Test Automation Tool for Electronic Control Unit's Software Testing

[1] Rajashree M Bhide, [2]Pratiksha Raut, [3]Rohini S Jadhav,[4] Vaishali S Kulkarni, [5] Sunil L Tade

[1] PG Student, Smt. Kashibai Navale College Of Engineering, Pune,
[2] PG Student, Pimpri Chinchwad College of Engineering, Pune,
[3] Electronics Engineer-Senior, Cummins Technologies India Private Ltd., Pune,
[4] Assistant Professor, Smt. Kashibai Navale College Of Engineering, Pune,
[5] Professor, Pimpri Chinchwad College of Engineering, Pune

[1]rajashreebhide@gmail.com, [2]pratikshar142@gmail.com, [3]rohini.jadhav@cummins.com, [4]vaishsan2000 @gmail.com, [5]suniltade_2006@rediffmail.com

Abstract— With the technology scaling down, reduced device sizes have enabled us to deploy a wide range of electronic devices in a small area. The Electronic Control Unit (ECU) along with various sensors is an integral part of today's vehicles. To satisfy various needs like safety, comfort, environment protection, etc., the control algorithm or engine software must be highly responsive and accurate. Manual testing of these algorithms requires longer testing time and may also include manual errors. Therefore, there is a need to develop test automation tool for testing engine software in order to verify all possible positive and negative effects before testing on actual vehicle. This tool will help us to test the modifications in engine software offline in closed loop system and predict results on vehicle. Proposed test automation tool uses NI TestStand as a software platform for test sequence development. To verify developed test automation tool NI VeriStand will be used as a real-time environment and LUIS bench as a hardware platform.

Index Terms— Electronic control unit (ECU), engine software, test automation

1. INTRODUCTION

TO provide safety and comfort, engine manufacturer gives various facilities which are called as 'Engine Features'. Initially, to control various properties of engine like Fuel economy, Speed of vehicle, emission of gases, etc. mechanical operations and settings were used. Though mechanical settings are used but optimal results are not obtained. As man has discovered various sensors and as he got idea of electronic signals and its processing, Electronic Control Unit (ECU) for engines were developed. Processing on electronic signals and controlling operations of mechanical engines using electronic controller is easier and gives better results. Today, design of engine feature is combination of both hardware and software. Engine feature as a part of ECU software plays very important role in operation of vehicle (mainly in construction application and trucks). One or more features can be activated simultaneously to achieve best performance of the vehicle in its respective operating conditions.

1.1 Electronic Engine

Electronic Control Unit (ECU) is a brain of vehicle whereas engine is heart. ECU performs operation on electronic signals received from various sensors and transducers which are placed in engine environment. ECU software takes responsive actions through actuators. ECU response is used, in case of closed loop systems, which is used as a feedback and its effects on engine parameters. Fig. 1 shows concept of electronic engine.

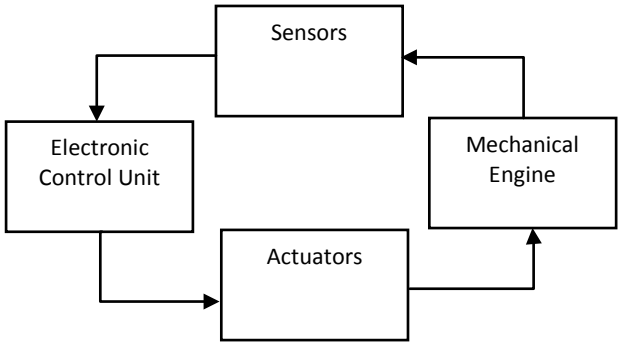

Fig.1 Concept of Electronic Engine

1.2 Engine Feature Testing

There are various engine features like Cruise Control, Power Take-Off, Gear down Protection, Trip Information, etc. Software Testing of any new or modified feature is necessary before its implementation. Different kinds of feature testing can be done on vehicle alongwith sensors, actuators and other required accessories. Sometimes feature testing may be repetitive. Such recursive testing leads to wastage of operator man hours and fuel used inc higher testing cost. Hence testing of engine feature using simulation can provide robust, reliable and low cost alternative.

1.3 Related Work

As engine features are becoming more complex, number of test patterns and risk involved in error validation are increasing. A reconfigurable NI PXIe based HIL simulation and test environment is used as a good alternative to automotive simulation environments [1]. Test Automation Support tool converts simulation patterns into model based design, which automatically adjusts the input signals, improves test quality and reduces man-hours by 50% [2]. Testing of an ECU is possible using XCP in combination with AUTOSAR [3]. A new model based diagnostic development process includes graph-based dependency model and mathematical models for online/offline diagnosis [4].

2. Proposed System

The proposed system includes testing of engine features in closed loop environment to achieve higher performance. Closed loop testing enables simulating various possible vehicle conditions to make the test software robust. Such testing provides various performance parameters like load taking capabilities, power of engine,etc. This system includes test-bench, ECU, NI TestStand, Communication protocols.

2.1 Software Platform

Test sequence is nothing but various test vectors which are used to test software logic of engine feature. NI TestStand from National Instruments is a test management software that helps to build and deploy automated test systems. NI VeriStand is a software environment for configuring real-time testing applications that helps to configure a real-time engine to execute tasks such as real-time stimulus generation, data acquisition for high-speed and conditioned measurements, and calculated channels using LabView. LUIS-Gen2 is software tool interface used to communicate with LUIS (LoadBox User Interface System) bench[5].

2.2 Hardware Platform

There are some engine feature simulation platforms called as 'Engine Simulator' which provides both hardware and software platform for engine feature testing. It is closed loop test bench used for emulating the engine environment. By using engine simulator, we can monitor certain performance parameters and also change controllable factors. (e.g. Coolant Temperature, Intake Manifold Temperature, etc.) . GarTech's LUIS bench is one such hardware platform. It is an engine simulator used to facilitate bench top engine control system hardware and software testing. LUIS system configuration consists of various modules like wave maker, analog, switch, resistive and application specific load modules [5]. Testing benches may also include FPGA based analog and digital I/O modules which emulate the sensors that are present on actual engine.

2.3 Working

Engine feature can be simulated in real-time environment using the proposed system. Calibration and configuration files are flashed on ECU as per engine feature to be tested. Fig. 2 shows block diagram of proposed system. NI LabView is interface between NI TestStand and NI VeriStand. NI TestStand send commands to engine simulator through NI VeriStand. For example – To set coolant temperature (50o C) on engine simulator, NI TestStand sends command defining the value of coolant temperature to be set. NI VeriStand receives this Set_Command from NI TestStand. Based on this Set_command, FPGA based analog I/O modules generate respective voltage levels on the channels of engine simulator. These analog channels are connected to ECU using wiring harness, which enables the ECU to read the set voltage levels. After this process, the value of coolant temperature sensed by ECU is approximately as expected (50oC). CAN is used as a communication protocol between ECU and Engine Simulator. Data Acquisition cards acquire data from Engine simulator and after signal conditioning transfers it to NI TestStand. Various test environments conditions are created on Engine simulator using test sequence. Functionality of ECU software is verified by analyzing response from ECU. Test sequence includes commands like engine simulator configuration, test report generation, test initialization and logic to test feature. Engine simulator must be configured with proper calibration and configuration files before testing any feature.

The work flow for automated testing is summarized as follows:

1. Flash calibration and configuration files on ECU (As per engine feature to be tested)
2. Configure engine simulator with appropriate (which are flashed on ECU) calibration and configuration files.
3. Execute automated test sequence.
   a. Sequence creates testing environment on bench and also observes response of ECU and bench
   b. Testing report gets generated as Excel file.
   c. Repeat steps 3a and 3b till feature gets tested completely.
4. End of testing.

2.4 Advantages and Disadvantages

The proposed testing approach saves test-cell and and vehicle testing time removes human errors leading to increased accuracy. Features which require higher testing time (more than working hours) can be tested using automated test sequences. After testing on bench, engine feature needs to be verified on vehicle. Any logical errors present in ECU software are identified and corrected during bench level testing after which error free software is available to test on vehicle.

Noise and vibrations present on running vehicle affects on various engine parameters. On engine simulator, noise and vibrations cannot be simulated. Hence, offset is observed between engine simulation and running vehicle environment. This is considered as a limitation of the proposed system.

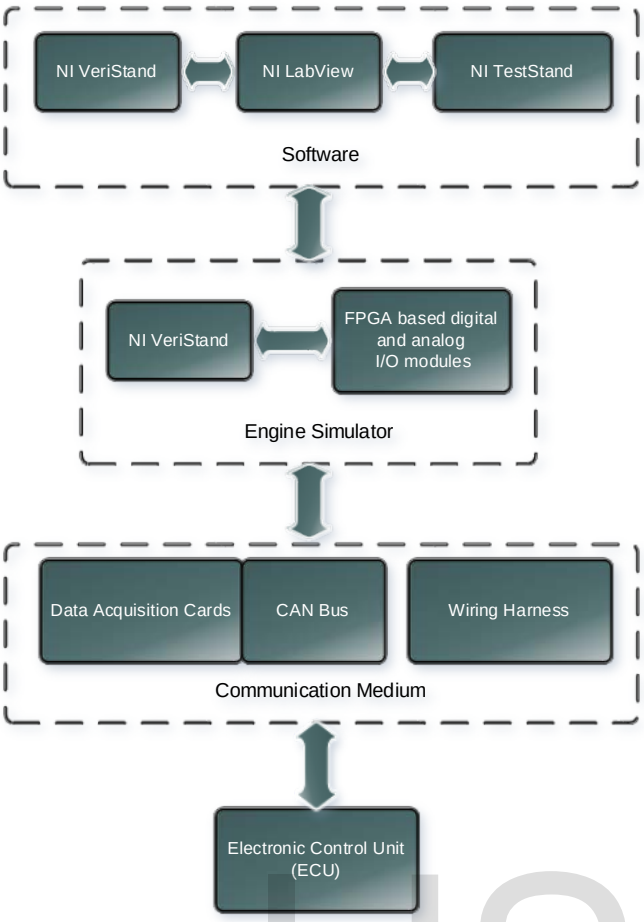

Fig.2 Block Diagram of Proposed System

3 Conclusion

With increasing number of engine features and ECU complexity, engine feature based ECU software logic testing has become important in electronic engine development process. Automated testing tool will detect errors well in advance (during development phase) providing robust ECU software logic to verify on vehicle. Test sequences can be developed for engine features to be tested across different engine platforms.

References

[1] Krisztian Enisz, Denes Fodor, Istvan Szalay, and Laszlo Kovacs, "Reconfigurable Real-Time Hardware in- the-Loop Environment for Automotive Electronic Control Unit Testing and Verification", IEEE Instrumentation and Measurement Magazine, August 2014.

[2] Philipp Caliebe, Reinhard German, ChristophLauer, "Flexible Integration Testing of Automotive ECUs by Combining AUTOSAR and XCP", International Conference on Computer Applications and Industrial Electronics, 2011.

[3] Tomomi KATAOKA, Ikuko SAKA, Ken FURUTO, Tatsuji MATSUMOTO, "Test Automation Support Tool for Automobile Software", SEI TECHNICAL REVIEW, NUMBER 77, OCTOBER 2013.

[4] Jianhui Luo, Krishna R. Pattipati, Liu Qiao, and Shunsuke Chigusa, "An Integrated Diagnostic Development Process for Automotive Engine Control Systems", IEEE Transactions on Systems, Man and Cybernetics, vol. 37, no. 6, pp. 1163-1173, Nov. 2007

[5] http://www.gartechenterprises.com/luis_gen2.html