

Enhancement of Vehicle Behavior System

Dr. Mehul Barot, Chauhan Mahaveersinh, Dheer Wani

Abstract: The purpose of this research paper is to make the usefulness of the OBD-II port which is provided in the car. So, what is an OBD port? It is a on board diagnostics tool which acts as a self-diagnostic tool for the car. So here in this paper we will use the OBD port for the functionality. So, we will do this with the help of the Arduino board. The Arduino board is the one which communicates with the OBD port. Takes the data which is being analyzed locally. The main objective of doing this is that there is not much awareness regarding this feature which can be really useful to the owners to maintain their car and monitor their car remotely. We nowadays didn't see much use of this feature generally among the normal people. If we make such a feature available to the customer then there will be a lot more benefits to the customer. As only there are very low number of the people which make the use if these Port.

Keywords: OBD-II port, Arduino Board, Vehicle Behavior System, Hexadecimal

I. INTRODUCTION

So, what is Vehicle Behavior System? so it provides the real time information of the working of the car. So, the people can get the up-to-date information regarding the statistics of the machines inside the car.

The information which we get from OBD-II port provides information like the check engine, RPM, speed, number of the ignition cycles etc., there a lot of information which you can get from the port which useful to improve the performance of the car. So here the main objective of this is to create such a system which is feasible to the use by the consumers and easy to use.

The reason any OBD-II scan tool can read the codes is because of the standardized pinout. Scan tools can read from any of the protocols listed above. The standardized pinout is as follows. [2]

- **Pin 1:** Used by manufacturer
- **Pin 2:** Used by SAE J1850 PWM and VPW
- **Pin 3:** Used by manufacturer
- **Pin 4:** Ground
- **Pin 5:** Ground
- **Pin 6:** Used by ISO 15765-4 CAN
- **Pin 7:** The K-Line of ISO 9141-2 and ISO 14230-4
- **Pin 10:** Used only by SAE J1850 PWM
- **Pin 14:** Used by ISO 15765-4 CAN

Revised Manuscript Received on October 15, 2019.

Dr. Mehul Barot, H.O.D., Department of Information Technology, LDRP Institute of Technology and research, Gandhinagar, Gujarat, India. Email: m.p.barot@gmail.com

Chauhan Mahaveersinh, Student, Department of Information Technology, LDRP Institute of Technology and research, Gandhinagar, Gujarat, India. Email: chauhanmahavir86@gmail.com

Dheer Wani, Student, Department of Information Technology, LDRP Institute of Technology and research, Gandhinagar, Gujarat, India. Email: dheerwani2498@gmail.com

- **Pin 15:** The K-Line of ISO 9141-2 and ISO 14230-4
- **Pin 16:** Power from the car battery

There are lots of people which are unable to know that actually happens inside the car. There can be a case that if Accident or Some malfunctioning occur into the car, so with help of the OBD-II we can find out what actually happened inside the Machinery. So basically, what the OBD does that it takes a snapshot of the parameters at the time when the trouble event has been occurred. So, this information can be fetched by the user and can be used to identify where exactly the error has been occurred. So, it can easily be diagnosed and can easily be repaired.

II. EXISTING METHODOLOGY

Currently the use of OBD-II port mainly done by the car manufacture and resellers for calculating the actual resell value of the car on the basis of the information received from the OBD-II port. But the use of OBD can be done far more than the imagination. It can be even use by the consumers who are buying the car and are unaware of its features. As it allows you to know what is what actually is currently the engine load, ignition cycles etc.

We are here mainly focused on the travellers who rents their car to the people, so here we can inform the owner with the real time information of the car location, the fuel level, number of kilometers actually driven etc. So, there can't be any fraud done by the driver.

III. PROPOSED METHODOLOGY

For this thing to work required hardware equipment's like Arduino Board, OBD Connector, GPS.

On-Board Diagnostics (OBD) is an automotive term referring to a vehicle's self-diagnostic and reporting capability. OBD systems give the vehicle owner or repair technician access to the status of the various vehicle subsystem. OBD implementation use a standardized digital communications port to provide real-time data in addition to a standardized series of diagnostic trouble codes (DTCs) which allow one to rapidly identify and remedy malfunctions within the vehicle.[1]

With the help of the OBD connector we connect the OBD port with the Arduino. The Arduino is the main part where all the conversions and calculations occur.

So basically, all the information that the OBD port provides it's in the form of HEXADECIMAL. All the information that we get is basically in RAW format. So, we must have some way to convert this value into readable format and some formula so that it can be made useful.

Every value has been assigned with the PIN known as Parameter ids.



So how does all this works?

As the engine starts the OBD port starts to collect information from different sources. All of this information which is being collected out of all the relevant information which we required for the calculations we get from its particular PID and its being transferred to the Arduino, inside the Arduino all of this information is being converted into the relevant from and it sent remotely to the firebase.

In total there are almost 167 different information which we can get from the OBD port. So, we basically in our research tried to get some of the parameters to try our thesis, we collected information such as RPM, Speed, Engine Load, run time from engine starts.

Letters (the first section) identify the type of the malfunction: [3]

- Pxxxx for powertrain (Engine Transmission)
- Bxxxx for body (Also A/C and airbag)
- Uxxxx for User Network (UART/wiring bus)
- Cxxxx for Chassis (Also ABS)

A single number from 0 to 3 (the second section) shows whether the code is manufacturer-specific or not: [3]

- X0xxx and X2xxx for generic code or government requirement code
- X1xxx for manufacturer-specific code

A single number from 0 to 9 (the third section) shows what system the code refers to: [3]

- Xx1xx and Xx2xx for fuel and air measurements
- Xx3xx for ignition system (misfire)
- Xx4xx for emission system
- Xx5xx for speed/idle system
- Xx6xx for computer system
- Xx7xx and Xx8xx for transmission
- Xx9xx for input/output system

A pair of number from 00 to 99 (the fourth section) shows the specific failure code and are based on the system defined in the third character. [3]

The data which is calculated in the Arduino is being sent to the firebase, which is continuously updating and as soon as any new value is being calculated its being updated in the database also.

All this value is being shown to the user and can be used to diagnose and identify the issue.

This is the snap shot of Database.

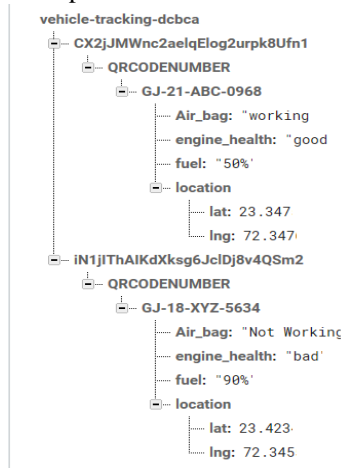


Fig.1.Database

IV. IMPLEMENTATION

In the table shown below consist of the PID, name of the parameter and the formula used to convert that value. [1]

Table- II: Name of the Table that justify the values

PID(HEX)	Description	Formula
02	Freeze DTC	-
04	Calculate Engine Load	A/2.55
0C	Engine RPM	(256A+B)/4
0D	Vehicle Speed	A
5E	Engine fuel rate	(256A+B)/20

The main part of the implementation was to retrieve all this data and put into some proper formula to get the results. So, here's below some main implementation part:

```

//Car Engine RPM
rpm=((rxBuf[3]*256)+rxBuf[4])/4; //Car Engine RPM
carspeed=rxBuf[3]; //Car Speed
Engine_load=rxBuf[3]/2.55 //Car Engine Load
Run_time_since_engine_start=(256*rxBuf[3])+rxBuf[4];//R
un time since engine start
Engine_fuel_rate=(100/255)*rxBuf[3]; //Engine fuel rate
    
```

This is the snap shot of the value taken from the OBD.

```

4RPM:
1154
Speed:
0
Engine Load:
41
Run time since engine start:
40
Engine fuel rate:
0
rxID:
7E8
rxBuf[2]:
CRPM:
1399
Speed:
0
Engine Load:
41
Run time since engine start:
40
Engine fuel rate:
0
    
```

Fig.2.Value Taken from OBD-II Port

V. CONCLUDING REMARKS

So, with the help of this research we can help the people identify real issues and with the help of the snapshot at the time when any trouble even occurred. And also, with the help of the check engine light we can notify the user that your car requires servicing or your airbag is working or not etc.

VI. FUTURE ENHANCEMENT

In the future up gradation, we can get some more information the port and get to know about how much ignition cycles it had gone through or from all the parameters we can calculate that in how much days that it will need a service.

REFERENCES

1. https://en.m.wikipedia.org/wiki/On-board_diagnostics
2. <https://www.makeuseof.com/tag/obd-ii-port-used/>
3. <https://www.carbibles.com/access-obd-ii-codes/>

AUTHORS PROFILE



Dr. Mehul P. Barot is done his Ph.D from C.U. Shah University. His publication details are More than 40 Publications in National as well as International Journals/Conferences. His research work is about 6-10 Post Graduate Research Projects has been Carried out in Data Mining, Big Data Analytics, Image Processing and NLP. He is the member of 'Annual Member of ACM and Life Time Membership of ISTE'. He has been working as the HOD of Information Technology in LDRP Institute of Technology and Research, Gandhinagar, Gujarat, India. He has teaching experience of more than 15 years.



Chauhan Mahaveersinh is pursuing his Bachelor of Engineering in Information Technology from LDRP Institute of Technology and Research, Gandhinagar, Gujarat, India. I have done many projects like (GIH116) Public Transportation System Management (Gujarat hackathon project), Vehicle Behavior System, Excel File Manipulation SRT to TXT convert using python. This all are project is done in python. I learn many coding languages like Python, C, C++, Computer vision with Python, Data Science with Python.



Wani Dheer is pursuing its Bachelor's in Information technology from LDRP Institute of Technology and Research, Gandhinagar, Gujarat, India. I had done projects in Hostel Management System, Bus ticket booking System, Public Transportation Management System (HACKATHON) and Vehicle behavior System. I like to do code in Arduino, C/C++, Java, Advance Java. he has done some project in java like Ticket Booking System GUI in Java.