Automation Framework of System Test Plan for Vehicle Battery Management System

Arti Patil[1], Shefali Sonavane[2], Tushar Jadhav[3]

[1,2] Walchand College of Engineering, Sangli.
[3]Tata Consultancy Services, Pune.

ABSTRACT-Battery Management System (BMS) is an electronic system that controls the charging, discharging and measures the capacity of the rechargeable battery used in electric vehicles. It involves the development of battery controller firmware. To evaluate the performance of BMS, manufacturers and developers of BMS needs an extensive testing of the controller firmware. It is basically a complex task. To avoid any serious project delay and to maintain the quality of battery controller firmware it is necessary to focus on the testing process. The lab testing of battery controller firmware is conducted on a battery simulator. In manual testing, a lot of physical time and efforts are required. It is essential to reduce the cost and time with simultaneously improving the effectiveness of testing. It can be achieved by automating the testing process. BMS requires different strategies for testing. Testing is classified in different ways based on either the processes, the methods or the testing levels (Unit, Functional and System). The integrated functioning of entire BMS is tested at the system level. The aim of this experimental work is to present the automation framework of system test plan for BMS testing on a simulator.

Keywords*- Battery Management System, Electric Vehicle, Li-ion Battery, System Test Plan, System Testing, Test Automation Framework*

I. INTRODUCTION

In electric vehicles, battery plays an important role and its supervision is to be done carefully.Battery Management System (BMS) is an electronic system that maintains the rechargeable battery (cell or battery pack). To ensure safety and prolong the life of battery packs, the BMS plays a vital role. The charging and discharging of Li-ion battery is handled by BMS. The BMS experimental platform [11] is designed, including both software programming and hardware. The aim of hardware design [12] is to simulate all the input and output signal interface to BMS. The aim of the software design is to simulate vehicle and battery power under different vehicle states and driving conditions, convertdigital signals into analog signals through hardware driver and monitor these signals in the host PC.

Testing is the process used to identify gaps, bugs or any missing requirements of the developed system. Tests can be classified into three groups: Unit, Integration (Functional) or System testing. These discussed categories correlate to test at either one of the three levels namely, atthe individual component, atthe integrated component or at the entire system level respectively. System testing is an integrated approach of hardware and software testing. The main part of BMS is controller firmware. In BMS, testing of developed battery controller firmware is done to check the correctness, completeness and quality. The focus of system testing is to verify the changes in outputs in correspondence to varying the inputs to the system. The variation in inputs is achieved by modifying the CAN message, current, temperature and voltage. The changes are proposed using the simulator. The implementation of a custom battery simulator is specifically designed for the system testing of BMS is discussed in [13].

A test plan is a document defining the test cases and procedures required to execute the test cases, resources and schedule of a software testing effort. System test plan (STP) is a plan for system level testing. In BMS, STP's are created to validate the integrated functioning of the entire battery controller system. Manual implementation of STP involves sequential execution of test cases. Moreover, it consumes a lot of time and decreases accuracy. Automated system testing eliminates errors. Test automation [14] is the use of software to execute tests and determine the equivalence between the actual outcomes and the predicted outcomes. A test automation framework is an execution environment for automated testing. It is defined as the set of concepts, assumptions and tools that create a work platform and support the automated testing. Current work lags in system-level BMS testing on a simulator along with the testing of controller firmware and the automation of testing. This paper focuses on the test automation framework for purpose of automated system-level BMS testing on a simulator. The main objective of this research is to develop test automation tools, implement the STP, generate reports and to optimize testing over time and accuracy.

The paper is organized as follows: Section I of this paper gives an overall introduction to the automation framework of system test plan. Section II presents the review ofa literature survey on BMS. Section III explains the methodology and essential steps required to develop the automation framework of system test plan. Section IV contain results and discussion while Section V concludes the research work.

II. RELATED WORK

Various approaches to the BMS testing is classified indifferent categories according to testing level-Unit, Functional and System, Methods-Static and Dynamic, Process-Functional and Non-functional and Box Method-Black box testing and White box testing. In comparison to the tests conducted on real batteries, simulator-based testing is more efficient. It has key advantages in terms of cost, testing times, flexibility, traceability,ease to reproduce and safety especially at early stages in the development process or during fault simulation [3]. The implementation of a custom battery simulator is specifically designed for system testing of BMS as discussed in [13].

The Design of BMS testing strategy must be based on a realistic approach with the aim of taking maximum advantage of the Hardware in Loop (HIL) simulation environment. The development of software and strategies for BMS testing on the simulator is explained in [1]. The huge possibilities of the strategies and ad hoc software development for BMS testing at the system level are well exhibited. BMS test platform for electric vehicle applications includes the details about monitoring and protection of the battery, communication technologies along with interfaces to enable open network communication and interoperability [2]. The BMS can monitor the parameters and protects the batteries from being overloaded by keeping the operations within the specified operating ranges. The development of the battery degradation model is based on fast charging and smart charging tests, considering two performance parameters such as energy capacity and energy losses of the batteries. The BMS able to monitor the states of the individual cells in terms of voltage, current and temperature. It considers the thermal insulation and thermal management which increases the life of the battery.

The state-of-the-art HIL simulation of a commercial BMS comprises following stages [3,4]: (1) Comprehensive review of battery simulation approaches (2) Development of an advanced multi-cell impedance-based model (3) Establishment and coupling of an aging model (4) Model verification (5) Validation of commercial BMS with HIL emulation (6) Fault insertion The work related with the following topics such as influence of battery models, validation of advanced BMS diagnostic algorithms, a thorough assessment of error and advanced fault insertions are missing.A flexibly configurable BMS prototype, which was designed by a continuum model based, verification-oriented development methodology is explained in [5]. A BMS prototype developed by the Ostfalia is used to perform the HIL simulations. The methodology of designing a framework for standardization and testing requirements of BMS for electric vehicle applicationconsists of four aspects: stakeholder, analysis, technical analysis, comparison of standards and test standards [6]. The BMS explained in [7] has higher detection accuracy and better power balance. This design uses a dissipative scheme in equilibrium. Compared with the no dissipative scheme, the energy consumption becomes an obvious disadvantage, so the equilibrium treatment scheme needs to be further studied and improved.

The design method of SS-BMS is explained in [8]. The advantage of SSBMS is that, it permits to set various charging conditions both normal and abnormal level by setting different running parameters. This system can simulate most types of charging objects which are likely to be encountered in the real time. The SS-BMS is going to be embedded in the software of electric vehicle charger testing platform for improving its testing capabilities and performance. The design of the BMS with the help of DSP, as a way for data acquisition, CAN communication, battery equalization, insulation detection, SOC estimation and data monitoring is explained in [9]. Moreover, the SOC estimation algorithm has been simulated, which proves a high prediction accuracy in SOC estimation. The designed Host can examine real-time battery data including the justification of CAN communication capability.

With static and dynamic accuracy tests, the system is proved to be accountable and stable, which can also provide a reference to other counterparts aswell.BMS is designed to achieve the full and efficient use of battery power [10].The modular design of BMS can help to achieve data collection, SOC estimation, data display, CAN communication and fault alarm. The system is based on C8051F040 microcontroller. It can effectively gather the battery parameters such as voltage, current and temperature and estimate the battery SOC. It also implements the fault diagnosis and alarming according to the battery status. It additionally aids to increase the life of the battery.

III. METHODOLOGY

A. Battery Management System

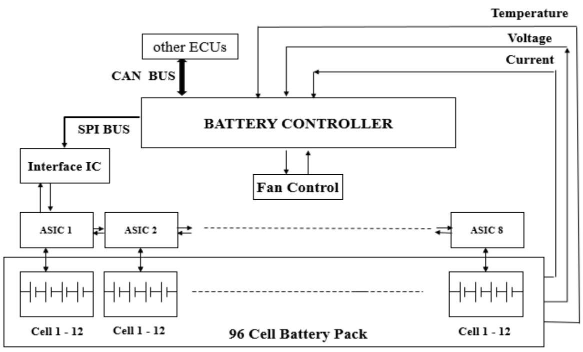

Fig. 1. Battery Management System

Battery Management ECU is used to monitor and control the charging/discharging as well as measures the capacity of the rechargeable battery. The main objective of the BMS is to protect the battery or the cells from damage. In addition, it helps to detect state of charge estimation (amount of energy left in a battery compared with the initial state of battery) and cell balancing. Battery Pack consists of various cells, which are serially connected in a pack. They are used to drive and charge various electronic devices in the vehicle. The deterioration of the battery life is derived if one of the cells is overcharged / over-discharged. Accordingly, the charge levels of the cells need to be maintained at the same level. The cell balancing is accomplished by either charging the over-discharged cell or discharging the overcharged cell. BMS involves the development of a battery controller firmware and the development of battery simulator for battery controller validation (includes: Firmware, Hardware & Application software).

Battery controller monitors various parameters such as battery voltage, current, and temperature. It helps to control the charging and discharging of the battery. The decision of charging the particular cell is based on two characteristics of the cell viz. cell life span and temperature. Lab testing is conducted on battery simulator, as the real battery pack is difficult to manage due to its physical size, weight and requirements of its allied components such as a battery charger. The test scenario generation is not feasible in some cases such as individual cell failure and individual cell overcharge\ undercharge. Repeatability in the test scenario generation is nearly impossible in the real battery pack. The advantages of the simulatorare as, it easily manageable in the LAB (in terms of physical size and interfacing requirement), full control on scenario generation and user-friendly interface.

B. STP automation framework overview

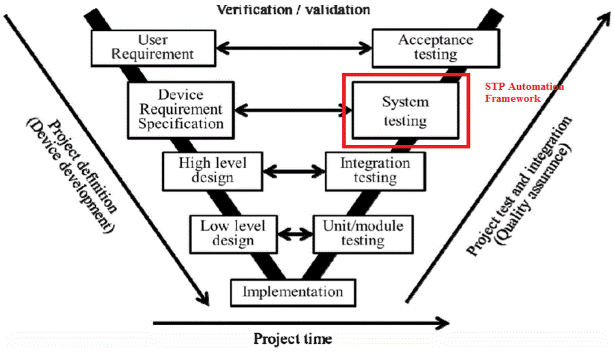

Fig. 2. V model of software development process (inspired by [1]).

Figure 2 shows a well-known V model for controller firmware development of BMS. The highlighted area depicts the position of the test automation framework that fits in the V model.

Proposed Phases of STP automation framework:

1) Automation feasibility analysis

The main goal of this phase is to study the functionality of the project, testing processes such as the current state of tasks, efforts and frequency, test data details and automation opportunities.

2) Automation Design

This step includes the design of STP automation framework.

3) Environment Setup

This stage represents a setup machine where the test cases are executed. Factor to be takeninto consideration is like, what is the configuration details of hardware and software.

4) Development of Test Script

The creation of the test script and review of the test script showcases in this phase.

5) Execution of Test Script

This stage depicts the test script checking and results verification.

6) Result Analysis

This stage includes insights on the time requirement of test case execution.

C. STP automation framework design

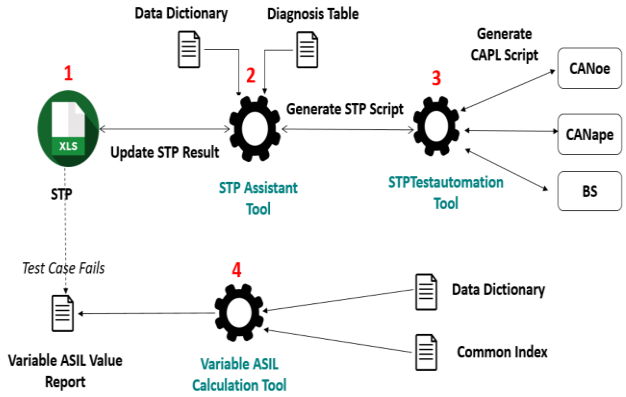

Fig. 3. STP Automation Framework Design

The design of the STP Automation Framework is depicted in figure 3. It involves the development of three tools viz. STP assistant tool, STP test automation tool and Variable ASIL calculation tool.The detailed design of these tools are as follows:

1) STP Assistant Tool:

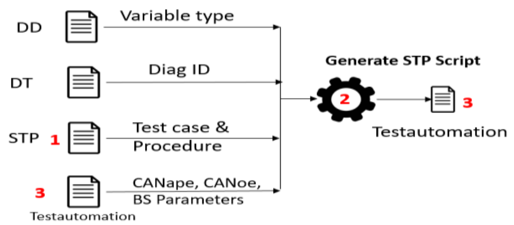

Fig. 4. Generate STP Script

As shown in figure 4 the STP assistant tool collects all the test cases and procedures to execute a test case from manually written STP. This tool collects variable type and Diag ID from the data dictionary and diagnosis table respectively. The STP assistant tool generates STP script into the STP test automation tool. After a successful generation of the script, the STP test automation tool ("STP Testautomation.xls") is ready for automated execution.

- *Data Dictionary:*

This document carries an explanation of all variable's viz. type, bit rate, initial value, min and max value.

- *Diagnosis Table:*

This document carries all flag variables and diagnosis ID of flags.

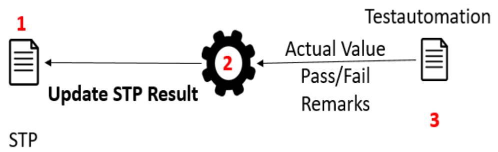

Fig. 5. Update STP Result

As explained in figure 5, the STP assistant tool takes input from executed STP i.e. actual value of test case, PASS/FAIL result and remark which is generated by the STP test automation tool and update it into manually written STP.

2) STP Test Automation Tool:

The connection between STPtest automation tool and CANoe, CANape and battery simulator is established by using the path of the files such as .exe file of simulator GUI and .cfg file of CANape and CANoe.To dumb the source code on the microcontroller board, the STP test automation tool takes .hex file path, .out file path and source code path. To execute test cases, it loads the simulator parameters, CANoe database, CANape A2L database and generates CAPL [16] (Communication Access Programming Language) script. Further, it automatically executes all test cases and generates the result.

3) Variable ASIL Calculation Tool:

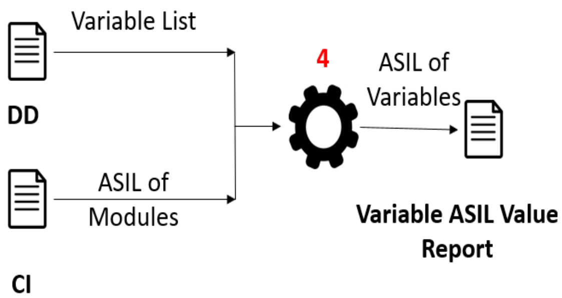

Fig. 6. Variable Asil Calculation Tool

As shown in figure 6, variable ASIL calculation tool collects a list of variables from data dictionary as well as ASIL value of modules from a common index and generates variable ASIL value report.Automotive Safety Integrity Level (ASIL) is a risk classification scheme defined by the ISO 26262 Functional Safety for Road Vehicles standard.There are five ASIL standards: ASIL B, ASIL B*1, ASIL D, ASIL QM, ASIL QM(D). The levels of ASIL are represented as follows:

ASIL QM < ASIL QM(D) < ASIL B < ASIL B*1 < ASIL

ASIL D is a most stringent level of safety measures which helps to avoid an unreasonable residual risk. In contrast, the ASIL QM needs no safety requirements.

- *Data Dictionary:*

This document carries all variable's explanation, type, bit rate, initial value, min and max value.

- *Common Index:*

The controller firmware is divided into modules. This document carries a list of all modules and ASIL value of each module.

D. Environment Setup

Figure 7 showcases the setup machine where the test cases are executed. Factor to be taken into consideration is the configuration details of hardware and software.

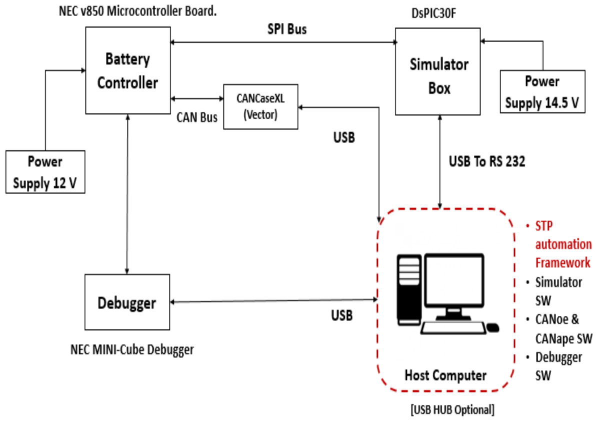

Fig. 7. Environment Setup

Battery controller firmware is placed on the NEC v850 microcontroller board. CAN Protocol system testing is done using Vector CANoe and CANape. The CANoe [15]is the comprehensive software tool used for development, testing and analysis of entire ECU networks and individual ECUs. The CANape [17] is the software tool used for optimizing parameterization of electronic control units. Simulator box is developed using DsPIC30F and VC++.

Host PC Specification:

- Operating System: Windows 7 (64bit)
- RAM: 8GB
- Processor: Intel(R) CPU @3.20 GHz
- Hard Disk: 1TB

Software List:

- Canoe
- CANape
- Microsoft Office
- MINI-Cube Debugger SW
- Simulator SW

Hardware List:

- NEC v850 Microcontroller Board
- NEC MINI-Cube Debugger
- Simulator Box
- CANCaseXL (Vector)
- USB To RS 232
- SPI Bus
- CAN Bus
- USB Cable
- USB HUB(Optional)
- 16-pin connector

The development three tools such as STP assistant tool, STP test automation tool and Variable ASIL calculation tool are done using Microsoft Excel VBA.

IV. Results and Discussion

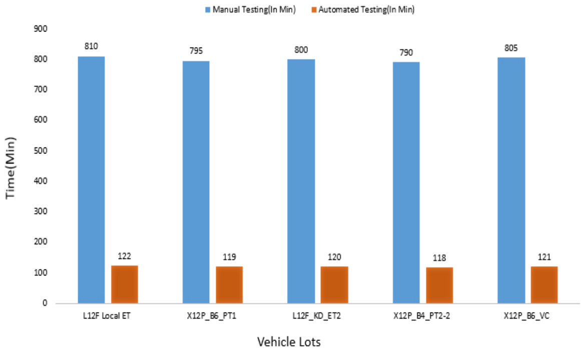

Fig. 8. Comparison of manual and automated testing time.

The experimental analysis is carried out using the data from 5 different vehicle lots (NISSAN project-TCS, PUNE) to analyze how much time it takes to complete the test case. STP of each vehicle lot contains 100 test cases. The time required for manual execution of STP as well as the time required for automation framework is shown in figure 8. Time reduction is achieved with STP test automation framework for system testing. Average time saving per vehicle lot when system testing is done using STP test automation framework is approximately equal to 85%.

V. Conclusion

Basic principles of the BMS are presented in this paper. A practical way to develop the test automation framework for BMS at the system level is well explained by use of a simulator. The methodology of framework contains the development of three tools viz. STP assistant tool, STP test automation tool and Variable ASIL calculation tool. The implementation of the framework is done in 6 distinct phases such as Automation feasibility analysis, Automation design, Environment setup, Development of the test scripts, Execution of the test scripts and Result analysis. The experimental analysis is carried out using the data from 5 different vehicle lots (NISSAN project-TCS, PUNE). The proposed methodology helps in easy implementation of STP and generating reports. The results drawn after testing are automatically stored in the sheet. Manual efforts of implementing the STP are replaced by the test automation framework explained in this work. Human errors are almost avoided using automated testing of STP. The use of automation framework optimizes the system testing of BMS over time and accuracy parameters. The average saving of time required to implement STP is approximately equal to 85%.

Acknowledgment

The research reported in this paper is supported by Tata Consultancy Services, Pune. We would like to gratefully acknowledge Tata Consultancy Services, Pune and all its engineers for letting us use the facilities and the research labs for the tests and support for executing this research project.

References

[1] Fleischer,C., Sauer, D.U., Barreras, J.V., Schaltz, E., and Christensen, A. E.2016. Development of software and strategies for Battery Management System testing on HIL simulator.In Proceedings of the Eleventh International Conference on Ecological Vehicles and Renewable Energies (EVER). Monte Carlo.pp. 1-12.

[2] Mihet-Popa,L. and Groza,V.2015.Battery management system test platform developed for electric vehicle applications. In Proceedings ofIEEE 9th International Symposium on Intelligent Signal Processing (WISP).Siena.pp. 1-6.

[3] Barreras,J.V.,et al.2016.An Advanced HIL Simulation Battery Model for Battery Management System Testing," in IEEE Transactions on Industry Applications, 2016,vol. 52, no. 6, pp. 5086-5099.

[4] Barreras,J.V., et al.2015.Functional analysis of Battery Management Systems using multi-cell HIL simulator.In Proceedings of Tenth International Conference on Ecological Vehicles and Renewable Energies (EVER).Monte Carlo.pp. 1-10.

[5] Liu-Henke,X.,Scherler,S.,and Jacobitz,S.2017.Verification oriented development of a scalable battery management system for lithium-ion batteries.In Proceedings ofTwelfth International Conference on Ecological Vehicles and Renewable Energies (EVER).Monte Carlo.pp. 1-7.

[6] Rahmawatie,B., et al.2017. Designing framework for standardization and testing requirements of battery management system for electric vehicle application.In Proceedings of 4th International Conference on Electric Vehicular Technology (ICEVT).Bali.pp. 7-12.

[7] Zhang, A., Song,S.,Wang,C.,Zhang,J.,Wang,K., and Li,L.2017.Research of battery management system for integrated power supply.In Proceedings of Chinese Automation Congress (CAC). Jinan. pp. 3178-3181.

[8] Yan,X., Li,W., Gu,J., and Xiao,X.2012.A Simulated System of Battery-Management-System to test Electric Vehicles Charger.In Proceedings of IEEE International Electric Vehicle Conference.Greenville.pp. 1-5.

[9] Liu,B., Liu,M., Jiang,X.,Tuo,X., Zhou,H., and Ren,J.Design of battery management system based on DSP for BEV.2017.In Proceedings of 9th International Conference on Modelling, Identification and Control (ICMIC).Kunming.pp. 857-862.

[10]Wang,Y., and Liu,Y.,2011.Electronic control system design and test of pure electric vehicle battery management system.In Proceeings of Second International Conference on Mechanic Automation and Control Engineering.Hohhot.1289-1292.

[11]F. Zhu, G. Liu, C. Tao, K. Wang and K. Jiang, "Battery management system for Li-ion battery," The Journal of Engineering, vol. 2017, no. 13, pp. 1437-1440, 2017.

[12]Li,Y.,Sun,Z., and Wang,J.2009.Design for battery management system hardware-in-loop test platform.In Proceedings of 9th International Conference on Electronic Measurement & Instruments.Beijing.pp. 3-399-3-402.

[13] Rienzo,R.,Roncella,R.,Morello,R.,Baronti,F., and Saletti,R.2018.Low-cost modular battery emulator for battery management system testing.In Proceedings of IEEE International Conference on Industrial Electronics for Sustainable Energy Systems (IESES).Hamilton.pp. 44-49.

[14]Fewster,M., and Graham,D.1994.Software Test Automation.Addison-Wesley Publication.New York.pp. 386-398.

[15]Vector.com.2018. CANoe product information.[online]Available at:https://vector.com/portal/medien/cmc/info/CANoe_ProductInformation_EN.pdf [Accessed 8 Jul. 2018].

[16]Can-newsletter.org.2018. Programming with CAPL.[online]Available at: https://can-newsletter.org/assets/files/media/raw/a456e3078f907a0482182ce831912427.pdf [Accessed 8 Jul. 2018].

[17]Vector.com.2018. CANape Product Information.[online]Available at: https://vector.com/portal/medien/cmc/info/CANape_ProductInformation_EN.pdf [Accessed 8 Jul. 2018].

Authors Profile

Arti Patilpursued the Bachelor of Engineering degree in Computer Science and Engineering fromCollege of Engineering, Miraj, India (affiliated to Shivaji University, Kolhapur, MH India) in the year 2015. She is currently pursuing the Master of Technology in Information Technology at Walchand College of Engineering, Sangli, India (Govt. Aided_Autonomous Institute) since 2016. Her main research work focuses on Automotive System.

Shefali Sonavane pursued Bachelor of Engineering, Master of Engineering and Ph.D degree in Computer Science and Engineering from Walchand College of Engineering, Sangli (Govt. Aided_Autonomous Institute) affiliated to Shivaji University, Kolhapur, MH India, in the year 1992,2001 and 2010, respectively. She is currently working as In-charge Head in the Department of Information Technology, WCE,Sangli, India. Her research interests include Information Security, Computer Vision, Machin Learning and Big Data.

Tushar Jadhav pursued Bachelor of Engineering in Computer Science and Engineering from Pune University. He is currently working as Manager, NISSAN LBC project, TCS, Pune.