Proceedings of the International Conference on Inventive Computing and Informatics (ICICI 2017)
IEEE Xplore Compliant - Part Number: CFP17L34-ART, ISBN: 978-1-5386-4031-9

ECU Diagnostics Validator Using CANUSB

Abhijit D. Sutar[1*], Sagar B.Shinde[2]
[1] M. E Scholar , Department of E&TC, JSPM, NTC, Narhe, Savitribai Phule Pune University, Pune, India
[2]Assistant Professor, Department of E&TC, JSPM, NTC, Narhe, Savitribai Phule Pune University, Pune, India
e-mail: jitabhisutar@gmail.com, sagar.shinde5736@gmail.com

***Abstract*— An ECU diagnostics provides information about ECU behavior which in turn provides the overall health of the vehicle. An OBD (On-Board Diagnostics) takes necessary corrective actions for preventing accidents and hazards by detecting vehicle system's malfunction. The driver and passenger safety can be taken care with the help of diagnostics. For ECU diagnostics validation, most of the OEMs, Tier-1, and Tier-2 use a Vector Tool "CANoe" during ECU development phase. The CANoe is very powerful tool. However, its cost is ~ €10,000. The proposed system can be used to perform the similar functionality of ECU diagnostics validation during ECU development phase. It uses CANUSB as hardware interface and software developed in Microsoft C# language as application software. It is the cost-effective special tool which uses single CAN channel for transmitting and receiving various CAN messages to/from ECU. Using this tool, the diagnostic messages and vehicle messages can be simulated & sent and ECU response can be validated. The tool would be having user-friendly easy to use GUI, manual as well as cyclic auto transmission mode with message logging feature. The overall tool cost would be ~ ₹35,000.**

Keywords*— *CAN, CANoe, CANUSB, Diagnostics, ECU, OBD, OEMs, and UDS

I. INTRODUCTION

With advancements in automotive electronics, the older day's mechanical systems are replaced with electronic systems. The electronics device which monitors and controls the vehicle systems is called as an ECU. An Electronic Control Unit (ECU) is a device which uses embedded systems having a microcontroller to control the various vehicle features. The vehicles are becoming more advanced, semi-autonomous, and connected. Vehicles are heavily regulated by law for consumer comfort & safety, to reduce emission and increase fuel efficiency. Since the late 90s; the vehicles are built with ECUs having on-board diagnostic systems to monitor vehicle health. The on-board diagnostic system uses OBD-II port and a related subsequent standard. This port allows access to the vehicle to monitor the performance and function. An ECU involves complex algorithms to control various critical and safety features. To provide more comfort and safety, the Original Equipment Manufacturers (OEMs) are focusing on to produce ISO 26262 Safety Standard compliant ECUs. To achieve this compliance, one of the major areas is On-board Diagnostics using CAN. An OBD (On-Board Diagnostics) takes necessary corrective actions for preventing accidents and hazards by detecting vehicle system's malfunction. The proposed system can be used to perform the functionality of ECU diagnostics validation during ECU development phase. Thus it enables the driver and passenger safety.

II. LITERATURE REVIEW

The late 80s, vehicles equipped with control systems which can alert the driver about a malfunction and allow the technician to retrieve DTC codes that identify a malfunction in the vehicle. It helped to reduce emission and guide the technician to diagnose the fault. OBD system continuously monitors vehicle feature behavior for correct functioning. The monitoring is of component monitoring and system monitoring. The Malfunction Indicator Lamp (MIL) is illuminated if any fault is detected. The fault is stored in ECU memory by saving the information about the fault in form of DTC and freeze frame. This information enables the technician take appropriate action to remove fault.

A. Automotive Diagnostic Gateway using Diagnostic over Internet Protocol

Young Seo Lee et al proposed the diagnostic gateway that supports CAN, FlexRay and Ethernet protocols. The diagnostic gateway can be connected to the external diagnostic device through the Transmission Control Protocol (TCP) and provides the reliability and integrity of the diagnostic data through the TCP connection. The diagnostic services using CAN-based diagnostic protocols take too long because of the low bandwidth. In contrast, Ethernet-based DoIP provides high bandwidth and large data sizes for large-scale communication, and it can diagnose the vehicle in near real-time [1].

B. On-Board Diagnostics (OBD) Scan Tool to Diagnose Emission Control System

Prajakta Kulkarni et al described emission compliance requirement with a brief introduction of the OBD system along with scan tool to diagnose the system. The requirement of aftertreatment emission control system and its different types are discussed. A brief history of OBD has been presented. The main parameters like Diagnostic Trouble Codes, Malfunction

Indicator Lamp, Freeze Frame and Parameters ID are discussed. The paper explains in detail the process of self-diagnosis technique for emission control system by using the scan tool. This paper helped to understand various diagnostics tool available for the ECU diagnostics [2].

C. Design of an In-Vehicle Network (Using LIN, CAN and FlexRay), Gateway and its Diagnostics Using Vector CANoe
Rishvanth et al described the design of In-vehicle network and Gateway using Vector tool CANoe. The simulated vehicle network provides an efficient method of communication between various ECUs. The papers discuss the communication protocols - Local Interconnect Network (LIN), Controller Area Network (CAN) and FlexRay. These communication protocols are used for low, medium and high-speed applications respectively. A gateway is a network node and acts as a bridge to transfer data from one communication protocol to another. Vector CANoe is OEM recommended a very powerful tool for the development, testing and analysis of entire ECU networks and individual ECUs. The paper explains in detail how to create 3 different networks (LIN, CAN, and FlexRay) and also how to design a gateway so that the messages can be transferred between different communication protocols [3].

D. Diagnosis in Automotive Systems: A Survey
Patrick E. Lanigan et al surveyed the technical literature for failure diagnosis approaches of a vehicle. It emphasizes the diagnosis approaches suitable for the emerging class of distributed, embedded control-systems for convenience, safety, and autonomous operation of automobiles. This survey addresses both the online failure diagnosis approaches and offline failure diagnosis approaches. The online failure diagnosis is necessary to maintain the safe operation of the vehicle at operation/run time. The offline failure diagnosis is suitable for maintenance of the vehicle in a service facility. This survey gives a candid and critical look at the current state-of-the-art, highlighting where the field practices fall short of accomplishing the goals of safety-critical autonomous driving systems [4].

F. Automotive Diagnostics Development Seminar by Jeff Craig emphasized the evolution of automotive diagnostic technology, Diagnostic Standards, Transport Protocol, UDS (ISO 14229), running UDS on Different Buses (J1939, Ethernet) [5].

G. Monitoring and Control System for Industrial Parameters Using CAN Bus
SargunaPriya.N et al described the system to monitor and control the various parameters such as pressure, temperature, and water level in the industry. The system uses PC, various sensors, and PIC microcontroller. The system provides an automatic control of various industrial parameters with the help of CAN bus application [6].

H. Design and Development of a Vehicle Monitoring System Using CAN Protocol
Mohammed Ismail et al described the prototype system to monitor real-time parameters of the vehicle using CAN protocol. It uses PIC microcontroller with its ADC module to gather data from analog sensors and converts it into to digital format. Then this information is displayed to vehicle driver through LCD display. It uses CAN bus as communication model which has efficient data transfer [7].

I. Vehicle Health Monitoring System
M.Jyothi Kiran et al described an in-vehicle embedded system which can generate Vehicle Health Report (VHR) as per user need. The data required for VHR can be obtained using OBD-II protocol. The system uses LabVIEW platform which has automotive diagnostics command set toolkit [8].

J. Interfacing CAN Bus With PIC32 Microcontroller for Embedded Networking
Umesh Goyal et al discussed on how to use the PIC32 microcontroller to develop CAN bus for embedded networking. It uses PIC32MX795F512L and IS01050 ICs to implement CAN bus. Also, the paper emphasizes the CAN protocol frame format and its working [9].

K. Road vehicles — Unified diagnostic services (UDS) — Specification and requirements
ISO 14229 Second Edition. This is an ISO standard 14229. It states the specification and requirements of Unified Diagnostics Services (UDS) protocol. It is must understand standard to get the knowledge of CAN diagnostics communication [10].

III. DIAGNOSTICS TOOL SUMMARY

The Table1 shows the comparative summary of various hardware and software available for diagnostics tool.

Table 1: Diagnostics Tool Summary

Parameters	CANoe with CANCase XL	PCAN	CANUSB
Environment	Hardware + Software	Hardware + Basic Software	Only Hardware. Application SW needs to be developed
Protocol Supported	CAN, LIN, XCP	CAN	CAN
Max Speed	1 Mbps	1 Mbps	1 Mbps
CAN Channel	2	2	1
Power	USB	USB	USB

Support ID	11/29 Bit	11/29 Bit	11/29 Bit
PC Isolation	Yes	Yes	Yes
Tool SW Access	No	Yes	Yes
CAN Controller	Phillips SJA 1000	NXP SJA1000	Philips SJA1000
CAN transceiver	Piggybacks	NXP PCA82C25 1	Philips 82C25
Operating Temperature	-20 to 70 °C	-40 to 85 °C	-40 to 85 °C
Cost	~ € 10,000	~ € 180	~ $ 120

IV. PROPOSED SYSTEM BLOCK DIAGRAM

The proposed system consists of following blocks explained hereunder in figure 1.

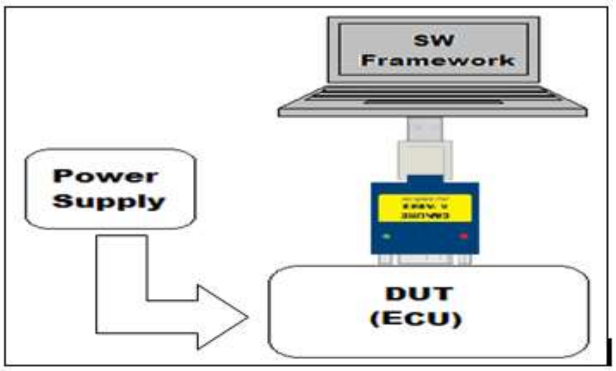


Figure 1. Proposed System Block Diagram

CANUSB: The system uses the LAWICEL CANUSB hardware as an interface between host PC and DUT.CANUSB does not need external power supply. It uses the power supply from host PC. The CANUSB supports both the 11bit as well as the 29bit CAN message ID format, RTR frames, built in FIFO queues and extended info/error information.

SW Framework: It is the application software developed using Microsoft C# language and DLL. It enables the user to transmit and receive the CAN messages to/from DUT using CANUSB. The figure2 shows the UI and Architecture of SW framework at a high level. It has GUI which enable the user to customize the CAN message request to be sent. The middleware & event handling handles all the working logic of the proposed system. The CAN device drivers take care of interaction between CAN bus and the middleware.

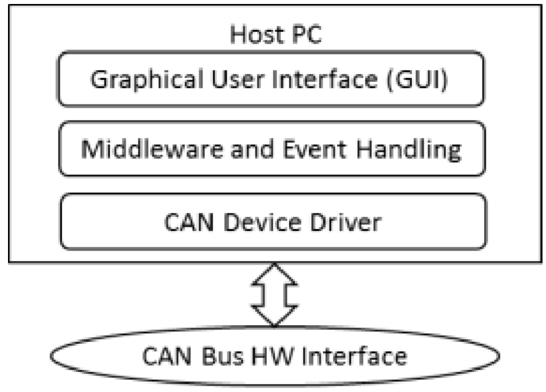


Figure 2. UI and Architecture

V. PROPOSED SYSTEM WORKFLOW

The figure3 shows the flowchart of the proposed system.

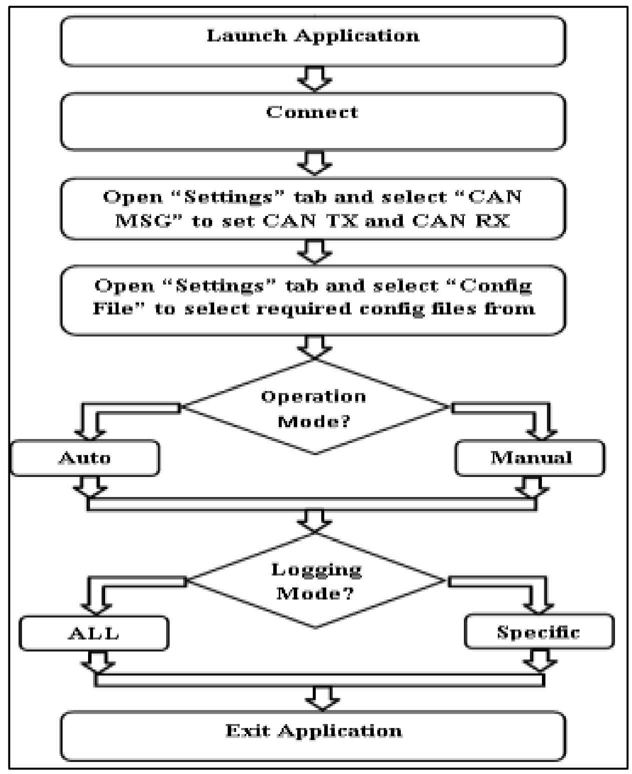


Figure 3. Flow Chart

VI. RESULTS

The proposed system is developed as per the requirements gathered. The figure4 shows the ECU Diagnostics Validator GUI. It has two operational modes; Manual operation mode and Auto operation mode. In manual mode, the user needs to fill-up all the fields of the message to be transmitted. The fields are 'DLC', 'MSG ID' and 'MSG Data bytes B0-B7'. Once all the fields are filled-up, the user needs to press 'Now' button to send the message. To send the message cyclically, the user needs to specify cycle time in 'Time' field and tick the checkbox present to the right side of the 'Now' button. In Auto Mode, the messages are transmitted automatically. For this, the user has to create a file by filling up all the fields of messages to be transmitted. The tool reads the messages from the file and sends these messages on bus automatically.

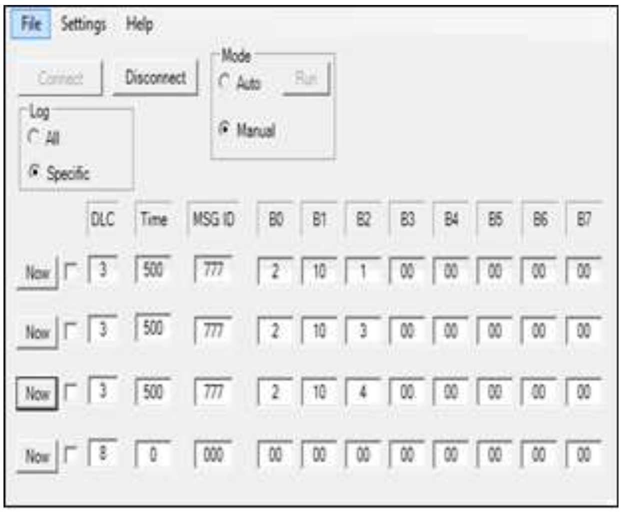
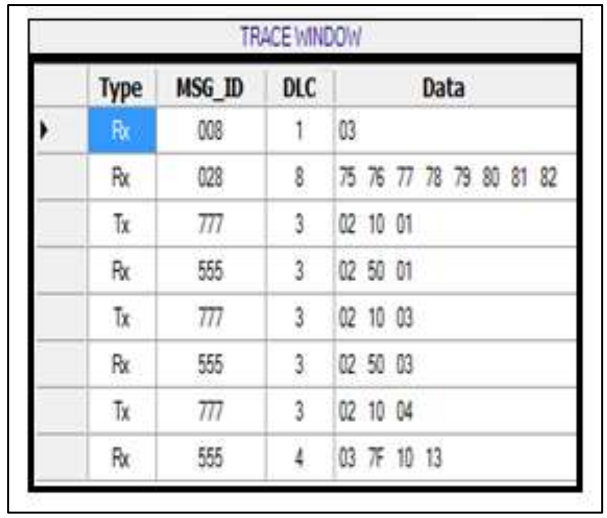


Figure 4. ECU Diagnostics Validator GUI

The figure5 shows the trace window of ECU Diagnostics Validator. It has various fields of message like 'Type', 'MSG ID', 'DLC', and 'Data'.



TRACE WINDOW

Type	MSG_ID	DLC	Data
Rx	008	1	03
Rx	028	8	75 76 77 78 79 80 81 82
Tx	777	3	02 10 01
Rx	555	3	02 50 01
Tx	777	3	02 10 03
Rx	555	3	02 50 03
Tx	777	3	02 10 04
Rx	555	4	03 7F 10 13

Figure 5. ECU Diagnostics Validator Trace Window

Using the Trace Window, the user can monitor the CAN communication happening on the CAN bus. The TraceWindow helps to analyze the Diagnostics Request Response. Thus it enables to perform ECU diagnostics validation.

The trace window is showing:

1. Communication Started

(0x008) – 01

(0x028) – 75 76 77 78 79 80 81 82

The ‘ECU A’ and ‘ECU B’ forms the CAN BUS network. An ‘ECU A’ sends the message 0x008 and an ‘ECU B’ sends the message 0x028. The appearance these messages on trace window indicates that the CAN communication is established between ‘ECU A’ and ‘ECU B’.

2. Diagnostics Request (0x777) - Diagnostic Default Session 02 10 01

3. ECU Positive Response (0x555) - 02 50 01

The message 0x777 is a diagnostics request from ‘ECU diagnostics validator’ tool to ‘ECU A’. Hence the message’s ‘Type’ is ‘Tx’ and it can be seen in trace window. In response to this request, an ‘ECU A’ sends a positive response 50. As this message is received by ‘ECU diagnostics validator’ tool, the message’s ‘Type’ is ‘Rx’ and it can be seen in trace window. This is a valid request from tool hence ‘ECU A' changes the diagnostics session to Default session. The diagnostics session is indicated by message 0x008. The value 01 indicates that it is Default session. All these response logic is written in ‘ECU A’.

4. Diagnostics Request (0x777) - Diagnostic Extended Session02 10 03

5. ECU Positive Response (0x555) - 02 50 03

The message 0x777 is a diagnostics request from “ECU diagnostics validator” tool to ‘ECU A’. In response to this request, an ‘ECU A’ sends a positive response 50 and changes the diagnostics session to Extended. The diagnostics session is indicated by message 0x008. The value 03 indicates that it is Extended session. All these response logic is written in ‘ECU A’.

6. Diagnostics Invalid Request (0x777) - 02 10 04

7. ECU Negative Response (0x555) - 03 7F 10 13

The message 0x777 is a diagnostics request from ‘ECU diagnostics validator’ tool to ‘ECU A’. This is invalid request as per the logic is written in an ‘ECU A’. So in response to this request, an ‘ECU A’ sends a negative response 7F.

VII. CONCLUSION

The “ECU Diagnostics Validator” tool is successfully designed and developed along with ‘ECU A’ and ‘ECU B’. The ‘ECU A’ and ‘ECUB’ suffice the need of CAN BUS ECUs network. This paper gives an overview of various Diagnostics tools, methodologies used by diagnostics tools, and Evolution of Automotive Diagnostic Technology. The paper explains the workflow of the “ECU Diagnostics Validator” tool and its principle of operation. Using “ECU Diagnostics Validator” tool, an ECU diagnostics validation can be performed. Thus it helps to develop safety compliant robust ECUs, ultimately enables the driver and passenger safety. The “ECU Diagnostics Validator” tool can be enhanced further to be used as 'Vehicle Health Monitoring System'.

REFERENCES

[1] Young Seo Lee, Jin Ho Kim, and Jae Wook Jeon, "*Automotive Diagnostic Gateway using Diagnostic over Internet Protocol*", IEIE Transactions on Smart Processing and Computing, Vol.**3**, no.**5**,pp.**313-318**, **October 2014**.

[2] Prajakta Kulkarni, Kanupriya Varma and Prof. P. K. Rajani, “*On Board Diagnostics (OBD) Scan Tool to Diagnose Emission Control System*”. IJRITCC, Vol.**4**, no.**5**,pp.**554-558**, **May 2016**

[3] Rishvanth, D. Valli2, and K. Ganesan, “*Design of an In-Vehicle Network (Using LIN, CAN and FlexRay), Gateway and its Diagnostics Using Vector CANoe*”, AJSP, Vol.**1**, no.**2**, pp.**40-45**, **October 2011**

[4] Patrick E. Lanigan, Soila Kavulya and Priya Narasimhan, Carnegie Mellon University, Thomas E. Fuhrman and Mutasim A. Salman, General Motors Research & Development “*Diagnosis in Automotive Systems: A Survey*”, CMU-PDL, **June 2011**

[5] Vector, “*Automotive Diagnostics Development A Technical Seminar*", **June 2014**

[6] SargunaPriya.N, Jothi Mani.M, Amudha.P “*Monitoring and Control System for Industrial Parameters Using Can Bus*”, IJETT, Vol.**9**, no.**10**, pp.**479-484**, **Mar 2014**

[7] Mohammed Ismail. B, K. Sasidhar, Syed Aquib “*Design and Development of a Vehicle Monitoring System Using CAN Protocol*”, IJARCCE, Vol.**4**, Issue**11**, pp.**386-390**, **November 2015**

[8] M.Jyothi Kiran, S.Ravi Teja“ *Vehicle Health Monitoring System*”, IJERA, Vol.**2**, Issue**5**, pp.**1162-1167**, **October 2012**

[9] Umesh Goyal, Dr. Neelam Rup Prakash “*Interfacing CAN Bus With PIC32 Microcontroller For Embedded Networking*”, IJERA, Vol.**3**, Issue**4**, pp.**689-693**, **August 2013**

[10] ISO 14229, *Road vehicles — Unified diagnostic services (UDS) — Specification and requirements*, Second Edition 2006-12-01

[11]