

Network Architecture of a Modern Automotive Infotainment System

Mashud Kabir*

Design & Evaluation Lab, Panasonic Automotive System Europe, Robert-Bosch-Strasse 27, Langen, Germany

Abstract

Due to the advancement in technology and consumer electronics, automotive infotainment system is growing rapidly. More and more new technologies and devices are integrated into the system. All the infotainment electronic control units (ECUs) are networked in the automotive platform. Therefore, it has always been a challenging topic for researchers and engineers to find an appropriate network architecture which can adjust with the newly added ECUs and user demands. Until the last few years, automotive devices were integrated using Controller Area Network (CAN) technology. Due to the slow speed of data transfer in Controller Area Network (CAN) (generally up to few hundred Kilo Bytes per second), a new technology named as MOST (Media Oriented System Transport) was introduced. Here, the data is transferred optically. For the last few years, more and more applications including Navigation system, Multi Media Player, iPod, SD Card, USB Memory, Instrument Cluster, Digital Tuner and Telephony have been integrated into the MOST system. Data transfer rate of up to 150 MB its per second is realizable using MOST. In this paper, MOST network architecture which is the backbone of modern infotainment platform, network management protocols and related devices are presented.

Keywords: Automotive multimedia; Network; Architecture; Communication layer

Introduction

This paper presents at first an overview of automotive infotainment features to get a general understanding of the system. Then infotainment network system with its nodes is discussed. As MOST is the main focus of the modern network architecture concept, its components along with their applications are explained. Then MOST communication layer model based on classical OSI layer model is explained.

Infotainment and Multimedia Features

To get a good understanding of infotainment and multimedia system, it is necessary to know the features in this area. The section describes the features based on the types of application modes. All features that the Infotainment and Multimedia system provides are realized by software components which are deployed on car devices and service centers. The features can be grouped as follows:

- i. Entertainment services (e.g. Radio, DVD, TV, Digital Radio (DAB), Rear Seat Entertainment, SDARS,SIRIUS, iPod, USB Memory, SD Card etc.)
- ii. Information services (e.g. Navigation including route planning, dynamic route guiding and travel guidance)
- iii. Telecommunication services (e.g. voice telephony, SMS, E-Mail, Remote Climate Control and Remote Door Lock/Unlock, Browser List)
- iv. Advanced use of car functions (e.g. adjustment of external mirrors, climate control, seat adjustment)
- v. Safety services (e.g. Emergency-call, Car Tracking)
- vi. System administration services (e.g. resource management, channel allocation/deallocation, software download and engineering mode).

Infotainment Network System

The infotainment system can be characterized as a distributed, heterogeneous hardware/software system, which provides functionalities and services to the customer via the interaction of several devices. The

hardware components are interconnected by standardized car networks (e.g. MOST, CAN bus) [1] and by wireless communication channels (e.g. Bluetooth). An overview of the main components of Infotainment & Multimedia in a passenger car is presented below:

Head unit

Head Unit (HU) is regarded as the master in the network. It contains one or more CPUs. The HU has atleast one CPU for real time communication purposes (e.g. infotainment gate way) and at least one CPU for the user interface to interact with infotainment applications (Main CPU).

Infotainment & multimedia gateway

Infotainment & Multimedia Gateway contains at least a CPU (Gateway CPU) for real time communication with other peripheral components like memory, CAN transceivers, MOST transceiver and other components. The gateway CPU is connected to the main CPU with an internal bus. As the main CPU is not connected directly to the MOST command/control channel and the CAN network [2], all messages from the Main CPU to the outside world are sent and received over the gateway CPU. Figure 1 shows that a HU must contain a main CPU and a Gateway CPU.

Can network

One of the main tasks of HU is to control the basic vehicle functions. The gateway is not connected directly to the control units of vehicle functions. It is connected to the Central Gateway (it manages the other CAN networks in the vehicle) by a high speed CAN (Backbone CAN).

*Corresponding author: Mashud Kabir, Design & Evaluation Lab, Panasonic Automotive System Europe, Robert-Bosch-Strasse 27, Langen, Germany, E-mail: mashud.kabir@ext.eu.panasonic.com

Received April 21, 2012; Accepted July 26, 2012; Published July 30, 2012

Citation: Kabir M (2012) Network Architecture of a Modern Automotive Infotainment System. Adv Automob Eng 1:101. doi:10.4172/2167-7670.1000101

Copyright: © 2012 Kabir M. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

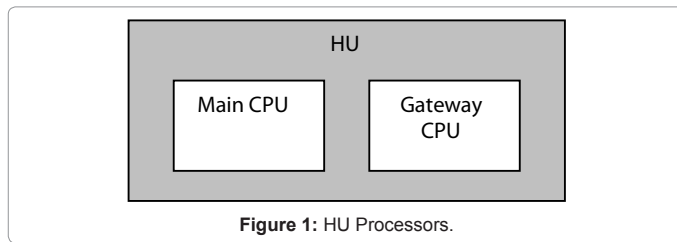


Figure 1: HU Processors.

The HU is connected with the HUCAN. Some control devices on the HU-CAN communicate only with the HU. There are some other devices for which the messages are only routed through the gateway from the HU-CAN to the Backbone-CAN and vice versa. The CAN messages and signals are specified in some matrices.

Most network and protocols

MOST (Media Oriented System Transport) is a high speed network, which connects infotainment components with the central unit (HU) through the gateway [3]. It is a closed-loop network. The gateway must be able to handle command/control data of MOST [4].

A MOST device is a physical entity which may include multiple functional components. These components are called Function Blocks (FBlock). For example, the MOST device Rear Seat Entertainment (RSE) consists of several FBlocks such as MultiMedia Player, HeadPhone Amplifier, and Auxiliary Input etc. An FBlock is identified with special ID names such as FBlockID.

In a MOST system, a specific FBlock may exist more than one time. To avoid any conflict among them, each FBlock is assigned with an instance identifier referred to as InstID. The combination of FBlockID and InstID makes an entity unique for addressing. This is termed as functional address. An FBlock consists of many functions, like mute, volume, allocate etc in MOST FBlock HeadPhoneAmplifier. In order to distinguish between different functions, every function has its own Identity which is called FktID.

With the help of a function, certain operations are performed on respective properties or methods. The kind of operation is expressed by OP TYPE, e.g. Start, Get, StartResult etc. Each OP TYPE may contain several parameters. To perform a specific operation on a specific function block, a message of the following format is used:

<FBlockID.InstID.FktID.OpType (Data)>

To realize an application of a MOST device, the above structure is implemented according to the detailed guidelines in a function catalog. A function catalog lists all the FBlocks, instances, functions, OP TYPES and parameters.

From the MOST application point of view, a MOST device can be said to be error free if the complete function catalog functions correctly. Therefore, it is of prime importance to test the function catalog thoroughly. This test is also called a Static Function Test, as a fixed set of functions are tested here.

The test of the function catalog starts by reading the Device Identity such as hardware/software/boot loader version from the MOST devices. Then all the implemented and non-implemented FBlockIDs (0x00-0xFF) are verified. Non-implemented FBlockIDs are tested to fulfill negative requirements where an appropriate error message is expected.

Each existing FBlock in a MOST device has certain InstID(s). InstIDs are verified by keeping in mind the assigned instances for

an FBlock, wild-cards (normally InstID 0x01n and 0xFF are defined as wild-cards) and all incorrect instances. The FktIDs are verified for positive and negative test cases to ensure that project specific FktIDs exist, and the other FktIDs do not exist in the FBlock. In addition to the project specific function catalog, a general function catalog obtained from MOST Cooperation is used to decide which functions should be taken to test negative cases. Usually the unused functions in the general function catalog are considered for negative tests. The same test strategy is also applied for OP TYPE verification.

Automotive infotainment system consists of several ECUs (Electrical Control Units). Some defined protocols should be developed to manage the connected devices. These sets of protocols can be termed as Network Management. Network management tasks include startup, shutdown, wakeup, power moding, addressing, allocation, de-allocation, audio management etc.

Startup behavior manages the by-pass close/open, timely response for NetBlock queries, registration of FBlocks, configuration control, resource de-allocation etc. NetBlock is a special function block, which provides functions related to the entire device. Each MOST device must contain this FBlock. Shutdown behaviors deal with the proper saving of current settings, communication ability, attachment of FBlocks in the central registry, light off behavior, whereas wakeup behaviors set the guidelines for light travel time from input to output in ECUs, lock/unlock, light off etc.

Power moding behaviors deal with the device characteristics in under voltage and over voltage. It also defines the consumption of power during normal mode and sleep mode. Network management address ingensures the correct sending and receiving of messages in the system.

Allocation, de-allocation and audio management are the areas where sources, sinks, resource management are handled.

The HU contains the MOST FBlocks and controllers, which are necessary for controlling the network behaviors of the HU.

Below is a list of logical interfaces for some common devices:

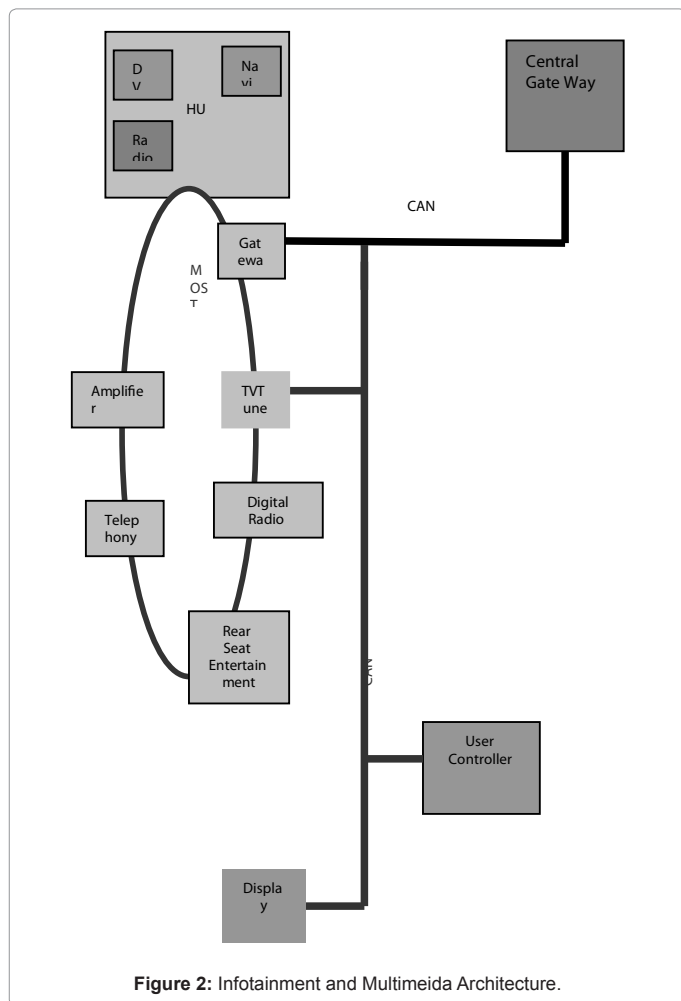
- Head Unit (through MOST)
- Speech Control System (through MOST)
- Mobile Phone (cradle or Bluetooth)
- Instrument Cluster (through Head Unit)
- Amplifier/Sound (through MOST)

Most Network Architecture

A MOST device contains a receiver and a sender. The input of a most device is optical light and the output is also optical light. The light is modulated inside the device to get or send the necessary information. It is essential that a MOST device is stable during operation. It means there must not be any light off, unlock without any specified and proper reasons. A light off or an unlock in a single device would make the whole system unstable. A typical MOST ring contains the following components which is depicted in Figure 2.

The components are described below:

- i. Amplifier: Amplifier is a mandatory ECU in MOST infotainment system. This works as a sink of audio data.
- ii. Telephony: Telephony contains a mobile Interface. This is an



optional device in a MOST ring. The interface connects the Bluetooth-enabled mobile phone and other MOST components.

iii. Rear Seat Entertainment: Rear Seat Entertainment (RSE) is an optional device. It may contain MultiMedia Player, HeadPhone Amplifier, and Auxiliary Input.

iv. TV Tuner: TV Tuner provides reception for video and audio. This is an optional component in a MOST ring. A TV Tuner can have variants of digital, analog or hybrid (analog/digital). The video output is sent to display and audio is sent to amplifier.

v. Gateway: This is a mandatory component in MOST infotainment system. Gateway is the interface between MOST and CAN network. MOST commands are translated into CAN commands whenever any command is sent from MOST to CAN. Similarly, CAN commands are translated into MOST commands if the target of the command is an element of the MOST network.

vi. Radio: Radio is a mandatory element of MOST. It can be an analog or digital tuner. This is a source for audio data which is transferred to an audio amplifier.

Software Download and User Controller (UC) are two commonly used terms which are briefly described below:

Software download

Software download means flashing software into Infotainment & Multimedia devices. Providing an interface to download software into Infotainment & Multimedia devices permits the separation, to a certain degree, of the hardware development cycle from the software development cycle and adds new (updated or additional) functionalities without changing the hardware.

User controller (uc)

The user control (UC) is a mechatronic interface between the head unit and the user. User commands can be provided through the movement of the buttons and pressure-operated rotary actuator. The commands are transmitted to the HU via CAN-bus.

Communication Layer Model of Most Network

This section discusses the Infotainment and Multimedia layering in an automotive system. At first a brief overview of OSI communication layer model is discussed. Then our infotainment and multimedia layering is presented.

In the OSI communication layer model, there are seven layers which is depicted in Figure 3.

The Physical Layer describes the physical properties of the various communications media, electrical properties and interpretation of the exchanged signals. For example, this layer defines the size of Ethernet coaxial cable.

The Data Link Layer defines the logical organization of data bits transmitted on a particular medium. For example, this layer describes the framing and addressing of Ethernet packets.

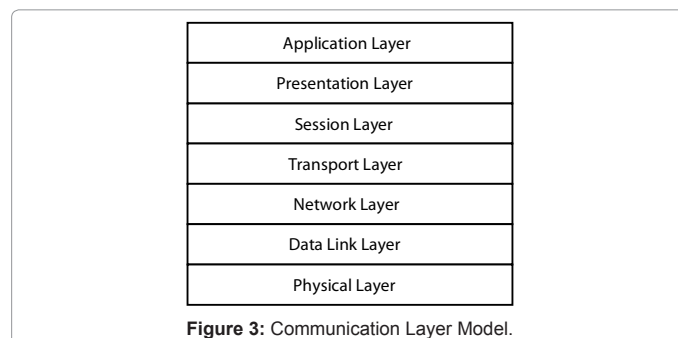
The Network Layer explains how a series of exchanges over various data links can deliver data between any two nodes in a network. For example, defining the addressing and routing structure of the Internet is one of the main tasks handled by this layer.

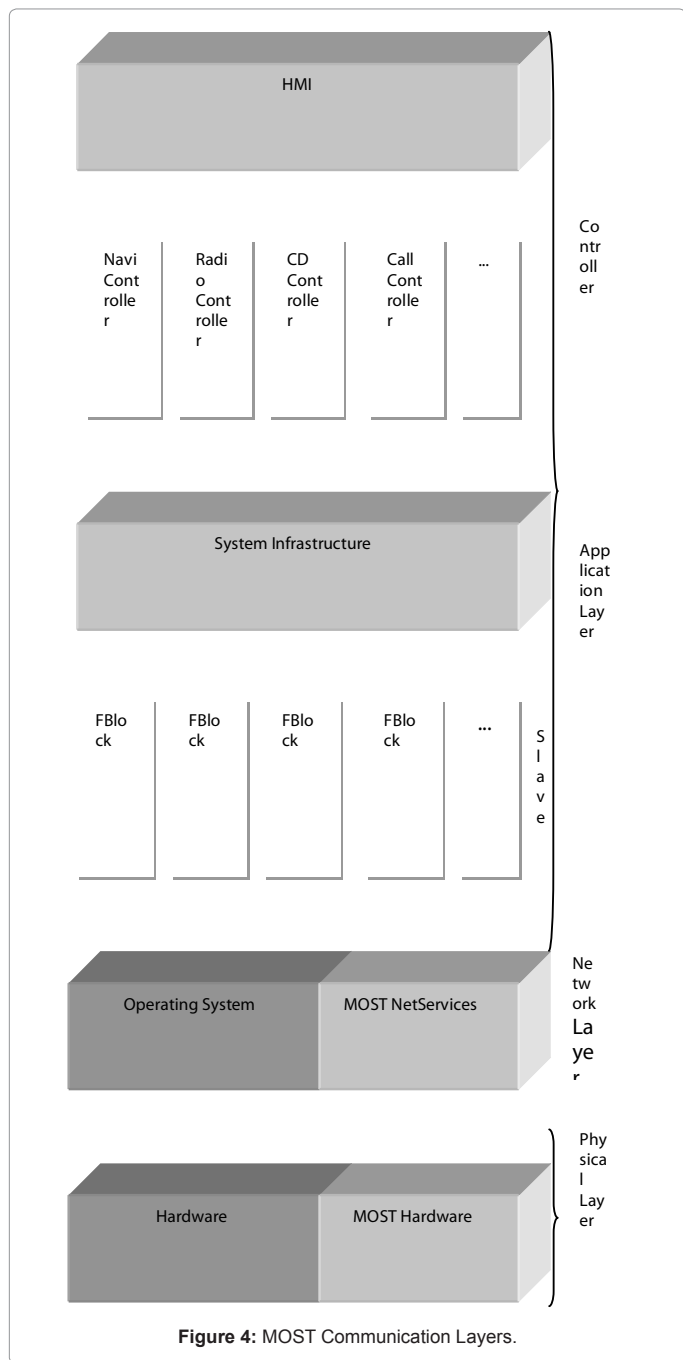
The Transport Layer describes the quality and nature of the data delivery. For example, this layer defines if and how retransmissions will be used to ensure data delivery.

The Session Layer describes the organization of data sequences that are larger than the packets handled by lower layers. The tasks of this layer include how request and reply packets are paired in a remote procedure call.

The Presentation Layer describes the syntax of data being transferred. For example, this layer describes how floating-point numbers can be exchanged between hosts with different math formats.

The Application Layer describes how real work actually gets done.





As for example, this layer implements file system operations.

Based on the above discussion regarding the OSI communication layer model, a general layer model for Infotainment & Multimedia system based on MOST is presented here.

MOST communication network is layered into three parts: i. Physical layer ii. Network layer iii. Application layer. Figure 4 shows the MOST communication layer model.

i. Physical Layer

Several Infotainment and Multimedia hardware platforms are distributed over the car and in the infrastructure. The devices integrated in the vehicle are inter connected by MOST and CAN buses. These in-

car platforms constitute the physical layer. The physical layer deals with the following:

- Transmitter
- Receiver
- Device Connector
- Cabling
- Power Budgeting
- Signal Timing
- Diagnostic Requirements
- Component Specification
- System Description
- Coordination

ii. Network Layer

The communication stacks and MOST/CAN network management functions as a whole constitute the network layer. The network layer deals with the following:

Net Services - Communication between devices is realized in the majority of cases via the MOST NetServices. There is also a Gateway to communicate with CAN devices. For further communication purposes via other networks additional protocol stacks, e.g. WAP, are implemented.

Transport Protocols - Transport protocols deal with control and segmented data communication over the control channel, the amount of data for such a connection, MOST High Protocol as transport protection for streaming and application data on transport layer level.

Boundary - The boundary for the distribution of synchronous and asynchronous channels is to set to a fixed number.

MOST network management level - MOST Address Handling, Instance ID Handling and FBlock Naming.

Network Master - After the MOST network on a basic level is ready for communication (NetOn), the network master checks if it has a valid logical node address. It also manages the following behaviors of a device:

- IP Address Handling
- MAC Address Handling
- Low Voltage Device Behavior

iii. Application Layer

The application layer is at the top of the system layer. It consists of the system infrastructure – including infrastructure software components (e.g. Resource Management - providing functionality) generally required in application components or required to ensure that the system as a whole functions properly. It consists of general requirements, architectural requirements, and a detailed specification of functionalities and interfaces.

The application layer deals with the followings:

- Application components (controllers and slaves) focus on the functionalities and logics required for a specific application.

- The MMI framework is responsible for the handling of interaction between the infotainment and multimedia system and its users.

Conclusion

In this paper I have described the features of a modern automotive infotainment and multimedia system. Existing network concept is not capable of handling and adopting the new electronic devices which require high data transfer rate. Therefore, existing CAN is replaced by MOST. MOST is relatively a new area in automotive sector. This is a ring topology. A modern MOST network architecture has been

presented here. Based on the classical OSI communication layer, a three layer model for automotive infotainment network has been discussed.

References

1. Bell J (2002) Network protocols used in the automotive industry.
2. Rusek A, Oakley B, Stevens D, Hillier L (2009) Computer Modeling of CAN Automotive Bus Transceivers.
3. Schöpp H (2001) Practical Aspects of Implementing MOST Systems and Devices MOST Devices.
4. OASIS, SMC (2009) Automotive Infotainment Solutions.