Integrated OBD-II and mobile application for electric vehicle (EV) monitoring system

Conference Paper · October 2017

DOI: 10.1109/I2CACIS.2017.8239058

4 authors:

Kavian Khosravinia
Ontario Tech University

Mohd Khair Hassan
Universiti Putra Malaysia

Ribhan Abdul Rahman
Universiti Putra Malaysia

Syed Abdul Rahman Al-Haddad
Universiti Putra Malaysia

Integrated OBD-II and Mobile Application for Electric Vehicle (EV) Monitoring System

Kavian Khorsravinia, Mohd Khair Hassan, Ribhan Zafira Abdul Rahman, Syed Abdul Rahman Al-Haddad
Department of Electrical and Electronic Engineering
University Putra Malaysia,
Kavian.khn@gmail.com, khair@upm.edu.my, ribhan@upm.edu.my, sar@upm.edu.my

***Abstract*—Electric Vehicles (EV) are becoming more popular, advanced and connected to the internet, the users are being encouraged to use new technologies for tracking, monitoring and controlling their EVs. The experienced users tend to monitor and control specific EV parts including ABS, Airbag, and Air Condition easily and seamlessly. Currently, there is no a distinct system at all that would integrate all these factors into one disposable application. This research develops a system which communicates with On Board Diagnostics of Electric Vehicle by using Controller Area Network communication protocol. The paper discusses how to collect CAN data by new and low-cost method through OBD-II port. The OBD-II is traditionally used by technicians for reading fault codes and monitoring the actual value of each connected control units. There are few other solutions already available, but none of them satisfies requirements for experienced users. Ordinarily, these available solutions can be applied for monitoring the state of charge and battery parameters. The extraction of all the available data in EV CAN network has been done systematically through OBD-II port and by using low-level software libraries and programming a development board based on ARM STM32F103RBT6 micro-controller. This paper discuss an implementation an integrated OBD-II system for monitoring CAN data such as SOC and Energy consumption based on Android. Finally, the CAN data (e.g., lock and unlock door and Air condition control) will be injected in EV CANs system. Overall, the proposed approach can facilitate the monitoring and control processes for most of specific EV parts considering flexibility and ease of use for the end users.**

Keywords*—*Electric Vehicle CAN, CAN protocol ,EV OBD-II, ARM CortexM-3, Inject CAN Data

I. Introduction

Today, one of the important issues that developed countries face with is undoubtedly climate change. Therefore, they are forcing car companies to produce vehicles, more environmental friendly than vehicles with combustion engine. In order to reach this goal supplier focus on Electric Vehicle (EV) which does not produce exhaust gassing during operation. This makes EVs interesting as one of the rapidly growing research area, because the percentages of the people which are use EVs increase dramatically every year. Further, vehicles are becoming more advanced and connected to the internet with the microprocessor revolution, mobiles have become cheaper and useful, and so enterprises are being encouraged to use these components for tracking and monitoring their vehicles.

Recently, On Board Diagnostic (OBD-II) connectors have been introduced to use a vehicle network for diagnostic and checking the real-time parameter of all the electronic control units such as Engine, Airbag, ABS and TCU. There are five communication types using this layout: J1850VPW, ISO9141-2, KWP2000, J1850PWM and ISO15765 are set by metal contacts. Basically, if a problem occurs, the OBD system generates a trouble code which could be identified by a service engineer by connecting an OBD scan tool and fixing the problem. After 2008, all the car companies are required to implement CAN as the communication protocol for the external OBD interface.

Todays, most of vehicles has been equipped with Controller Area Network (CAN) bus, the modern automobile may have as many as 70 electronics control units for various subsystems [1,2]. The CAN protocol has been designed by Robert Bosch in 1986 for automotive applications. It is a serial communications protocol designed to allow microcontrollers and devices to communicate with each other without host computer. In fact, all these electronic units are connected using CAN with various bitrates up to 1 Mbit/s. It has been heavily involved to the EVs application. Most EVs rely on CAN bus to link their batteries, energy management and electric drive train.

This paper is organized as follow: In Section II, the main problem has been discussed. In Section III, we have reviewed the literature including current and previous related work. Section IV illustrates CAN bus protocol. Section V discusses the details of our proposed approach, introducing the device and software which are used for running the project. Section VI, explains how the entire system works. Section VII presents the experiment results and finally Section VIII provides the conclusion.

II. PROBLEM STATEMENT

Despite the growing interest in using EVs in recent years, the users always under pressure about the status of the electrical system of their cars. For example, they tend to monitor the state of charge (SOC), the state of health (SOH), and information on each of the 96 cell-pairs in the pack, as well as the KWH remaining in in the battery pack. The further example of this information includes: assessing the energy consumption since starting the car and distance to empty based on the users personal setting of efficiency. All the above examples are used for monitoring the electric status of the vehicle, but there are some important units which connected to EV section and if any error occurred in these systems, it would defect the whole system of the EV. For example, the user can have some specific parameter of the gearbox of the vehicle (e.g., oil temperature and generator status which is controlled by TCM (Transmission Control Unit).

On the other hand, some user tends to control their EVs by the remote application. In EVs, the air condition system can run by electricity voltage directly from batteries in the stationary position by the driver even without run completely the system. Thus, the driver can send a request to the air condition unit remotely for running the air compressor of the air condition system to modify the temperature of the passenger's cabin before using the vehicle. In fact, there is no distinct system at all that would integrate all these factors into one disposable application. The main objective of this research is to build software application by some special features and functionality which is more beneficial for EVs users.

III. LITREATURE REVIEW

OBD-II system has been mandatory equipment for diagnostic emission performance and vehicle maintenance and products for reading information from the OBD interface are common and widely used the vehicle service engineer.

Research groups commonly use commercial devices to collect CAN data through OBD-II port. For example, [3] used neoVI Fire, a commercial CAN sniffing tool, for recording and extracting of CAN packets. Accordingly, the CAN packets have been used for Carbon Footprint Evaluation by comparing CO_2 emission of both EVs and combustion vehicles (CVs). They have read CAN packet from data link layer and have used two different methods to manipulate the CAN bus by injecting random or specific CAN IDs into a layer for vehicle control.

Another project, Nissan Leaf On Board Diagnostic [4]. Implemented AVR-CAN board which was used to receive the CAN packet through OBD and transmit to Android application by Bluetooth for monitoring SOC, Voltage, Current and Rpm in real-time Android application. They also use CAN-DO box data that has already been produced by John Dunning for specifically Nissan Leaf and connected to OBD port by wire. However, the project has focused in three CAN IDs and it is not clear how to collect them.

The implementation of integrated OBD-II connector with external network [5] transmits real-time driving valuable information of a vehicle. The protocol consists of OBD dongle ELM327, OBD chip STN1120, Cortex-M3, Wi-Fi module, Bluetooth module and WCDMA module for long and short distances. The central MCU (Cortex-M3) transmits the specific PID (00 01) to the OBD microchip. The OBD microchip recognizes the vehicle protocol among the five existing ones, and transmits the PID to the vehicle network. Accordingly, The OBD microchip get responses from the network, transmits data to the MCU. The MCU transmits data to an external device by using variety of methods based (e.g., Bluetooth, Wi-Fi, etc.) on distances. However, they bought OBD interpreter like ELM327 for automatically send PID and the ELM327 already designed and programmed. In this project they used commercial tools.

Our research covers much bigger scope than the Nissan Leaf On Board Diagnostic and our goals are different. First, we are interested in investigating EVs CAN bus network in order to distinguish between different CAN bus EVs and CVs. As part of our research we will code an ARM development board which is used for collecting and injecting CAN packets through OBD-II link layer. Second, we aim to monitor data and collected android devices by implementing a low cost and easy-to-incorporate and efficient approach.

IV. CAN BUS PROTOCOL

In 1991 the CAN bus was the first bus system to be introduced to a motor vehicle in mass production. The CAN is a serial communication protocol which efficiently supports distributed real time control with a very high level of security [6].The protocol is also widely used today in industrial automation and other areas of networked embedded control. When using CAN as communication protocol there are three relevant standard listed in "*Table I*".

Table I. CAN COMMUNICATION PROTOCOL

CAN Communication Protocol	
Standard	**Description**
ISO 11898	CAN
ISO 15765	Diagnostic on CAN
ISO 15031	Legislated OBD on CAN

ISO 15765 defines the requirements for vehicle diagnostic systems implemented on a CAN bus [7]. This standard illustrates the layered services and specifies the ISO standard that describes the service. The Open System Interconnection (OSI*)* model can be planned to the following layers:

- Application Layer: ISO 15765-3
- Transport Layer: ISO 15765-2
- Network Layer: ISO 15765-2
- Data Link Layer: ISO 11898-1 and ISO 15765-4
- Physical Layer: ISO 11898-2 and ISO 15765-4

A. NORMAL CAN PACKETS

In ISO 11898-1 illustrated CAN bus as a protocol in the Data Link Layer of the OSI model. The messages that are transmitted to CAN bus are called frames and there are four different frame types: Data frames, Remote frames, Error frames and Overload frames [8].

Data frame is the type of the messages that transfer data on CAN bus. A data frame is consisted of seven different bit fields: Start of Frame, Arbitration Field, Control Field, Data Field, CRC Field, ACK Field and End of Frame. The frame begins with a SOF bit and is followed by arbitration ID which can be 11 bit that are called Standard frame, using 11 bit for node identification or can be 29 bit that are called Extended frame using 29 bit for node identification. The identifier bit is used as a priority field, the lower hex value in the bit, means the higher priority. The structure of the CAN data frame is displayed in *"Fig. 1-1"* and *"Fig. 1-2*". Data field contain of a minimum of 0 to a maximum of 8 bytes, the data maybe contain checksums or other mechanisms.

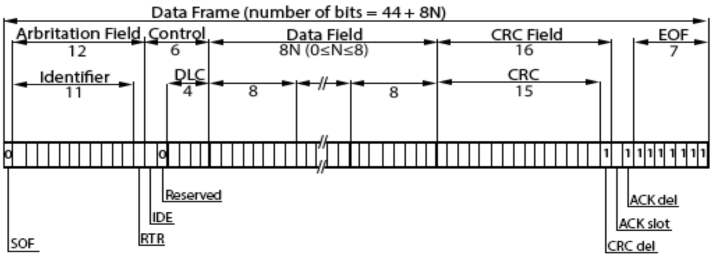

Figure 1-1. CAN data frame using 11-bit addressing

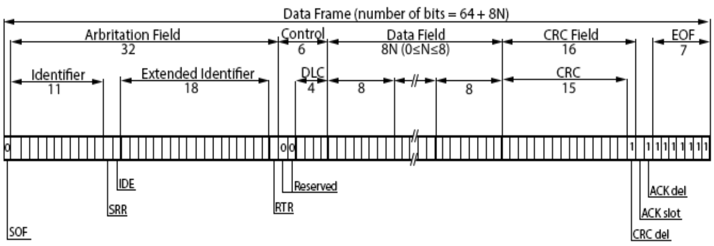

Figure 1-2. CAN data frame using 29-bit addressing

B. DIAGNOSTIC CAN PACKETS

The other type of CAN packets in automotive is diagnostics packets. ISO-TP or ISO 15765-2 is a standard that describes how the data sent over a CAN bus and how it operates in Transport Layers of the *OSI* model. These packets are sent by scan tools and control unit which response them. Since normal CAN frames can only contain eight bytes, for sending more than 8 bytes need to break up bytes by sending node and reassembled by receiving node which is why ISO 15765-2 is designed [9]. In ISO-TP the beginning byte contains PCI (Protocol Control Information). The first byte broken into two portion and the first four bits are the PCI type which are as follows [10]:

- 0 – Single Frame
- 1 – First Frame
- 2 – Consecutive Frame
- 3 – Flow Control Frame

The below is an example:

ID: 07 60, Len: 08, Data: **03** 14 FF 00 00 00 00 00

V. METHODOLOGY

The methods for solving the problem need to select high performance and low-cost device. First of all, we need CAN bus Sniffing tools, which can be used for reading and writing CAN packets over vehicle network. STM32-P103 board is development board which allows you to explore the feature of the ARM Cortex-M3 STM32F103RBT6 microcontroller. There is CAN transceiver SN65HVD230 which allows CAN application be developed.

In order to program the development board the language that used is *C* and the user need to program their own interpretation firmware by using Keil uVision4 and transfer it to the board through a Universal Serial Bus (USB) debugger connected to the Joint Test Action Group (JTAG*)* connector on the board.

A. Bluetooth module

One of the significant wireless communications is Bluetooth network which is short-range wireless technology that operates in 2.4GHz band and low-cost and low power technology [11].

B. App Inventor for Android

MIT app invertor is an intuitive, visual programming environment that allows to build fully function apps for android phones and tablets. Furthermore, app is a web interface development environment and then can be uploaded directly on smartphone.

VI. DESIGN

The developed integrated OBD-II mobile application consists of some components and software that have been illustrated below. "*Fig. 1-3"* the experimental layout of project.

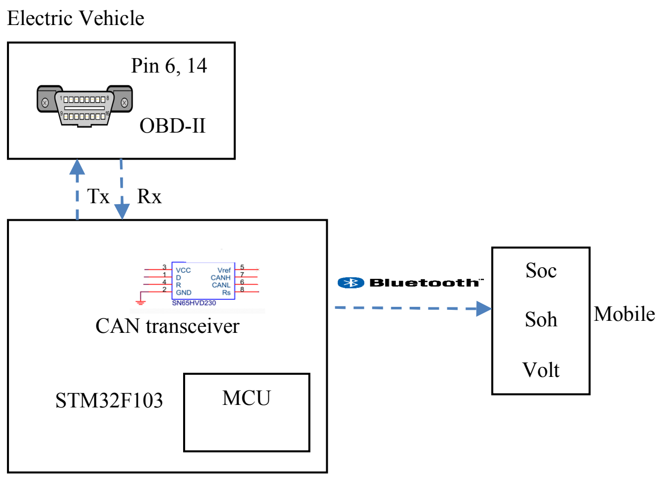

Figure 1-3. Experimental Layout

Development board connects with the car through OBD-II, and smartphone connects with development board via Bluetooth. STM-P103 board can be set by OBD commands and the real-time data, Diagnostic Trouble Code (DTC) are acquired by OBD command. The OBD command which is used is called Mode [12], and all modes and PIDs are described in documents such as the SAE J1979. For collecting data the mobile send specific mode (byte) via Bluetooth to STM-P103. The STM-P103 after receiving the data byte transmits the data byte by Tx line to OBD-II port to intended pin. The control unit receives the data byte and response as predefine standard and condition of the vehicle. The response data is transmitted by Rx line to the STM-P103. Finally, STM-P103 transmits data via Bluetooth to mobile for displaying.

VII. RESULT

A pre-validation experiment has been performed in the Auto Ecu service center in two different methods for collecting data. The first method, on Porsche hybrid 3000cc by connecting, CAN port of the development board to the OBD-II port and second method, collect data on the bench which means connect the particular control unit to the power and connect the CAN-H and CAN-L of the development board to intended pins of the control unit. The communication structure is shown in "*Fig. 1-4*"

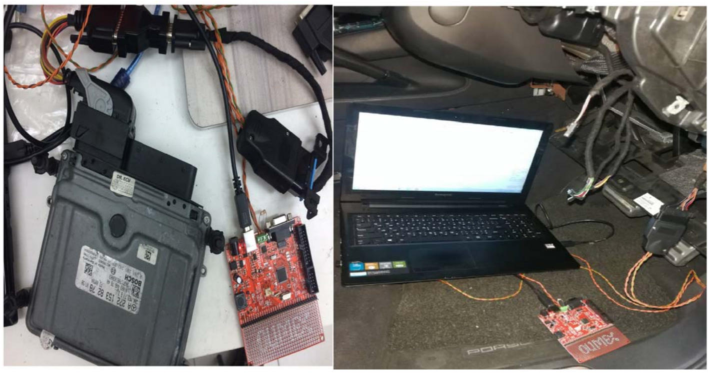

Figure 1-4. The Communication Structure

The first packet sent to the control unit by STM32fF103 is a first frame data with (000006F1) ID (12 03 19 02 0C) to the specific control unit for reading DTC (Diagnostic Trouble Code). Indeed, this is a specific byte for requesting DTC according to the modes and PIDs which are discussed in previous Section (03) is the specific byte for requesting DTC. "*Fig. 1-5"*shows a standard frame. After a few millisecond, the control unit with (00000612) ID (F1 10 53 59 02 FF 10 30) responds the request. The next frame is an acknowledgement frame (12 30 00 00). It is followed by four frames that start with F1 (common byte for showing the presence of the tester in the diagnostic process).These are consecutive frames because the second byte starts with (2) and with indicators 1, 2, 3 and 4. In the data line (F1 21 08 AF 10 31 08 AF), the fifth byte (10) indicates that the DTC belongs to power train system. Similarly, sixth and the seventh byte respectively (31 08) show the actual DTC. The received message will be transferred to the mobile device by Bluetooth device which has been installed on development board. "*Fig. 1-6"* demonstrates an actual DTC.

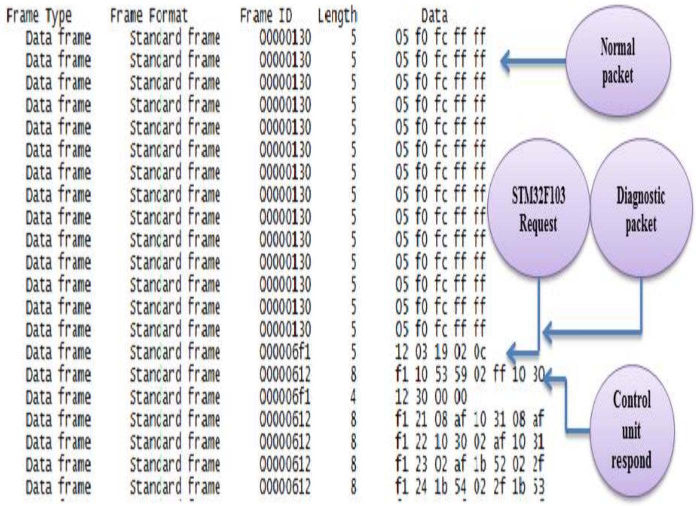

Figure 1-5. The Message Format

Figure 1-6. The Actual DTC

VIII. CONCLUSION

In this paper the integration of hardware and software in the vehicle is demonstrated. The CAN protocol IDs are monitored as such thus interested signals are identified. All the major data is available in the CAN bus serial communication with two single wire. The possibility to integrate new application in mobile with the vehicle CAN using OBD-II for diagnosing is investigated. This newly developed mobile application prepares a secure and convenient interface for car users with a lot of new features.

In the future, we will use all the useful data and can be share them with server to monitor more information including accident scene for requesting help, traffic congestion and vehicle position.

References

[1] S. Corrign, "Introduction to the Controller Area Network (CAN)", Application Report, Texas Instrument, Sloa101b, May 2016

[2] Mohd, T.A.T, Hassan, M.K. Aziz, W.M.K, "A Mathematical modeling and simulation of an electric vehicle", Journal of Mechanical Engineering and Sciences, Vol.8, 1 June 2015, Pages 1312-1321

[3] C. Tseng, W. Zhou, M. Al Hashim,C. Chau, S. Song, E. Wilhem, "Data Extraction from Electric Vehicles through OBD and Application of Carboon Footprint Evaluation" , EV-Sys, June 21-24 2016, Waterloo, ON, Canada

[4] J. Allande, P. Tyler, and E. Wodruff, "Nissan Leaf On-Board Diagnostic Bluetooth Utility",Interdisciplinary Senior Design Project, 2013

[5] S. Baek, J. Jang, "Implemention of Integrated OBD-II Connector with External Network", Science Direct Journal, 2014

[6] R. Bosch, "Automotive Handbook", 5rd ed, vol.4.Wiley,2004, pp. 92-100

[7] S. Vincent, "Development of a CAN Based Electric Vehicle Control Syestem", Technical Paper, University of Kansas, June 24, 2014

[8] C. Valasek, C. Miller, "Advantures in Automotive Networks and Control Units" , Technical White Paper, pp. 6-10, 2014

[9] C. Valasek, C. Miller, "CAN Message Injection" , Technical White Paper, Cyber Crime Division, June 28, 2016

[10] M. Kabala, J. Gamek, "Wireless Real-Time Vehicle Monitoring Based on Android Mobile Device" , Versita, Emerging Science Publishers, Vol 12, No 4, 2012

[11] N. Sriskanthan, F. Tan, A.Karande, "Bluetooth Based Home Automation System" , Science Direct Journal, Microprocessors and Microsystem, 2002

[12] Y. Yang, B. Chen, L. Su, and D. Qin, "Research and Develop of Hybrid Electric Vehicles CAN-Bus Data Monitor and Diagnostic system through OBD-II and Android-Based Smartphones" , Hindawi Publishing Corporation, vol. 2013