IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 50-55
www.iosrjournals.org

Pervasive Traffic Management Using OBDUINO & Cloud Server

Priti S. Sharma[1], Prof. Sangeetha Rajesh[2]

[1](MCA (2015-18), K.J Somaiya Institute of Management Studies and Research, India)
[2] (Assistant Professor, IT, K.J Somaiya Institute of Management Studies and Research, India)

Abstract: *Smart city portray a city that redesigned innovation so as to enhance the quality with less efforts. One approach to enhancing the Traffic Management Control is taking real time data about each vehicle and analyzes it to for different requirements of the traffic control system.Current traffic management system has many limitations because of lack of information gathering from vehicles. In this paper we propose an pervasive traffic management systemthatutilize information which is collected through OBDuino device and send to the data server and the client end. Information stored on the cloud server by utilizing Arduino Microchip and it is shown to the Traffic administration control. The arrangement of information ought to be characterized by the master learning and intelligible shape. By developing this proposed framework,the continuous information log will be accessible to client end as well as activity administration control framework.*

Keywords: *Android OBD-II Reader, End-device coordinator, Internet of Things(IOT), on Board Diagnostics (OBD-II) device, Traffic management System, ZigBee.*

I. Introduction

Traffic management plays an important role in our daily life. A good traffic management systemis needed to control the traffic flow and to provide security for individuals and vehicles. The hit-and-Run cases, unsafe driving, car hacking, causing life loss of manyhuman and animal are the various types of vehicle crime reported every day. [4] The residentsare less concerned about the traffic rules in a country like India. The odds of breaking the tenets are more. Various traffic management systems were developed to control the traffic which in turn helps in controlling the snarling traffic. Still the number of accidents and the vehicle thefts are increasing in rapid speed.

OBD device are installed in vehicles to transfer vehicle information to an Arduinoand from Arduinoto the server. On-load up Diagnostics is a vehicle term which is use for self-diagnostic and gathers the ongoing information of the automobile and show to the client using Bluetooth or WIFI. (i.e., [1][2]). In this paper we propose a real-time traffic management system which is able

- To get the logs of the car and through which any suspicious action can lead police to subsidies any future mishap.
- Give real time data of the car with its current geographical location information
- To follow roadways rules by the users.
- Minimized the Road accidents.

The rest of the paper is organized as follows: section-II gives details on the background study done for the paper. Section-III depicts the proposed system. Evaluation of the proposed system is done in section IV and section V concludes the paper by specifying future enhancements.

II. Background Study

2.1 OBD-II (OBDuino)

On-Board Diagnostics(OBD) is a system used in many vehicles. OBD-II gets data from the engine control unit(ECU). It gives the wellspring of information about the autocontinuous parameters like RPM, speed, pedal position, begin push, wind stream rate, coolant temperature, status of "Check Engine" light, emission planning status, Diagnostic inconvenience[trouble] codes (DTCs), Oxygen sensor test comes to fruition, Vehicle Identification Number (VIN), Number of begin cycles, Number of miles driven with MIL on and freeze diagram: a "review" of parameters at the time a burden event. [3]

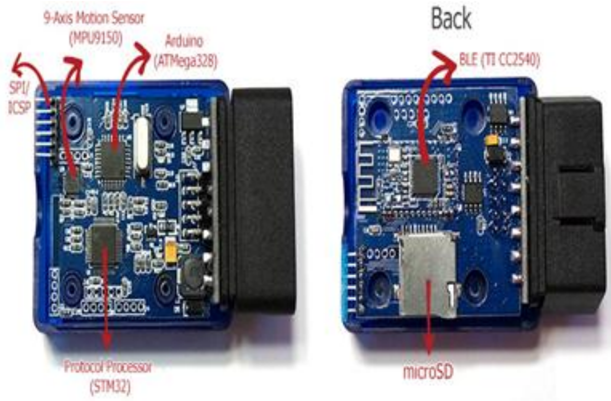


Fig. 1. OBDuino

To inculcate the capabilities of Arduino into OBD, the device is transformed to OBDuino shown in Figure 1. The Arduino depends on an Atmel AVR ATMEGA168P chip i.e. contain streak memory up to 256 KB (384 KB on XMega), EEPROM up to 4 KB, SRAM up to 16 KB (32 KB on XMega) (i.e.,[2]).

2.1.1 Controller Area Network

Controller Area Network[CAN- BUS] is network of independent controllers communicating securely.

- CAN bus defined in ISO 11898 and only cover 2 Layers [physical & data link layer] (i.e.,[5]).
- CANUSB used to read the CAN signals from a vehicle by connecting CAN_L, CAN_H and CAN GND to connect the pins of OBD-II connector.
- OBD-II scanner or scan tool, uses PID codes be fined by SAE J1979 (i.e., [3]).
- Interface connect to the Arduino on a Few pins, LCD connected differently

2.2 OBD-PID (On Board Diagnostic Parameter ID):

OBD-PID codes used to ask for information from vehicle, and indicative device, makers additionally characterize numerous more PID to their particular vehicle. PID characterize by utilizing the institutionalized information interface connector and a subset of SAR J/1979. The sweep device sends the information to the vehicle control area network[CAN]-BUS and the device on the transport perceive the PID as one it is dependable to peruse the esteem and send the incentive to the PID to the BUS. Scan tool reads the information and send to the client application or store those information on to the Arduino smaller scale shabby. The 10 modes operation utilized as a part of OBD-II standard SAE J/1979 for various reason i.e. 01-to indicate current information, 02-to solidify outline data,03-diagnostics inconvenience codes and store those esteem and so forth(i.e., [11]).

Mode 01 – take information and encode those information in Hexadecimal configuration or decimal arrangement, Mode-2 – acknowledge same PID as mode 01, with some significance yet data given is from when the stop outline was created. You need to send the casing number in the information segment of the message, etc. another mode decodes those information into comprehensible configuration.

2.3 Arduino(ATMega168P)

Arduino is single board micro-controlled or micro-controlled based ATMega(i.e., [2]).The Arduino depends on an Atmel AVR ATMEGA168P chip i.e. contain streak memory up to 256 KB (384 KB on XMega), EEPROM up to 4 KB, SRAM up to 16 KB (32 KB on XMega) [2]**.** The AT charges are utilized to send information logs to Open source Internet of Things from Arduino board. Commands are instruction which is used to control modem. There are 3 operation modes(i.e., [7]).

- The first operation mode is independent mode which is utilized for cloud.
- The second operation mode is getting to information mode which is utilized as a part of OBD-II.
- Combination of getting to information mode.

The joint access convention utilized for wi-fi association amongst Arduino and the cloud. The exchange control convention is utilized to distinguish the IP of the server which is utilized for putting away information logs. The information log gives the estimation of OBD-PID. The esteem get put away into smaller scale shabby and refreshed on the cloud server (i.e., [7])Using the MQTT broker the data logs send to the cloud sever (i.e., [8])

III. Proposed System

This section details the architecture for Pervasive Traffic Management shown in Fig 2,OBD-II is a 16-port device which acts as an interface with the auto's ECU [engine control unit]. All the vehicle related information like RPM, vehicle speed, battery voltage, fuel utilization, temperature. OBDuino (i.e.,[11]).Xbee shield allows an Arduino board to pass on remotely using Zigbee. It relies upon the Xbee module from MaxStream. This module can pass on up to 100 feet inside or 300 feet outside. Xbee module utilizes the mesh topology to speak with another Xbee. It utilizes Xbee802.15.4 module (series 1) and the recipient end likewise require Xbee802.15.4 module for correspondence.

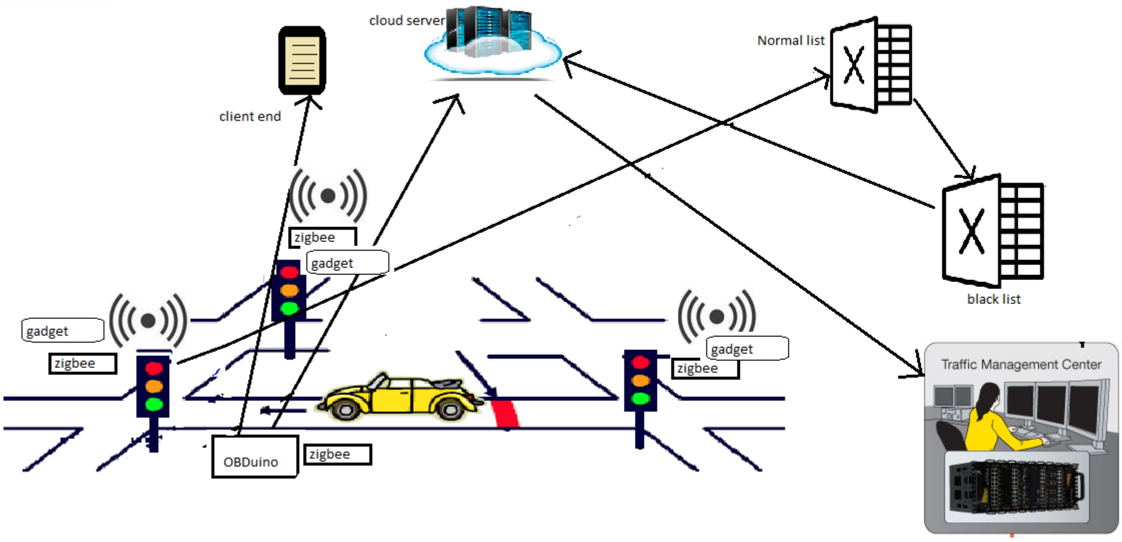


Fig 2. System Architecture

3.1 Data Acquisition from vehicle

In the proposed model,OBD-II is employed to get the auto sign in the hexa-decimal configuration and sends vehicle information to the application store. those information to the smaller scale modest which is introduced in Arduino controller after each specific time those information log are been sends to the database which is on cloud and utilizing EEPROM the information log will consequently erase in the wake of sending the information to the server. The logs contain RPM, vehicle Speed, Heat temperature and so on in JSON design

Algorithm: Vehicle OBD to Arduino Chipset

Checks for the remote association [wireless connection].
1. Connect the OBD device with the circuit
2. Check the association is effective or not?
3. Start information transmission with the device.

3.2 Communication between end devices and light poles.

Once the data is acquired from the vehicle, the next step is to established communication with nearby Traffic Signal. Xbee module working on the 2.4 GHz wireless RF ZigBee protocol.ZigBee is a standard based remote innovation to empower ease, low power remote M2M, Internet of Things (IOT) network(i.e., [3]). ZigBee RF4CE is intended for reliable and secure connection, two-way communication to-gadget control applications,low power consumption with hidden target that do not need the full-featured work organizing functionalities offered by the ZigBee specification.The device is programed to automatically establish communication with the nearestTraffic Signaland transfer information after a particular time.

Algorithm: Vehicle end-device to traffic signal device

Check for the Zigbee coordinator
1. Send OBD data log to light pole device

3.3 Storage of information log into Cloud server

The vehicle information logs are stored on the cloud with the goal that the Traffic Management Administrator get auto information and through which they will break down the suspicious or ordinary action of the driver. The information put away onto the cloud in JSON format [key-esteem pair].

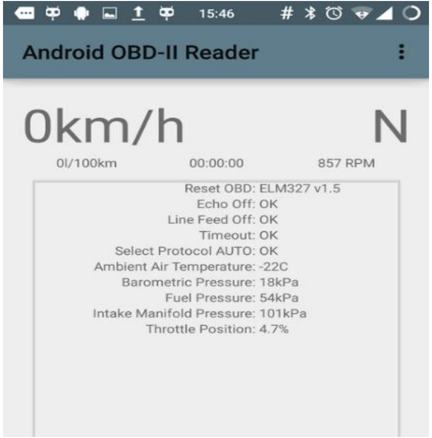
Algorithm: Arduino Circuit to Cloud server

Get Data OBD and if information got, send it to server

1. Check the last date time of the information and send just the new information from the memory
2. Else wait for information to get

3.4 Display Value in human readable form

The Mobile application is utilized for the client end where the present status of the vehicle has been shown on the device and through which the client can have the current logs information about the vehicle Engine Controller Unit [ECU]. Those information logs are send through the Bluetooth. Furthermore, client gets constant information and geological area of the vehicle in simple wave or numerical esteem. Fig 3 shows the Parameter notification about the vehicle Engine control unit.



0km/h N

0l/100km 00:00:00 857 RPM

Reset OBD: ELM327 v1.5
Echo Off: OK
Line Feed Off: OK
Timeout: OK
Select Protocol AUTO: OK
Ambient Air Temperature: -22C
Barometric Pressure: 18kPa
Fuel Pressure: 54kPa
Intake Manifold Pressure: 101kPa
Throttle Position: 4.7%

Fig. 3 Parameter Notification

3.5 Name List

These are list which contains the specific Vehicle details, who break the principles of roadways. The Vehicle details are sent by the traffic signal System to this list that some specific Vehicle breaks the rule and this are its information.

Algorithm: Traffic signal device to Name List

Check the speed [If speed is > 160kpmh]

1. If any rule break, get vehicle id and Send it in the Normal rule break list[Name List].
2. Else do nothing

3.6Black List

This is the list containing Vehicle details where the details are been checked by the Traffic Management Team about the suspicious action. The data which are supervised by the Traffic Management Team are come from the name list to the black list.

Algorithm: Blacklist Algorithm

Traffic Management Administrator checks the blacklist vehicle id

1. Get the vehicle details of specific vehicle on selection.
2. Fetch details of vehicle from cloud server.
3. Display details in Traffic Management System.

3.7 System Application

The System application demonstrates the whole Vehicle points of interest of the particular vehicle. The vehicle points of interest incorporate vehicle data to its area, following of that vehicle is significantly simpler.

Algorithm: Normal rule break list

Traffic Management Administrator checks for the type of rule break

1. If rule break severe then send Vehicle Unique Id to Blacklist
2. Else Do nothing
4. Take decision accordingly.

Methodology

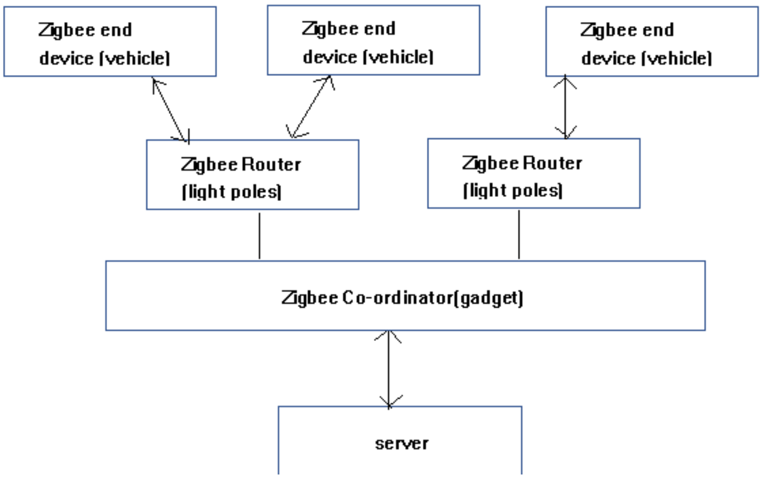

Zigbee end device (vehicle)
Zigbee end device (vehicle)
Zigbee end device (vehicle)
Zigbee Router (light poles)
Zigbee Router (light poles)
Zigbee Co-ordinator(gadget)
server

Fig.4 Communication through Zigbee

In this proposed framework the vehicle sends its log to the server and from which the Traffic Management System can have its area followed. As the Vehicle have the information port opening which is utilized to get the information with respect to the vehicle i.e. RPM (Speed), Engine Temperature, Fuel Consumption and so on the OBD device is associated with the information port which is utilized to send the auto sign in the remote way to the Arduino circuit which is use to get the OBD information and store it in its nearby EEPROM and send those information after some interim to the server. A Zigbee standard protocol used to transfer the log information from OBD-II device to light pole device. In movement flag a circuit is utilized to send vehicle Unique ID to ordinary managebreak list which bring utilizing device into the light pole. This getting of vehicle id has been work at whatever point the driver breaks any roadway run the show. A Traffic Management Administratorregulates the vehicle unique ID which are in ordinary govern break list contingent upon the kind of lead break by the vehicle driver. At the point when any suspicious activity is being seen by the manager then he/she sends the vehicle id operating at a name list where a rundown of suspicious vehicle id is recorded., when the Traffic Management Administrator chooses a particular vehicle relying upon its movement, on its choice the all vehicle logs are been brought from the server to the framework use of the Traffic Management Administrator where he/she can see all the action and vehicle data which is appeared in a continuous way. Also, contingent upon the investigation of the vehicle logs and its movement the Traffic Management specialist can take a preventive measure against the suspicious vehicle driver.

IV. Evaluate

The evaluation of the system is done in an Traffic Management System. Important parameters were observed. The threshold values for various vehicle parameters are shown in Table 2.

Object	**Parameter**	**Expected Output**	**Actual output**	**Status**
Vehicle-1	Speed	156	160	Fail
	Stole vehicle	Send alarm or flag to server	Send Alert to vehicle/legitimate user	Pass
	Accident Case	Send Vehicle data to server	Datasend to server / No accident	Pass
	Vehicle Engine Temp.	Send log to Server	Data send to server / No action	Pass
	Traffic Rule break	Send alarm or flag to server	Data send to server / No action	Pass

Table 1. Threshold value for vehicle parameter

V. Conclusion

A smart traffic management system helps to automate the whole process. Using this system traffic management administrator can know the real time information (vehicle speed, engine temperature, etc.) about the vehicles.As per the current process the tracking of the vehicle is not simple,using this system not only the tracking process will be easier but also the administrator will know the real-time information about the vehicle.

Future Enhancement of this Proposed System is Car to Car Communication using some IOT Protocol, where one car transfers and receive the data from another car which is nearby for safe driving. And another Enhancement is Introducing Big data concept in Pervasive Traffic management for costing and easy analysis purpose.

References

[1]. Byung Yun Lee; Yong Yoon Shin; Hyun Joo Bae,Development of Insurance Server System based on Vehicle Driving Information, Computing and Convergence Technology (ICCCT), 2012 7th International Conference on, Issue Date: 3-5 Dec. 2012, 156 - 159

[2]. Kristian Smith, Jeffrey Miller, OBDII Data Logger Design for Large-Scale Deployments -IEEE Annual Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, October 6-9, 2013

[3]. Sujit H. Ramachandra, K. Nitesh Reddy, Vivek R. Vellore, Vivek R. Vellore,Tareesh Kamath,A Novel Dynamic Traffic Management System using On Board Diagnostics and Zigbee Protocol, Communication and Electronics Systems (ICCES), International Conference on DOI: 10.1109/CESYS.2016.7889867

[4]. Category: Car crime[Online]Available:https://en.wikipedia.org/wiki/Category:Car_crime (2017, November 23)

[5]. Atmel-AVR [Online]Available:https://en.wikipedia.org/wiki/Atmel_AVR (2017, November 20)

[6]. Xbee Shield- Information on the current Wireless Shield (2017, November 23)

[7]. [Online]Available:https://www.arduino.cc/en/Main/ArduinoXbeeShield (2017,November 21)

[8]. Leo Raju, Sajna Gokulakrishnan, Prithika Rani Muthukumar, IoT based autonomous demand side management of a micro-grid using arduino and multi agent system

[9]. Jorge Zaldivar, Carlos T. Calafate, Juan Carlos CanoProviding Accident Detection in Vehicular Networks Through OBD-II Devices and Android-based Smartphones,http://ieeexplore.ieee.org

[10]. Arduino Create[Online]Available: https://www.arduino.cc/en/Main/Create (2017, November 21)

[11]. T. Ruslan, K. Yerden, Md. Hazrat Ali,Development of Arduino Glove-Based Autonomous Car, System Theory, Control and Computing (ICSTCC), 2017 21st International Conference on , **DOI:** 10.1109/ICSTCC.2017.8107036

[12]. 11.Alex Xandra Albert Sim , Benhard Sitohang ,OBD-II standard car engine diagnostic software development Data and Software Engineering (ICODSE), 2014 International Conference , **DOI:** 10.1109/ICODSE.2014.7062704

[13]. M. Awais Khan Niazi , Anique Nayyar , Ali Raza ,Development of an On-board Diagnostic (OBD) Kit for Troubleshooting of Compliant Vehicles,Emerging Technologies (ICET), 2013 IEEE 9th International Conference **DOI:** 10.1109/ICET.2013.6743551

[14]. Avner Pereira, Marcel Alves,Hendrik Macedo, Vehicle driving analysis in regards to fuel consumption using Fuzzy Logic and OBD-II devices.Telematics and Information Systems (EATIS), 2016 8th Euro American Conference **DOI:** 10.1109/EATIS.2016.7520160

[15]. EOBD/OBD2 Simulator -Instruction Manual 739 660 - 10/14-W2010-Wei

[16]. Prajakta Kulkarni, P.K. Rajani , Kanupriya Varma,Development of On Board Diagnostics (OBD) testing tool to scan emission control system, Computing Communication Control and automation (ICCUBEA), 2016 International Conference **DOI:** 10.1109/ICCUBEA.2016.7860098