Design And Development of Automated Generic Test Environment For Bluetooth Based Devices

Mr.Anand Kurlekar[1*] , Mr.Vikas Nimbalkar[2] , Mr.Rajan Mevekari[3]
Department of Electronic Engineering
[1]M.tech student, Walchand College of Engg. Sangli , Maharashtra, India
[2]Manager , Hella India Automotive pvt Ltd
[3]Assistant Professor, Walchand College of Engg. Sangli , Maharashtra, India

Abstract-*This paper presents design and development of Automated Generic Test-Environment for Bluetooth Low Energy (BLE) based devices. Raspberry pi Embedded Linux development board is used as a hardware element, which is used as a BLE transceiver to communicate with Device under Test (DUT). Raspberry pi reads or writes data to DUT, depending on request from GUI. GUI is used to control test environment operation and to display response data from DUT in more readable manner. Further the test environment is capable of testing automatically various working scenarios for DUT just by using test-sequence file, which is Microsoft Excel file written in specific format. All test-cases from test sequence file are tested one by one automatically also the response is compared with expected response and test case result is prepared pass or fail. At last very important thing about this test environment is, it is a generic test environment which means whenever DUT features changes test environment will get modified without re-coding.*

Keywords-Bluetooth Low Energy (BLE), Device under test (DUT), Automation, Generic, GUI, Python..

I. INTRODUCTION

Bluetooth Technology has great impact on todays embedded systems. Because of its advantages it comes to the first priority used to make various devices wireless. It is also growing the use in automotive industry. Every product in the industry has to go through product development process. This process ensures the proper development of the product.

After development every product has to pass through the testing process. As Bluetooth is a wireless technology, Bluetooth based applications cannot be handled by conventional wired testing environment. Hence for testing such Bluetooth based product we require a device which can sense the Bluetooth messages, which can send requests and receive the response from Bluetooth peripheral devices. Further this test environment should be generic to avoid the rework of the development of new test environment every time and it should be automated to reduce human efforts and high accuracy.

The Automated Generic Test environment For Bluetooth based devices is the system which is used for testing the functionality of DUT (device under test). In this paper we will consider DUT as a Bluetooth based ECU (Electronic control Units). There are various ECUs which are using Bluetooth technology for wireless communication. To ensure that these devices are working properly proposed environment is used. Automated is the term which makes the environment smart to reduce the man power and increase the accuracy. Generic is the term which makes it standard environment for all Bluetooth based product testing. Developing such test environment consist of development of the test setup and the GUI for the Automation. For test setup the main requirement is a device which can sense the Bluetooth messages and can act as a BLE transceiver.

The best option of such device is using a development board which consists of Bluetooth low energy support. Raspberry pi is an embedded Linux based development board. Latest raspberry pi model 3B supports Bluetooth classic and Bluetooth 4.0 (BLE). Also, it allows us to write a script in python language. Further for the automation development of GUI is very important work. GUI will be a python script which will make connection with the DUT and automatically request for the required data and will capture the response and show on the GUI. Test cases are written for testing of the DUT in various conditions and for various supply voltages. Test cases are written by considering all possible ways the device can be used. Further the Test Environment is developed to such a higher level that all these Test cases are automatically executed one after the other and the whole process and the results are recorded.

II. PROPOSED SYSTEM

Proposed system consists of embedded Linux based development board (Raspberry pi model-3B) as a hardware element. PC is used to run GUI. Detail block diagram of the system is as shown in following figure.

GUI consist of various buttons, checkboxes, entry fields etc. depending upon the need user will press button in

GUI because of this action particular request will be sent to Raspberry pi using Ethernet network.

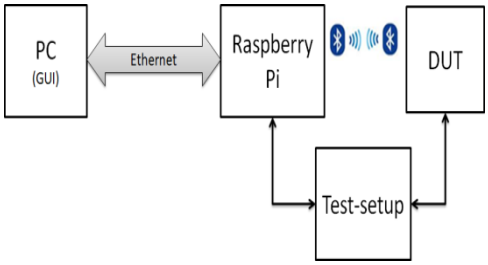

Figure 1- System Architecture

As soon as raspberry pi receives request it will do the required task as per request. Request may be to read any particular feature of a DUT over Bluetooth or to write some data on DUT using Bluetooth or to read particular hardware pin of DUT. After action made by raspberry pi there will be response from DUT over Bluetooth. The responsibility of Raspberry pi is to send this response to GUI. GUI will always be ready to receive data from DUT. This response is recorded and after completion of all requests GUI will produce the result by comparing the response with expected response and result about pass and fail will be recorded in result sheet.

PC and Raspberry pi are connected to an Ethernet network both will get an unique IP-address using that IP-address both can communicate using TCP/UDP protocol.

Raspberry pi has in built Bluetooth v4.1 supporting hardware. It supports for both classic Bluetooth as well as BLE. Raspberry pi is embedded Linux based development board hence it has support of Linux operating system. Along with Linux features Bluetooth communication is easy task. Using Linux commands raspberry pi can be configured to work as a BLE transceiver.

Test setup consist of various loads (relay, LED, Bulb) required to show output of DUT. Raspberry pi can read DUT output on GPIO pins using test setup.

III. CONFIGURE RASPBERRY PI FOR BLE

Though raspberry pi has built in Bluetooth hardware it do not have required software and updated Bluez package which is required for Bluetooth communication. Using Linux commands one can easily install the bluez package.

After installation of bluez package raspberry pi is ready to use as a BLE transceiver. There are tools provided by Linux to handle Bluetooth. These tools are hcitool and gatttool.

IV. GENERIC TEST-ENVIRONMENT

Generic means the test-environment is not specific for any particular Bluetooth device. Means if device is modified in its features or some new improved feature device developed same test environment can be used.

Test-environment is able to update itself as per feature of the DUT. Depending on the DUT supported feature GUI will get modified for each DUT. This is done using one feature-sheet which consists of feature information about the DUT. This sheet will be read by the test-environment and depending on data new GUI will be prepared.

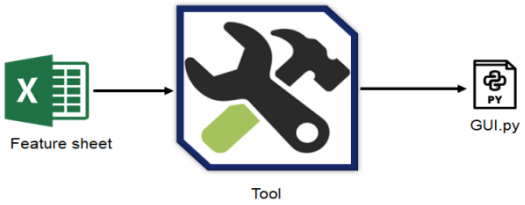

Figure 2- Generic Test Environment concept

Feature sheet is an excel sheet in which various features of the DUT are listed. Tool will read the feature and depending on the data inside sheet one python file is generated which is nothing but the source code for GUI. Tool is also a python code which will generate a GUI code by reading data from feature sheet.

V. AUTOMATED TEST ENVIRONMENT

Testing of any device starts with writing various test scenarios these test scenarios are nothing but different conditions on which system should be tested. These test scenarios are also called as test cases. List of these test cases is called test sequence.

Testing of any device is started with writing test sequence file. Test sequence file consist of various action and inputs to be applied on DUT for various preconditions and response from DUT is also recorded in same document which can be compared with expected response whenever required. In manual testing tester have to follow this file and manually perform the various actions. Tester has to record the response from DUT manually and compared with expected response manually and at last result in terms of pass and fail is written manually. So in this tedious process because of manual process results are not always sure also the manual process takes very much time. Hence to reduce time required for testing and to increase accuracy it is important that testing should be automated.

Automation is carried out using python script. Tester has to browse the test sequence file and press one button. In that single button click whole test sequence will get executed and results are generated. GUI code itself is enhanced for the automation.

VI. SIMULATION & EXPERIMENTAL RESULT

Figure 3: GUI

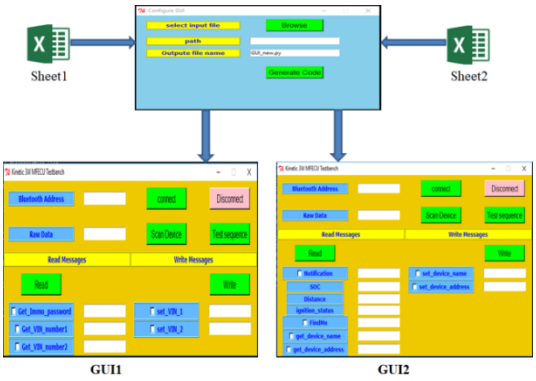

Figure 4: Generic Test Environment Demo

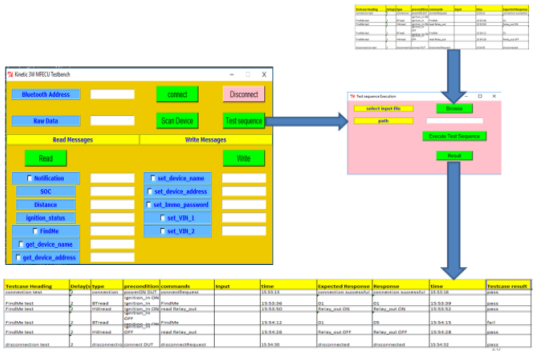

Figure 5: Testing Automation Demo

VI. CONCLUSION

Manual testing of one test takes 2 to 3 minutes. Time required for 1 test case after automation is 2 to 3 seconds. Each time test sequence is executed automatically testing report is created. Report tells the summary about test case passed or failed.

According to comparison between manual and automated test environment time required for testing is reduced to very large extent. Because of Automation it is possible to find faults due by considering all scenarios, hence testing accuracy is increased. System cost is very less which saved lot of money.

REFERENCES

[1] Matt Lewis (March 2013) "Testing Bluetooth Low Energy Devices - Punch Through"
[2] Jared wolff (Apr 2014)"Getting started with Bluetooth Low Energy on Linux"
[3] https://www.bluetooth.com/.
[4] https://en.wikipedia.org/wiki/Bluetooth_Low_Energy
[5] http://www.instructables.com/id/Control-Bluetooth-LE-Devices-From-Raspberry-Pi/
[6] https://www.elinux.org/RPi_Bluetooth_LE
[7] https://www.python.org/
[8] https://en.wikipedia.org/wiki/Python_(programming_language)
[9] https://www.tutorialspoint.com/python/