BITS Pilani
Pilani Campus

Firmware verification for Automotive Wireless Battery Monitoring Systems

Author: Sai Kartik (2020A3PS0435P)
Manager: Mr. Abhinandan Subbarao
Professor: Dr. Sathisha Shet K
PS Station: Analog Devices, India

Outline

Aim and Problem statement

innovate achieve lead

Aim

This project aims to verify the firmware of battery monitor sensors for a car's wireless battery monitoring system (wBMS).

Problem statement

Use automation concepts to test the software and deliver it to customers quickly and efficiently without bugs.

Main Objectives

- To identify the testcases to be executed on wBMS system
- To write scripts to perform manual testing of all tests
- To automate the running of the test suite and generation of test report

Design Methodologies

- To perform manual testing, we flash the firmware onto the respective devices with JLink lite debuggers
- Download files to the gateway and the nodes to faciliate communication between them
- Configure the front-end application (GUI) to control the network and test the functionality.
- Ensure the setup is RF shielded fairly well

High level implementation

The software test cycle (STLC) mainly consists of 4 major steps to go through:

- Test planning

High level implementation

The software test cycle (STLC) mainly consists of 4 major steps to go through:

- Test planning
- Test case development

High level implementation

The software test cycle (STLC) mainly consists of 4 major steps to go through:

- Test planning
- Test case development
- Test environment setup

High level implementation

The software test cycle (STLC) mainly consists of 4 major steps to go through:

- Test planning
- Test case development
- Test environment setup
- Test execution

High level implementation

The software test cycle (STLC) mainly consists of 4 major steps to go through:

- Test planning
- Test case development
- Test environment setup
- Test execution

Test Planning

- This step is the most significant part of software testing where the required testing strategies are created.

Test Planning

- This step is the most significant part of software testing where the required testing strategies are created.
- It is typically the team lead's/manager's role to establish the project cost and efforts required.

Test Planning

- This step is the most significant part of software testing where the required testing strategies are created.
- It is typically the team lead's/manager's role to establish the project cost and efforts required.
- This phase begins once the requirement collection phase has been completed.
- The major outcome of this phase is the finalised test plan/strategy which has to be adhered to.

Test Planning

- This step is the most significant part of software testing where the required testing strategies are created.
- It is typically the team lead's/manager's role to establish the project cost and efforts required.
- This phase begins once the requirement collection phase has been completed.
- The major outcome of this phase is the finalised test plan/strategy which has to be adhered to.

Test Case Development

- Once a strategy of the tests to be performed is outlined, the required data for it is gathered.

Test Case Development

- Once a strategy of the tests to be performed is outlined, the required data for it is gathered.
- This data is organised to fit various test cases to ensure coverage of all possible scenarios.

Test Case Development

- Once a strategy of the tests to be performed is outlined, the required data for it is gathered.
- This data is organised to fit various test cases to ensure coverage of all possible scenarios.
- Once the design of individual test cases is complete, each test case is linked in a chain in the Responsibility Traceability Matrix.

Test Case Development

- Once a strategy of the tests to be performed is outlined, the required data for it is gathered.
- This data is organised to fit various test cases to ensure coverage of all possible scenarios.
- Once the design of individual test cases is complete, each test case is linked in a chain in the Responsibility Traceability Matrix.

Conclusion

References[1]

[1]Other documents regarding specific hardware/software architecture are for internal use only and cannot be shared as open sources/references