se

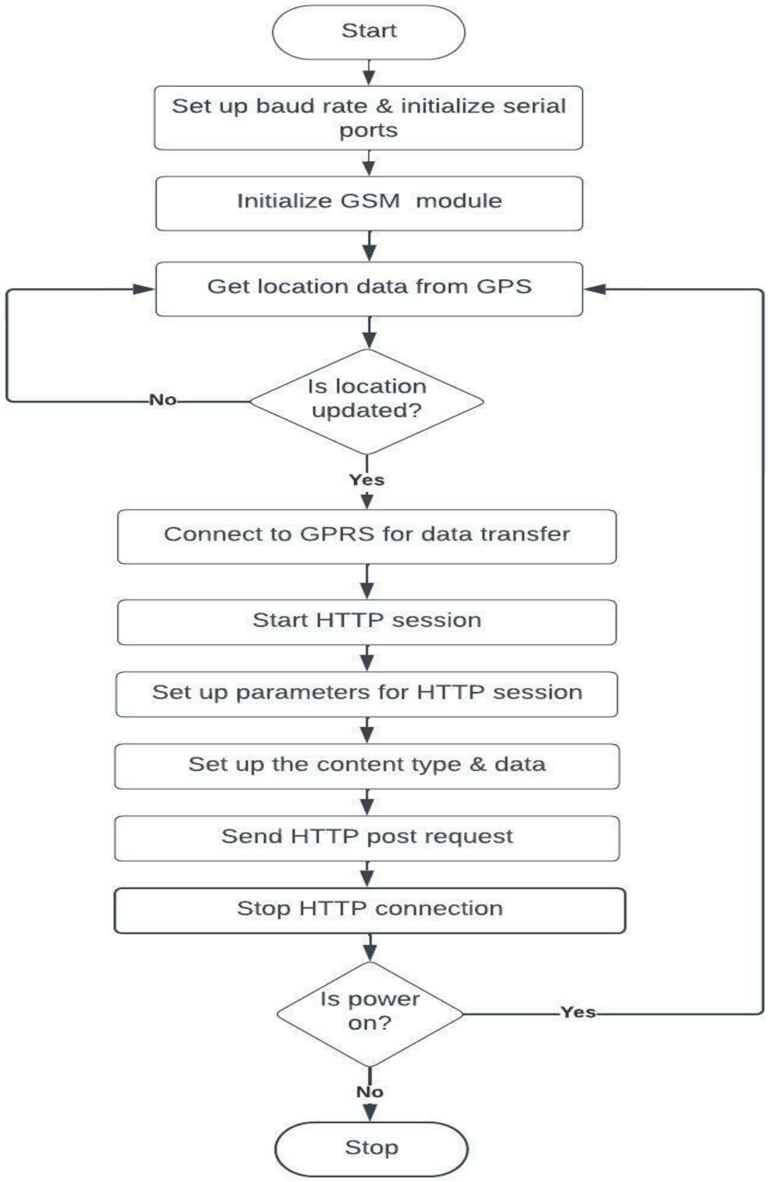

Figure 6. Flow chart of the algorithm followed in the proposed model.

Results

The IoT prototype device is developed following the circuit diagram shown in section 2.2 and placed on a Bajaj motorbike for testing, as shown in Figure 8 and 9.

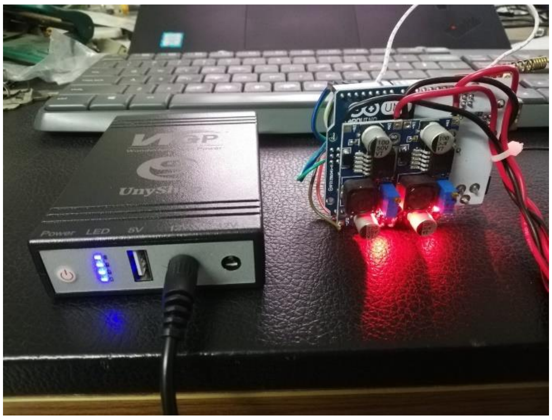

Figure 8. The prototype is connected with a 12V portable battery.

Figure 9. One of the authors places the prototype in the test device

The device is successfully sending data to the firebase database, which is depicted in the debugging output of Figure 10.

```
23:06:14.630 -> Latitude= 22.797396 Longitude= 89.552605
23:06:14.678 -> {"lat":22.797396,"lng":89.552605}
23:06:14.678 -> AT+CGATT?
23:06:14.725 -> +CGATT: 1
23:06:15.245 -> AT+SAPBR=0,1
23:06:15.245 -> OK
23:06:15.812 -> AT+SAPBR=3,1,"Contype","GPRS"
23:06:15.859 -> OK
23:06:16.379 -> AT+SAPBR=3,1,"APN",internet
23:06:16.426 -> OK
23:06:20.378 -> AT+SAPBR=1,1
23:06:20.378 -> OK
23:06:20.944 -> AT+SAPBR=2,1
23:06:20.944 -> +SAPBR: 1,1,"10.154.128.8"
23:06:21.007 ->
23:06:21.007 -> OK
23:06:21.454 -> AT+HTTPINIT
23:06:21.501 -> OK
23:06:22.020 -> AT+HTTPSSL=1
23:06:22.020 -> OK
23:06:22.585 -> AT+HTTPPARA="CID",1
23:06:22.585 -> OK
23:06:23.336 -> AT+HTTPPARA="URL",https://bus-location-d7f79-default-rtdb.firebaseio.com/.json?auth=
```

Figure 10. Arduino Serial Monitor shows a portion of the debugging output along with some AT commands.

As seen in Figure 10, the GPS module uploads its coordinate data successfully. As per the figure, the latitude of the device location was 22.797396, and the longitude was 89.552605. According to the proposed algorithm (as shown in Figure 7), after extracting the location data, the GPRS connection is established. The required AT commands (for SIM800L) to establish GPRS connection are- (a) AT+SAPBR=0,1 (b)

AT+SAPBR=3,1,\"Contype\",\"GPRS\" (c) AT+SAPBR=3,1,\"APN\","+APN (d) AT+SAPBR=1,1. As figure 7 portrayed, all the AT commands are successfully executed, and the GSM module is connected to the internet through IP address 10.154.128.8, which is displayed using the AT command AT+SAPBR=2, 1. After establishing the GPRS connection, the next step is to establish HTTP session. As Figure 7 depicts, the module is successfully connected to the HTTP server whose address is- *bus-location-d7f79-default-rtdb.firebaseio.com*. (The required AT commands to establish the HTTP session are given in section 2.3). The device is capable of sending its location information to this server from anywhere in the world when it is powered up, and a valid SIM card is inserted. When the device was placed in the test vehicle, it successfully sent data to the "*bus-location-d7f79-default-rtdb.firebaseio.com*" database. The screenshot of the corresponding firebase database is given in Figure 11.

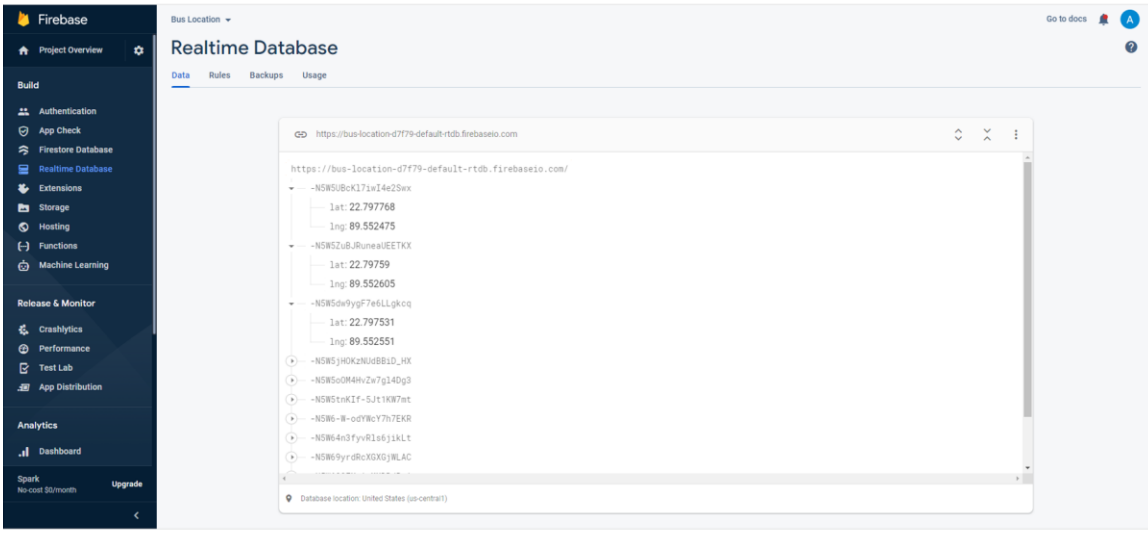

Figure 11. The device is sending the latitude and longitude coordinate to the firebase database.

From the above Figure 11, it is seen that the device can successfully send its coordinate information to the firebase database at a regular interval. In future, these coordinate data will be accessed by an android application to show the vehicle location on Google Map.

Additionally, the proposed device can also be used for SMS-based tracking. The prototype can send SMS with its real-time location with a google map link. That link shows the exact position of the vehicle at some time on a smartphone device, as depicted in Figure 12 and Figure 13.

Discussion

This work embodies one in-vehicle device along with an off-the-shelf server. The in-vehicle gadget consists of a GPS module, a microcontroller (Arduino UNO), and a GSM/GPRS module that collects vehicle position data and transmits the data to a server over the GSM/GPRS network. In this project, the famous Google Firebase is chosen as a server. The unique ID of a vehicle and its geographic coordinates were collected from the proposed in-vehicle device and are stored in the Firebase real time database. In the future, the position of vehicles will be displayed on Google maps by retrieving information from the server database. As exhibited, the technology can track the whereabouts of a vehicle from anywhere at any time. Furthermore, the developed device is a very low-cost device. Almost all the existing work in literature, where GPRS is used to send data to a remote server, use SIM9xx series or higher as GSM module, which are high-cost devices (Hafiz, n.d.; Lee et al., 2014; Yosif et al., 2017). Some also use Arduino Mega (Yosif et al., 2017). In this project, Arduino UNO is

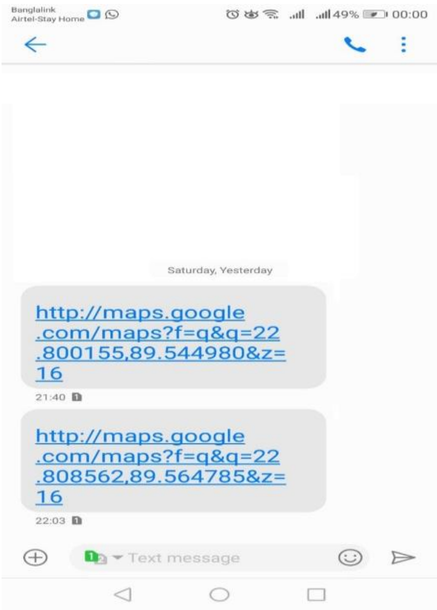

Figure 12. Google map links with real-time. coordinates sent from the GPS device.

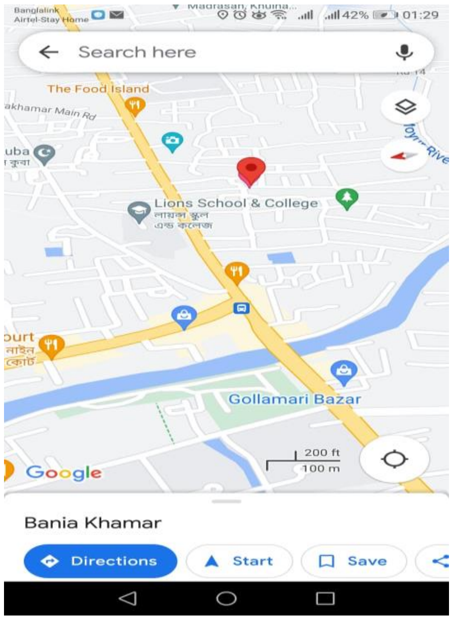

Figure 13. Real-time vehicle location on. Google map.

used along with a SIM800L module. Using SIM800L reduces the cost significantly as it is the lowest-cost GSM module. As displayed, SIM800L is able to send the location data to the remote server at a regular interval.

Conclusion

This paper demonstrates one of the key portions of the vehicle tracking project for Khulna University, which is the tracking device. The proposed model is inexpensive, and uses open source ready-to-use electronic modules. The next target of this project is to develop a smartphone application to track the vehicles of Khulna University, which will make it easier for KU students, teachers and employees to use the transportation system more effectively. This system will assist the target passengers in getting along with their time without being rushed. A complete system will also ensure proper use of the vehicles and help the university to reduce transportation costs and save precious time for the target group. This will also open the path for future studies such as intelligent vehicle route optimization by recording real time data from Khulna University's vehicles in an online database. Likewise, this low-cost and effective model can also be used for any other transportation company or railway for real-time tracking.

Acknowledgment

This research is supported by Research Grants Program (RGP), Khulna University Research Cell.

References

Arduino Uno Rev3. (n.d.). Arduino Online Shop. Retrieved June 24, 2022, from https://store-usa.arduino.cc/products/arduino-uno-rev3

ArduinoCore-avr/SoftwareSerial.h at master · arduino/ArduinoCore-avr. (n.d.). GitHub. Retrieved May 15, 2022, from https://github.com/arduino/ArduinoCore-avr

Chandurkar, M. S., Mugade, S., Sinha, S., Misal, M., & Borekar, P. (2013). *Implementation of Real Time Bus Monitoring and Passenger Information System*. *3*(*5*), 5.

Hafiz, A. (n.d.). *Vehicle Tracking System for School Bus by Arduino*. Retrieved June 25, 2022, from https://www academia.edu/37777138/Vehicle_Tracking_System_for_School_Bus_by_Arduino

Hart, M. (2022). *Mikalhart/TinyGPS* [C++]. https://github.com/mikalhart/TinyGPS (Original work published 2013)

Kannaki, A., Vijayalashmy, N., Yamuna, V., Rupavani, G., & Jeyalakshmy, G. (2007). GNSS Based Bus Monitoring And Sending SMS To The Passengers. *International Journal of Innovative Research in Computer and Communication Engineering*, *1*, 5.

Lee, S., Tewolde, G., & Kwon, J. (2014). Design and implementation of vehicle tracking system using GPS/ GSM/GPRS technology and smartphone application. *2014 IEEE World Forum on Internet of Things (WF-IoT)*, 353–358. https://doi.org/10.1109/WF-IoT.2014.6803187

Manikandan, R., & Niranjani, S. (2014). Implementation on real time public transportation information using GSM query response system. *Contemporary Engineering Sciences*, *7*, 509-514.

MH-Abid. (2022). *Vehicle-Tracking-System-Using-Arduino-through-GPRS-Network*. https://github.com/MH-Abid/ Vehicle-Tracking-System-Using-Arduino-through-GPRS-Network (Original work published 2022)

Public Transportation Management Service using GPS-GSM - PDF Free Download. (n.d.). Retrieved May 15, 2022, from https://docplayer.net/20411417-Public-transportation-management-service-using-gps-gsm. html

Raja, G., NaveenKumar, D., Dhanateja, G., & Karthik, G. V. (2014). BUS POSITION MONITORING SYSTEM TO FACILITATE THE PASSENGERS. *International Journal of Engineering Science*, *3*(3), 3.

Ramani, R., Valarmathy, S., SuthanthiraVanitha, N., Selvaraju, S., Thiruppathi, M., & Thangam, R. (2013). Vehicle Tracking and Locking System Based on GSM and GPS. *International Journal of Intelligent Systems and Applications*, *5*(*9*), 86-93. https://doi.org/10.5815/ijisa.2013.09.10

SIM800L Design Datasheet pdf—Hardware Design. Equivalent, Catalog. (n.d.). Retrieved June 25, 2022, from https://datasheetspdf.com/pdf/989664/SIMCom/SIM800L/1

Wen, W., & Hsu, S. W. (2005). A route navigation system with a new revised shortest path routing algorithm and its performance evaluation. *WIT Transactions on The Built Environment*, *77*, 11.

Yosif, S. A. E., Abdelwahab, M. M., Abd Elrahman ALagab, M., & Muhammad, F. (2017). Design of bus tracking and fuel monitoring system. *2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE)*, 1-5. https://doi.org/10.1109/ICCCCEE.2017.7867679

Zhian, L., & Han, H. (2010). *A bus management system based on ZigBee and GSM/GPRS*. https://doi.org/10.1109/ ICCASM.2010.5620407