International Journal on Recent and Innovation Trends in Computing and Communication
Volume: 5 Issue: 5

ISSN: 2321-8169
796 – 798

Automated Testing of Infotainment System

Akshay Pola
Master of Technology in Industrial Electronics,
Sri Jayachamarajendra College of Engineering
Mysuru, India

B A Sujatha Kumari
Associate Professor, Department of E&CE
Sri Jayachamarajendra College of Engineering
Mysuru, India

Abstract— Infotainment system is one of the most complex Electronic Control Unit (ECU) which is integrated with number of ECU's in the car. Infotainment system is getting more features in recent years because of which the testing of these systems has become very complex. Testing of these systems with very high accuracy is a difficult task. This paper presents a method of testing the various features of Infotainment System with very high Accuracy and less Human Effort.

Keywords- *ECU, Bluetooth, Wi-Fi, ADB, HMI*

*****

I. INTRODUCTION

The suppliers of Infotainment systems are facing a challenge to develop an Infotainment system that satisfies their customer requirements. This is mainly because of integration of different technologies in the same ECU (Electronic Control Unit). As the result of this complex system the testing of various requirements becomes a very difficult task.

Infotainment system is built on an Operating System which supports number of technology such as Bluetooth, FM, AM, Wi-Fi (Wireless Fidelity), CarPlay, CarLife, MirrorLink. Because of these different technologies integrated into the single ECU the task of testing the system with very high accuracy becomes very essential.

Figure 1: Infotainment system with CarPlay

The functional features of this ECU is usually tested manually, this process involves creation of test cases which describe the steps to test the various requirements of the ECU. The problem with this method is huge amount of time is required to test the features and the accuracy needed is not fulfilled to the specified requirement. Some other problems include, the test engineer of the ECU should be familiar with the language used in the infotainment system. For the systems that are in different foreign languages it becomes very difficult task for the test engineer to test the ECU.

One of the other methods is to integrate the camera with the robotic arm. The problem with this method is it is very expensive and difficult to generalize.

Software method of automating the testing process is mostly economical and efficient method of testing the Infotainment System. This method of automating the test actions also helps in automating the actions that are carried out in Android phones. Majority of the test cases involve connecting the Mobile Phones to the Infotainment system, hence choosing software method of automating the test actions is preferred. As it becomes very easy to integrate the automated test actions which are done for android mobile phones and infotainment system. Automating the test actions in Android mobile phone side requires ADB (Android Debug Bridge) which is used to communicate with the computer system.

The majority of the requirements given by the customer will be related to Bluetooth technology. The requirements provided by customer also cover different technologies such as MirrorLink, CarPlay and CarLife. This paper concentrates on automation of Bluetooth and connectivity test actions.

II. BLUETOOTH IN INFOTAINMENT SYSTEM

Bluetooth technology helps the customer to make a Hands-Free call while driving, play the audio files present in the mobile phone, which is connected to the infotainment system using Bluetooth. Download phone book from the mobile to the infotainment system and many more. All these features of the Bluetooth are realized using profiles such as HFP (Hands Free Profile), A2DP (Advanced Audio distribution Profile), AVRCP (Audio Video Remote Control Profile), PBAP (Phone Book Access Profile).

The services provided by Hands Free Profile are, this profile enables mobile phone to send voice data to the infotainment system connected in the vehicle, thus providing a Hands-Free call feature, responsible for answering or rejecting the phone calls, indicates voice call connection and disconnection, gives a notification of call waiting, responsible for switching between active calls and calls on hold, retrieves phone book records that was downloaded with the help of Phone Book Access Profile (PBAP), retrieves the paired phone call list, notifies an application during incoming call.

Audio Video Remote Control Profile realizes the options such as play, pause, fast forward and backward, next, previous. These options are used while playing media files in the infotainment system.

Phone Book Access Profile is used to download the phone book or the contacts from the phone which is connected to the infotainment system. This also involves downloading the profile pictures of the contacts that are saved on the mobile phone.

Automation of testing these profiles is possible using the serial communication port of the infotainment system. The scripting language used to achieve the automation is TCL (Tool Command Language), which is an open source scripting language.

One of the stress case that can be automated is click on the Play/Pause button every 500ms , of the media that is been played in the infotainment system using A2DP profile of the Bluetooth technology. This can be automated by periodically sending the touch co-ordinates of Play/Pause to the infotainment system.

III. ANDROID DEBUG BRIDGE

Android Debug Bridge (ADB) tool is used to realize automatic execution of the test actions on the Android mobile phone, thus realizing the automation on the Android Phones.

ADB works in simple client/server architecture, and is made up of three key components, namely: Client, Server and Daemon as shown in Figure 2

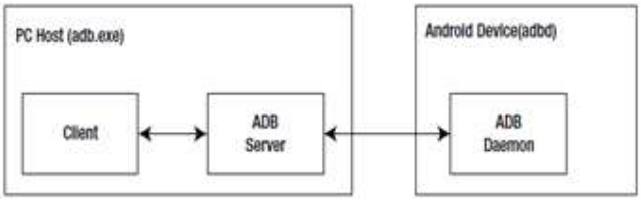

Figure 2: Client Server Architecture of ADB

A server runs in the background of the host system and communicates between the client and the ADB daemon running on an Android device. The server also maintains details and state of the devices that are connected to the computer system. Two types of services are provided by ADB commands, the commands that are serviced within the server without communicating with the Android device is referred to as host service and the commands that are serviced after fetching information from the device through the daemon are termed as local services.

Some of the ADB commands that are used in Automating test actions on the phone side are “adb devices”, “adb shell input keyevent”.

The “adb devices” command is used to list out the number of devices attached to the computer system. This command also prints the device identification number of the connected Android mobile phone. Some ADB commands requires deviceidentification number of the Android phone attached to the system in order to carry out actions, in such cases this command is used. This command is a good example of host service.

“adb shell input keyevent” is the command that is used to send event code to the android phone. Event codes are just a number, for example 26 is event code to power up android devices, similarly there are various event codes that are can be used to carry out test action automation in android phones. Thiscommand is a good example of local service as the computer system communicates with the daemon in the android device.

Android phones can be connected to the computer either by a TCP port or by USB. USB (Universal Serial Bus) connection is preferred method to connect the android phones to the computer as shown in figure 3.

Figure 3: Android phone connected to the computer.

IV. CONCLUSION

Many of the stress and performance based test cases can be automated using this method. However Automating some of the requirements cannot be covered by using this method of automation. For example, some of the requirements include testing the contents that is being displayed on the infotainment screen. Such as, testing and verification of the signal strength, battery status, of the connected Android mobile phone on the display of the infotainment system, Verification of the contact names that is being displayed on the HMI during outgoing or incoming voice call.

REFERENCES

[1] Akshay Pola, “Automated Testing of the Features of InfotainmentSystem on the Basis of Graphical Human

MachineInterface", IJSRD, National Conference on Advances in Computing and Communications, October 2016.

[2] Ihsan F. I. Albittar , Jiho Kim, HyungWon Kim, "The Design of 13 bits ΣΔ ADC for a mutual-capacitance large touchscreen controller", SoC Design Conference (ISOCC), 2014 International.

[3] Sushant Patil,Urvil Parmar , Rohit Karegaonkar,"Automated software testing for high-end touch screen automotivedisplays", Transportation Electrification Conference (ITEC), 2015 IEEE International.

[4] Linshu Duan, Alexander Hofer, Heinrich Hussmann, "Model-Based Testing of Infotainment Systems on the Basis of aGraphicalHuman-Machine Interface", Second International Conference on Advances in System Testing and ValidationLifecycle 2010.

[5] Gyeongseop Choi, M.G.A. Mohamed, and HyungWon Kim, "Distributed Architecture of Touch Screen Controller SoCfor Large Touch ScreenPanels", SoC Design Conference (ISOCC), 2014 International.