Automation of Test Efficiency Improvement in Diagnostic Release Test for Electronic Brake System

Sabiha Khanam[1], Dr. H.R. Shashidhar[2]

[1,2]*Dept of CSE, RNSIT*

Abstract--Release test is an essential part of Software development life cycle in the field of Automotive embedded systems. The 'V' cycle is followed at the moment for software development and test. Considering the idle case of the lifetime of the proposed system for Release testing, the entire testing process takes the duration of 3 weeks. The release Test process can be categorized in to 3 distinct phases which are 'Pre-execution', 'Execution' and 'Post-execution' phases. The aim of this proposed system is to develop an tool interface to reduce the Manual effort in the 'Pre-execution' and 'Execution' test phases so that overall time line of the release testing process to a week. With the manual testing method the Tester has to analyze the test requirements and then manually generate the test cases which adds to the proposed system timeline. The aim of the proposed system is to improve the efficiency of Diagnostics Release test by creating a tool to recognize the keywords &patterns from the input customer requirement document and create all the possible positive and negative test cases for all the requirements to provide the maximum possible requirement coverage. Another added advantage of the tool is that it also creates the dependency files required to create a Test environment required for testing. Basically Automating the Test scripts using the excel sheet, text document ,PDF file and Converting it to the text file into the Required Format to reduce Employees time and the manual work of the employees.

Keywords: Automation, V Model , Execution, Test Cases

I. INTRODUCTION

In the basic Waterfall model process seen some disadvantages or limitations in the model which leads to a new SDLC model. As seen in the Waterfall model the issues found in the end of the SDLC, this is due to the testing is occurred in the end phases of the SDLC. To overcome this problem the V-Model is comes into the picture. It is always better to introduce testing in the early phase of SDLC, as in this model the testing activity gets started from the early phase of the SDLC. Before starting the actual testing, testing team has to work on various activities like preparation of Test Strategy, Test Planning, Creation of Test cases & Test Scripts which is work parallel with the development activity which help to get the test deliverable on time. Based on requirement specification, the development& testing activity is started and document developer team started working on the design & after completion on design start actual implementation and testing team starts working on test planning, test case writing, test scripting. Both activities are working parallel to each other. In Waterfall model & V-model they are quite similar to each other. As it is most popular Software Testing Life Cycle model so most of the organization is following this model The V-model is also called as Verification and Validation model. The testing activity is performed in each phase of Software Testing Life Cycle phase. In the first half In this process "Do-Procedure" would be followed by the developer team and the "Check-Procedure" would be followed by the testing team to meets the mentioned requirements. In the V-Model software development life cycle different steps are followed however taking a most common type of V-model example. The V-model typically consist of the following phases:

1.Unit Testing: Preparation of Unit Test Cases

2.Integration Testing: Preparation of Integration Test Cases

3.System Testing: Preparation of System test cases

4. Acceptance Testing: Preparation of Acceptance Test Cases

Let's see the typical process involved in delivering software from 'development phase' to the 'testing phase' for a successful bug-free software release to production/client. These processes are either overlooked or skipped by software companies, which results in poor test management and thereby' a "buggy" software releases to the client, which leads to "unsatisfied customers". Even though lot of time and great effort being given by the testing team for each and every release, when the released software is not having the quality as defined or bench-marked or not meeting the expected criteria, it will not only affect the company reputation with the customers but also de motivates and demoralizes the project team most importantly the testing team as a whole. The part of a testing team in this scenario, "how to improve my testing capabilities and is there any better way of overcoming this situation". I want to give some tips and suggestions, based on my experience with various testing teams involved in software applications and enterprise products releases with multiple domains and platforms and with multiple testing frameworks, on how to improve the test release processes,

which will simplify professional life as a test engineer or a test manager for delivering world-class software. The Figure 1 gives an overview of V Model - Software Development Life Cycle of a test release process.

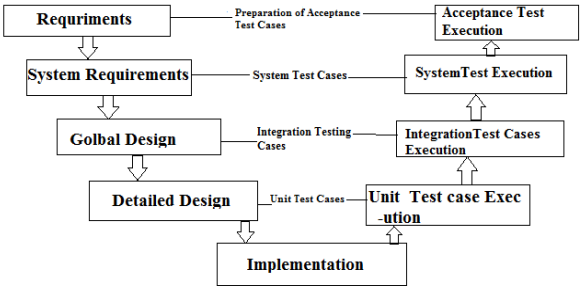


Figure 1. V Model - Software Development Life Cycle

II. PREVIOUS WORK

Electronic Hydraulic brake system has been studied actively. Many significant results are achieved by the automotive researcher. EHB system name "Electronically Controlled Brake System" (ECB). The ECB they presented integrates brake control functions such as the cooperative control between the regenerative brakes of the front/rear motors and the hydraulic brakes, ABS (Antilock Brake System), VSC (Vehicle Stability Control), TRC (Traction Control), etc. The ECB is definitely an EHB system that controls the wheel cylinder hydraulic pressure electronically. The pressure of each wheel cylinder is controlled by linear solenoid valves. So the hydraulic pressure of wheel cylinders is controlled individually and smoothly. David F. Reuter[8] presents a multiple EHB hydraulic design configurations. The configuration adopted in Delphi's EHB development has included use of four-wheel failsafe with individual isolation pistons and utilization of mechanical pedal feel lockout. This particular design allows system flexibility, inherent accumulator pre charge isolation, and the ability to tune for optimum failed system stopping performance for all vehicle classes. Luigi Petruccelli [9] etc. carried out a realistic EHB hydraulic model using the AMESim. They build an EHB test bed and conceived a control logic without ABS intervention devoted to medium-low intensity braking maneuvers to determine optimal braking force distribution and pedal feeling depending on the maneuver. They also implemented the vehicle model in Matlab/Simulink. Developed a Hardware-In-the-Loop Simulation system for EHB that includes a high pressure generator and an independent brake pressure control system[5]. The HlLS system consists of three parts; the hardware, software, and interface parts. The EHB control logic has been developed in their work and was implemented on a PC. The performance of the EHB controller has been tested under various driving conditions in the HILS system, and the results are compared with those from the conventional VDC logic. Even though a lot of researchers have resolved so many key technologies in EHB designing and control, there still exit many technologies need to be resolved in practical applications, such as reliability, system performance.

III. METHODOLOGY

This proposed system explains to extract the required pattern from the given file which may be excel, pdf or any other file using Regular Expression and some of the logics from the required pattern. Here the technology used is visual basics in visual studio. Sometimes if it is excel Macro in VBA for creating the whole template.

Algorithm:

Step 1 : Read the given Input file.

File Reader=My. Computer. File System .Read All Text("C:\test.txt")

Step 2 : Storing it in the Variable or an Array

Dim Variale_name as type of Variable

Step3:Writing the Logic According to the Required Format

My. Computer. File System. Write All Text("C:\test.txt","New text to be added",True)

Step 4:Create a new text file write into the txt file into the required format

Regexrx=newRegex(@"\b(?<word>\w+)\s+(\k<word>)\b" Regex Options .Compiled| Regex Options. Ignore Case);

Match Collection matches = rx. Matches(text)

Step 5 : That text file is used for the further usage of test Engineer

Dim fs As File Stream = File. Create(path)

IV. EXPIREMENTAL RESULTS

There some of the inputs which are shown in the Fig 1, Fig 2, and the outputs obtained from the given input are shown in the Fig 3 and Fig 4.

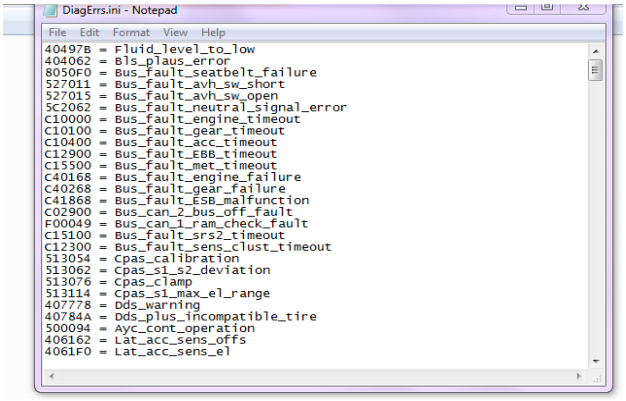
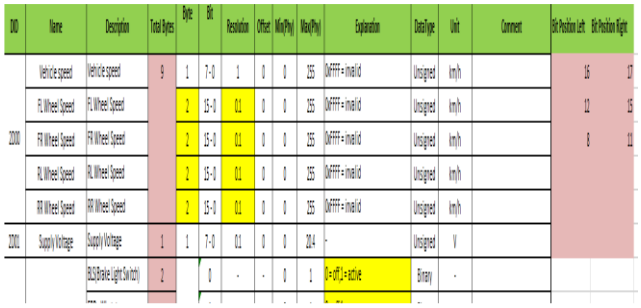


Figure 2. The image of the txt file as an input.



ID	Name	Description	Total Bytes	Byte	Bit	Resolution	Offset	Min(Phy)	Max(Phy)	Explanation	Data Type	Unit	Comment	Bit Position Left	Bit Position Right
	Vehicle speed	Vehicle speed	9	1	7-0	1	0	0	255	0xFFFF = invalid	Unsigned	km/h		16	17
	FL Wheel Speed	FL Wheel Speed		2	15-0	0.1	0	0	255	0xFFFF = invalid	Unsigned	km/h		12	15
2000	FR Wheel Speed	FR Wheel Speed		2	15-0	0.1	0	0	255	0xFFFF = invalid	Unsigned	km/h		8	11
	RL Wheel Speed	RL Wheel Speed		2	15-0	0.1	0	0	255	0xFFFF = invalid	Unsigned	km/h			
	RR Wheel Speed	RR Wheel Speed		2	15-0	0.1	0	0	255	0xFFFF = invalid	Unsigned	km/h			
2001	Supply Voltage	Supply Voltage	1	1	7-0	0.1	0	0	20.4	-	Unsigned	V			
		BCS.Brake Light Switch	2		0	-	-	0	1	0=off,1=active	Binary	-			

Figure 3. The Excel file given as an input

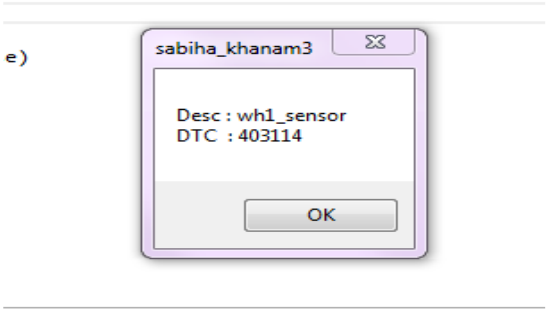


Figure 4. The Result Obtained from the given text file

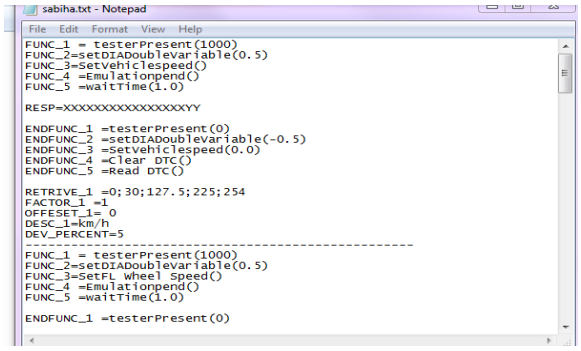


Figure 5. The .txt file generated from Data extract from the excel file

V. CONCULUSION AND FUTURE WORK

This proposed system provided a chance to learn the different tasks to attain the automation the test scripts. Working in a different domain provided a chance to learn different technology. The theoretical knowledge was put into practice and exposure to the industry has been an added advantage and could explore and learn tutorials and go through how it can be used. Writing a code for extracting required information from the input file, comparing it with the data present in the excel sheet and writing it into the text file. This proposed system provided an opportunity to improve the ability to solve problems efficiently and in a timely manner without difficult. It is implemented using different tools. Creating optimization tool which is needed for efficient development of test scripts directly from the requirements as input. Ultimately problem-solving ability has been improved since the main aim of this proposed system is to create better environment for testers and to reduce the manual work and to reduce the time consumption which was achieved through this project

Future Work

Automating the test scripts of the model and the traceability in the given model which is electronic brake system will be made easy and efficient and improving the test efficiency.

ACKNOWLDGEMENT

We would like to thankyou to our guide Dr. H R Shashidhar(Dept of CSE, RNSIT, Bangalore) for his valuable suggestion and comments that helped improving this work. This support is greatly appreciated.

REFRENCES

[1] IVI-4.3: IviFgen Class Specification. April 14, 2008 Edition. Revision3.0. http://www.ivifoundation.org.

[2] David Kaushansky. IEEE AUTOTESTCON 2008. Design Approachesfor Developing a High Density, High Performance Analog Test Instrument.

[3] Eric Freeman, Elisabeth Freeman, Kathy Sierra, Bert Bates. (2004).Head First Design Patterns. United States of America.

[4] IVI-4.2: IviDmm Class Specification, June 9, 2010 Edition, Revision4.0. http://www.ivifoundation.org

[5] M. Dewey & D. Carey, “Modernizing Legacy Automated Test Systems for DoD Depots”, Autotestcon 2010

[6] M. Rafik, F. Dieudonné and G. Morin, “Low Cost Wafer Level Parallel Test Strategy for Reliability Assessments in sub-32nm technology nodes,” in Proc. IEEE International Conference on Microelectronic Test Structures, April 2011.

[7] T. Adachi, A. Pramanick and M. Elston, “Parallel, multi-DUT testing in an open architecture test system,” in Proc. IEEE International Test Conference, November 2005.

[8] J. Rivoir, “Lowering cost of test: parallel test or low-cost ATE?”, in Proc. IEEE Asian Test Symposium, November 2003.

[9] J. Rivoir, “Parallel test reduces cost of test more effectively than just ac heap tester”, in Proc. I Electronics