Smart Connectivity for Automotive Head Unit

Nagaveni Mallikarjun
P.G Student : I.S.E Dept.,
National Institute of Engineering,
Under VTU, Belagavi,
Mysuru, India
nagaveni1292@gmail.com

Prajakta Madankar
Assistant professor: I.S.E Dept.,
National Institute of Engineering,
Under VTU, Belagavi,
Mysuru, India
prajakta270976@yahoo.com

Basavaraj Havale
P.G Student : I.S.E Dept.,
National Institute of Engineering,
Mysuru, India
bvhavale90@gmail.com

Abstract **-** In-vehicle infotainment (IVI), is a combination of entertainment and information. The higher end, luxury automobiles incorporate safety and connectivity along with the advanced audio/visual entertainment. The future of car radio includes integration of more comfort maximizing features meanwhile ensuring safety of the drivers and passengers. The major safety concern for a driver is the distraction due to the cell phone usage during driving. The driver of the car is deprived of his independence of using cell phone due to safety reasons. This can be resolved if the applications on the mobile phone can be used on the car's head unit. The feature which allows this implementation is Smart Device Link (SDL). SDL is an innovation in the automobile industry. The basic requirement of this feature is that the car radio's display unit should have a touch screen head unit. SDL can be integrated into any vehicle's system to provide communication to a mobile application, with the condition that the mobile application itself is also SDL capable. By SDL capable we mean to say that the mobile application should be compatible with the vehicle's head unit (HU). SDL manages the transport, connection and communication between a head unit and mobile device.

Keywords **-** *In-vehicle infotainment , Smart Device Link, Head Unit, Human Machine Interface, Remote Procedure Calls*

*****

I. INTRODUCTION

Nowadays In-vehicle Infotainment is flourishing with audio and/or audio/visual entertainment, as well as automotive navigation systems (SatNav). This includes playing media such as CDs, DVDs, Freeview/TV, USB and/or other optional surround sound, or DSP systems [1]. Also increasingly common in car entertainment are the incorporation of video game consoles into the vehicle. The higher end automobiles incorporate safety and connectivity along with the advanced audio/visual entertainment. With the advances in technology, the automotive industry is moving towards the radios with more integrated features on them.

The present day radio in the automobiles has features like :

- Tuner sources - FM, AM, XM, DAB, Pandora, Stitcher
- Media Sources like CD/DVD players, USB Audio, Picture, Video, SD Card support, iPod, AUX
- Wireless connectivity – Bluetooth : Hands-free calling, Phone book access, Internet
- Telematics: GPS, Modem
- Speech Recognition
- Reconfigurable displays
- Rear-seat entertainment
- Traffic control radar
- Vehicle to vehicle communication etc.

SDL is an extensible method for a HU and a mobile application to communicate and present data to each other in an effort to present a mobile application to a driver inside a vehicle. It allows the user in the vehicle to control the application in his mobile phone via the vehicle's HU. Mobile applications are of three kinds : media, non-media and navigation applications. SDL supports media and navigation applications. For a media application, the user can control the play, pause, play next, play previous and so on from the vehicle's HU. For a navigation application, the map will be displayed on the HU and the user can perform the zoom in and zoom out and other operations through the HU.

Any reference to a mobile application herein implies an SDL - enabled application, indicating one that is modified to send the proper messages to a head unit equipped with SDL.

The transport, connection and communication between a HU and mobile device is managed by the SDL Core. A transport represents a physical connection. It communicates with the HU through a HMI (Human Machine Interface) Handler which arbitrates the HMI resources on behalf of SDL Core. It is the component responsible for communication with the mobile device's applications from a head unit perspective. SDL core must be integrated into an HU and a corresponding HMI handler. The HU integrated with the SDL Core will contain a listing (text, iconic, or both) of all the SDL applications identified on connected mobile devices [2]. SDL consists of a message handler (to handle application level key/value parameters), a protocol to packetize data for a given transport, and a transport manager (to select and send data over a given transport).

II. SMART DEVICE LINK

Smart Device Link is a project which intends to standardize and wrap the many in-vehicle interfaces which exists in the automotive context. The end goal is to provide an expandable software framework to both mobile application developers and automotive HU creators for the creation of applications that appear integrated onto a HU.

SDL is an OS-independant definition of remote procedure calls ("RPC") which is common for in-vehicle use cases, such as writing to the display, accessing buttons, creating menus, using voice controls and outputting audio to the vehicle speakers. These RPC's allow a mobile application to combine its input along with vehicle data and generate output to the HU, giving the appearance of an integrated solution which is actually driven from an application on the mobile device.

SDL uses a template-based approach to HMI and application design. On the vehicle's HU each mobile application starts with a home screen which can then be navigated via an associated menu structure. Additional HMI elements can be added , such as pop ups, sliders, dialog boxes or images which can be accessed via additional RPC's. The SDL philosophy starts with a base of RPCs that have a defined format. Within this format, it's software recognizes the function type, (e.g. notification, request, response) its purpose, (e.g. write to display, show a popup, listen for button event) and the parameters of that specific function (e.g. text to write, duration of popup, button pressed). Each RPC (or API offered to a developer on a mobile application) has a corresponding function or result on the HU inside a vehicle. Core to the SDL philosophy is the ability to extend SDL with new RPCs as well as add parameters to existing RPCs.

For a mobile application to be SDL capable, the application developers have to simply add the required SDL software to it which makes it compatible with SDL HU. End users can download the application from existing distribution channels (e.g. Google Play, iTunes App Store, etc.) and connect their mobile device to their vehicle HU. The mobile device does not require certification, and no additional applications or services need to be downloaded to the mobile device for the applications to work.

Fig 1 : connection establishment and data transfer between mobile phone and vehicle HU over BT

Smart Device Link consists of two distinct parts of software, a core automotive piece which wraps in-vehicle services and integrates to the in-vehicle HMI, and a mobile proxy, which executes APIs to access and use services on the automotive HU. The in-vehicle core has some of the following features:

- provides the physical connection and the logical transport connection to a mobile device application
- supports single app, single function use cases as well as multi-function but single application use cases
- a consistent interface for applications is provided to understand the capabilities of an in-vehicle HU
- the state of connected mobile applications is continuously monitored and the applications are notified of the appropriate state changes
- the in-vehicle services such as buttons, displays, voice, menu systems, audio controls and other common vehicle inputs and outputs are wrapped into a common messaging format and API
- a template and meta-based UI is created that creates consistent developer experiences regardless of the actual HMI representation

The mobile proxy components include some of the following features:

- provides a common, abstracted interface to the in - vehicle infotainment system to integrate to Smart Device Link enabled mobile applications
- is available for Android and iPhone applications
- it supports Bluetooth, USB and Apple-specific protocol implementations for transport, and can be expanded to include other transports
- it simplifies transport discovery, connection, and protocol use
- Is expandable to allow for specific-OEM or HU manufacturer APIs or messaging

III. SDL ARCHITECTURE

The architecture and design of a HU integrated with SDL is depicted in the below block diagram. The flow of control and information between the mobile and the HMI is also shown.

- The upper layer is the HMI and the lower layer contains BT and USB stack.
- The S/W architecture Adapter is the interface between the HMI and the SDL
- The SDL can be again sub divided into 3 layers as shown in below figure
- The HMI Message Handler as the name indicates sends the messages from HMI to SDL and vice versa
- The Application Manager is the brain of SDL that takes all the decisions
- The Transport Manager is like a transport layer and handles the flow of messages

- The mobile phone can be connected to the car radio either through USB or Bluetooth (BT)
- The two stacks are the interfaces between the Transport Adapter and the mobile phone

SDL must be integrated to an infotainment systems transport mechanism and provide various services for the purposes of device discovery, end point discovery, or other. Currently Android applications communicate with SDL over Bluetooth and USB, and iOS applications utilize the iPod Accessory (iAP) protocol over USB.

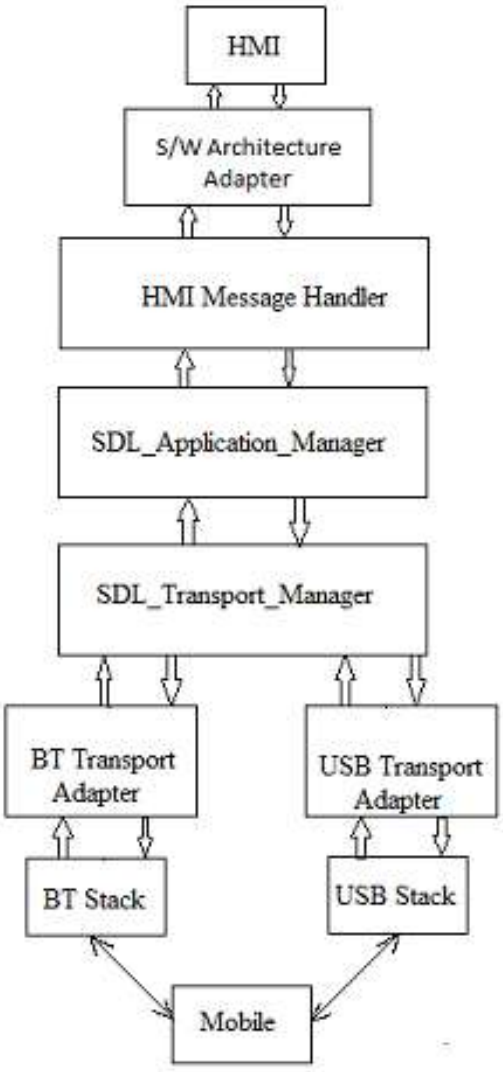

Fig 2. Block diagram depicting flow of control between mobile and HMI

IV. SDL CORE

A. *DEVICE CONNECTION*

The mobile device has to be connected to the HU. If the mobile device consists of any applications that are SDL compatible then it is acknowledged by the vehicle's HU. If not it will be treated as a MTP device.

Applications in the phone will make requests to the vehicle and the vehicle responds using the RPC's (Remote Procedure Calls). Application appears to be running inside the vehicle, but all the logic is controlled by the mobile device.

B. *INTERACTION*

Once the application gets connected to HU, user interacts with the application via the vehicle's HU and the HU notifies the application of any system or user events.

Application reacts to the messages sent by HU and also sends request to modify application behavior or appearance in the vehicle.

C. *TRANSPORT MANAGER LAYER*

The function of Transport layer as the name indicates is to control the flow of data or messages. The events or messages or data sent from phone to HU or vice versa is handled by the transport layer or the transport manager. Each application is responsible for establishing a transport connection either through Bluetooth or USB.

HU and mobile device may have a separate connection for existing features in the automobile industry such as hands-free profile(HFP), mass transport(MTP), etc.

The currently supported transports are as listed below :

- Android phones connected via Bluetooth use RFCOMM protocol
- Android phones connected via USB use AOA (Android Open Accessory) protocol
- iOS or Apple phones connected via USB use iAP protocol
- SDL does not support Bluetooth connection for iOS phones

The Transport Manager Layer differentiates between the data to be sent to BT and USB Transport Adapter Layer's accordingly depending on the type of connection between phone and HU.

The TM layer does not differentiate between the data sent from lower to upper layer, i.e the data from USB and BT Adapters appears to be same for TM layer.

D. *TRANSPORT ADAPTER LAYER*

The TA layer is connection specific i.e there are separate adapters for BT and USB connected devices respectively.

For a mobile device connected over BT:
The system must perform the following steps to connect to a mobile device over BT :

- Iterate through a known list (paired or pre-set) or available list of discoverable devices

- Identify the applications or services on the device that are SDL capable by performing a query on the device
- Connect to the user specified application on the device [6]

For a mobile device connected via USB :

- When a device is plugged in via USB, the device is discovered.
- The protocol supported by the connected device is identified, i.e AOA protocol for Android devices and iAP protocol for iOS devices.

- Application queries for the available protocols and chooses one for its connection, this establishes the connection between the mobile device and the HU.

E. APPLICATION LAYER

The data from phone is decoded in Application Layer and the corresponding commands are sent to HU via the Message Handler. Thus the actions performed on the phone appear on the HU. For eg : the swipe action done on the navigation screen or the play/pause actions done on a media application on phone side is reflected on the HU.

Similarly the stack in the lower layer converts the data coming from the upper layer into appropriate format supported by mobile and sends to it.

F. SESSION ESTABLISHMENT AND APPLICATION REGISTRATION

A session has to be established between the mobile application and the HU for the mobile application to get registered on the HU. A transport is a physical connection and a session is a logical connection. Services of Application Layer are carried out over a given session.

The request for session establishment is sent by the mobile application. Depending on the availability of resources the HU decides whether to acknowledge or not acknowledge the request to start a session. Each established session has a unique session ID. Once a session is established, the existing RPC services are used to register the application and provide commands to the HU. After the session has ended the resources acquired by the session are made available.

G. REMOTE POCEDURE CALLS

SDL APIs are request/response driven. The following are few of the RPC's sent by mobile application to the HU. Some of these RPC functions have HMI output, while others will change the internal state of SDL or the in-vehicle HU.

Requests/Response RPC's :

- Register App Interface Request : Provides the HU with the application's name, type and other information used for establishing the interface that is used.

- Register App Interface Response : Provides the application with settings and configuration of the HU, such as the language, supported buttons, type of display, and additional information.

- Add Command : Adds a command or function to the application's Command Menu.

- Add Sub Menu : Adds a Sub Menu to the Command Menu. A Sub Menu can only be added to the Top Level Menu and may only contain commands as children.

- Delete Command : Removes a command from the Command Menu

- Delete Sub Menu : Deletes a submenu from the Command Menu.

- Register App Interface : Sends a request to register an application interface

- Set Media Clock Timer : Sets a media clock timer available on media displays along with a method value (to count up or down from a given start time)

Notification RPC's :

- On App Interface Unregistered : Sends a notification on termination of an application's interface registration to mobile.

- On Button Press : Sends a notification to mobile on press of a button on HU.

- On Command : Sends a notification to HU on send of a command by mobile device.

Each Remote Procedure Call message is a JSON message structure . Each JSON message is structured in the following format {"key":"value"}.

H. POLICIES

Listed below are the basic policies of SDL :

- Management of application : it manages which App works in vehicle and which does not

- Control the App Functions : it controls what features or functions App can access and also manages the concerns or user notifications

I. INTERFACES

Any SDL applications must also be notified via existing notifications of system events or state changes which may impact the application. This may include, but is not limited to:

- Power state changes of the vehicle (i.e. ignition on/off) [8].
- Notifications from phone or other application priorities.
- State changes from audio source, attenuation in audio source, or loss of source.
- Change in vehicle data or subscriptions from vehicle data

V. CONCLUSION

Smart Device Link is a novel feature in the automotive industry introduced for the benefit of the drivers. It minimizes the distraction due to the use of mobile phone in the vehicle. The mobile application developer only has to integrate the mobile proxy SDL layer during developing an application to make it compatible with the vehicle's Head Unit which itself is SDL capable. SDL is developed to serve as an intermediary between vehicle's HU and an application that runs in any of the mobile devices. The device might be any of the Android or iOS phones and it should be connected via Bluetooth or USB. SDL system allows the application to use vehicle HMI features such as Voice Recognition, Text To Speech conversion, buttons (hard keys and soft keys), vehicle display, audio system etc and also operate with Vehicle Data (seat belt position, transmission shift lever position, airbag status, etc.).

REFERENCES

[1]. William B. Ribbens and Norman P. Mansour (2003). Understanding automotive electronics (6th ed.).
[2]. Enabling the Next Generation of Automotive Head-Up Display Systems, Application Report DLPA043–October 2013
[3]. "Cars Get Internet Connection, Study Says" : Automotive-eetimes.com. European Business Press, 2008-11-07.
[4]. Making Embedded Systems Design Patterns for Great Software *by Elicia White*, O'Reilly Media, October 2011
[5]. Bluetooth Profiles : The Definitive Guide by Gratton, Dean A, Prentice Hall, December 2002
[6]. UNIX Network Programming, Volume I, Second Edition : Networking APIs : Sockets and XTI, Prentice Hall, 1998
[7]. Programmable Logic Controllers and Industrial Automation by Madhuchhanda Mitra and Samarjit Sengupta.
[8]. Vehicular Technologies – Deployment and Applications by Lorenzo Galati Giordano and Luca Reggiani
[9]. Delphi News Letter