Advances in Networks
2018; 6(1): 58-65
http://www.sciencepublishinggroup.com/j/net
doi: 10.11648/j.net.20180601.15
ISSN: 2326-9766 (Print); ISSN: 2326-9782 (Online)

Review Article

Communication Technologies and Network Protocols of Automotive Systems

Aroosa Umair*, Muhammad Gufran Khan

Department of Electrical Engineering, FAST National University of Computer and Emerging Sciences, Islamabad, Chiniot-Faisalabad Campus, Pakistan

Email address:
aroosa.umair@nu.edu.pk (A. Umair), m.gufran@nu.edu.pk (M. G. Khan)
*Corresponding author

To cite this article:
Aroosa Umair, Muhammad Gufran Khan. Communication Technologies and Network Protocols of Automotive Systems. *Advances in Networks*. Vol. 6, No. 1, 2018, pp. 58-65. doi: 10.11648/j.net.20180601.15

Received: May 31, 2018; **Accepted**: June 19, 2018; **Published**: July 11, 2018

Abstract: Automotive industry has gone through rapid changes in the past few years. The usage of electronics and electronic control units (ECUs) have increased manifold, and this has also affected the way different subsystems communicate. Communication technologies and protocols are required to fulfill demands of fault-tolerance, dependability, bandwidth and determinism of demanding and safety-critical applications. This paper presents a survey of state-of-the-art and the most commonly employed communication technologies and protocols; both wired and wireless for in-vehicle and vehicle to vehicle (V2V) communication in the automotive systems. The technologies such as LIN (Local Interconnect Network), CAN (Controller Area Network), MOST (Media Oriented Systems Transport), and Flexray are compared in terms of the performance, reliability, cost and protocol characteristics. The study shows that Flexray is an excellent network topology for in-vehicle communication that has higher degree of fault tolerance, and is suitable for hard real time systems with high bandwidth. Moreover, wireless technologies i.e. Bluetooth, ZigBee, Wi-Fi and UWB are discussed that satisfy different requirements of diagnostics and multimedia communication for in-vehicle and vehicle to vehicle communication and can be used for advanced autonomous driving systems. The paper also presented issues that need to be addressed to fully realize the potential of these communication technologies and other advancements in automotive industry.

Keywords: Communication Technologies, Automotive, Electric Control Units (ECUs), Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), Wireless Networks

1. Introduction

Advances in the field of electronics, control and embedded systems has greatly impacted the automotive industry in recent years. Cars of today have become smarter with the advanced driver assistance systems (ADAS) such as collision avoidance system, automatic parking, lane departure warning, pedestrian detection and adaptive cruise control etc. Significant developments have also been made in the area of vehicle dynamics, energy efficiency and emissions reduction. Another trend is to replace mechanical and hydraulic parts of the cars with electronic components. The approach of using electronic control systems instead of traditional mechanical control systems is called x-by-wire or drive-by-wire (DbW) technology [1]; for example electronic throttle control is called throttle-by-wire. Some other examples of such systems are brake-by-wire and steer-by-wire.

Likewise the number of electronic control units (ECUs) integrated in contemporary cars has increased a lot. A typical automotive system contains around 100 ECUs and thousands of communication signals [2]. These software-intensive embedded systems put great challenges on the design of system architecture and communication mechanism. To handle this level of complexity, many efforts have been done by the automotive industry to standardize the software architectures as well as the communication protocols. Recent developments in this regard are AUTOSAR (AUTomotive Open System ARchitecture) software architecture [3] and

FlexRay automotive network communications protocol. AUTOSAR standard provides a common, scalable and integral platform for automotive software development.

Many functions in an automotive system are distributed over several ECUs and control algorithms are also dependent on different subsystems [4]. This not only increases the complexity but also requires a reliable communication architecture which consists of a hierarchy and interconnection of different network technologies. As different subsystems and applications have different requirements and constraints, these factors need to be considered in the design of communication architecture. In the literature, communication technologies and related issues have been discussed in [5-7]. This paper is aimed at presenting a survey of different automotive communication technologies in contemporary automotive systems. In particular, the advantages and disadvantages of the technologies are compared and their unique characteristics are discussed and analyzed. In addition, major issues in the design and implementation of communication architecture are investigated.

The rest of the paper is organized as follows. In Section 2, typical vehicular network architecture is discussed and its different domains are presented. Section 3 provides a brief overview of communication technologies and protocols currently used in automotive systems. Section 4 finally concludes our paper and presents some future challenges.

2. Automotive System Domains

A typical automotive system can be divided in the following domains:

2.1. Body Domain

Body domain contains subsystems to control doors, seats, windows, mirrors and lights. Moreover, climate control module that controls heating, ventilation and air-conditioning (HVAC) is also part of Body domain. The controller modules are mostly implemented as state-machines and event-triggered communication is required.

2.2. Powertrain Domain

This domain consists of subsystems that control the engine and transmission of the power. It involves task that have stringent timing requirements. In addition, hardware resources need to have a higher computational power. Many functions depend on the subsystems in other domains and frequent data transfer is required.

2.3. Chassis Domain

Chassis domain includes subsystems such as suspension control, anti-lock brake system (ABS) and electronic stability program (ESP). The function of these systems is to assist the driver in steering and braking under specific conditions. Hardware requirements of the systems in this domain are similar to the powertrain domain. Further, most of these control systems require a feedback and thus have very critical timing requirements. New functions such as steer-by-wire and brake-by-wire require time-trigger communication and real-time behavior.

2.4. Vehicular Safety Domain

Systems which are used to ensure the safety of vehicle occupants can be subdivided into active and passive safety systems. Those systems which are used to prevent a crash of the vehicle are termed as active safety systems. Such systems obtain information from different sensors employed in the vehicle to assist the driver in steering and braking, or automatically controlling the vehicle to avoid a crash. Examples of such systems are anti-lock brake system (ABS), automatic stability control (ASC), adaptive cruise control (ACC), collision avoidance system and lane departure warning. On the other hand, systems that are referred to as part of the passive safety domain are air-bags and belt pre-tensioners. There is an increasing trend in the number of functions integrated in vehicular safety domain.

2.5. Human Machine Interface (HMI) Domain

HMI domain contains subsystems related to multimedia and infotainment. Due to the requirement of integrating numerous multimedia and infotainment functions in the vehicles, importance of HMI domain is increasing.

2.6. Wireless and Telematics Domain

It includes functions that require wireless connectivity within a vehicle, vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V). Wireless connectivity within a vehicle is required between devices such as cell phones, laptops and hands-free phones. V2I connectivity provides traffic information, fleet management systems and anti-theft systems. V2V communication is another emerging concept in which vehicles will form ad-hoc wireless networks to share useful information.

3. Communication Technologies

A typical layered view of an automotive communication network is shown in Figure 1. Over the years, different communication technologies have been proposed for in-vehicle communication. Different systems and functions have different requirements such as bandwidth, performance and timeliness; a single communication network cannot be employed. The most commonly used technologies in contemporary in-vehicle networks are LIN (Local Interconnect Network), CAN (Controller Area Network), MOST (Media Oriented Systems Transport), and the newest one called FlexRay. In the following, these technologies are described briefly and a comparison is presented in Table 1.

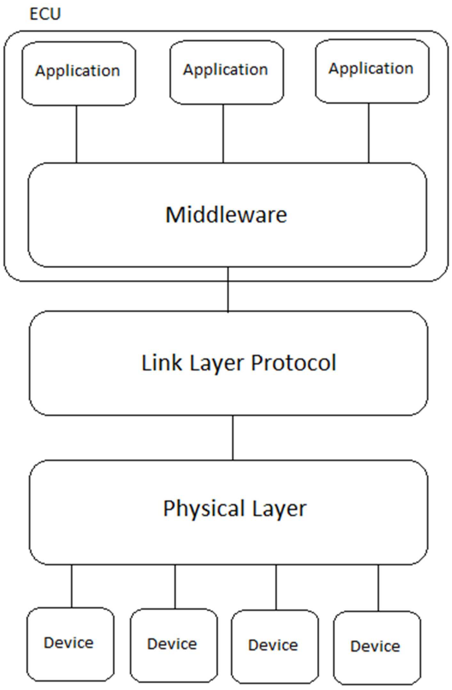

Figure 1. *An illustration of the layered view of an automotive communication network. [8].*

3.1. Local Interconnect Network (LIN)

LIN architecture, its applications for automotive systems, its protocol characteristics and its node and frame structures have been discussed here.

3.1.1. Architecture and Applications

Many subsystems and applications in vehicular networks can function properly even at low data-rates. An ideal solution is to have a low-cost network that not only fulfills the requirements of data-rate but also it is simple and easy to implement [9]. A consortium of car manufacturers, electronics and tool suppliers was formed in 1997 with this idea. The consortium finalized the specification of LIN technology in 2000. The use of LIN for low-cost applications kept on increasing and today it has become a technology of choice for low data-rate (20 Kbps) automotive applications. A large number of small mechatronic elements in body domain require only local communication. LIN networks are well-suited for such local control operations, for instance, controlling door locks, electric window operations, seats adjustments, wipers, rain sensor, light control and climate control functions.

For subsystems requiring low data-rate, expensive CAN network are replaced by small LIN sub-networks. This solution fits in at the low end of automotive networking. Interconnection among LIN sub-networks is achieved using CAN network as backbone. An advantage of creating sub-networks is to reduce the data load on the main bus. In this scenario, a hierarchical structure of CAN-LIN nodes and their sub-networks is formed. LIN nodes based on a SCI (Serial Communications Interface) are connected via a single-wire in master-slave fashion. All the nodes are connected to a common bus with one master and multiple slave nodes. The signals are transmitted by the frame transceiver using the LIN protocol.

3.1.2. Protocol Characteristics

LIN utilizes a Master/Slave protocol in which Master determines the order and priority of the messages. The Master node uses a schedule table which contains the frames to be transmitted and their associated time slots. In addition, Master also monitors Data and controls the error handling. The Clock base of the Master serves as a reference for all the communication on the network. On the other hand, there can be 2-16 Slave nodes on the LIN bus. A Slave node receives or transmits data when an appropriate ID is sent by the Master node. Moreover, a Master node can also serve as a Slave node on a LIN bus.

LIN protocol provides deterministic transmission as there is only one node that is controlling the rest of the nodes on the network. In addition, maximum latency of the transmitted signal is guaranteed. As the Master node is controlling the whole network, bus arbitration is not needed in this protocol.

3.1.3. LIN Node Structure

A LIN node structure may be divided into three layers: Application, Communication Manager and Bus Transceiver. Application layer consists of the software running in the LIN ECUs. This layer is developed according to the LIN Application Program Interfaces (APIs) provided by the standard. Moreover, messages and identifiers are also assigned at the Application level. Second level in the node structure called Communication Manager contains LIN protocol specifications. This layer is configured for LIN communication using specific configuration tools. Finally, hardware level in the node structure called Bus Transceiver is developed according to Physical layer specification of the LIN standard.

3.1.4. LIN Frame Structure

A LIN frame is transmitted within a period called frame slot, which consists of Header and Response. The Master node broadcasts an identifier in the Header part of the frame, while the Slave node corresponding to that identifier responds with data in the Response part of the frame. The Header part contains following fields:

1. Break: It is used mainly to signal the beginning of frame; contains 13 dominant bits and one recessive bit (that acts as break delimiter).

Table 1. *Comparison of Physical-layer Networking Technologies.*

Technology	Automotive Domain	Subsystems	Speed	Advantages
Flexray	Car Backbone, X-by-Wire	Brake, Steering	10 Mbps	Time-triggered, Redundancy, Safety and Fault Tolerance
HS-CAN	Powertrain	Engine, Gearbox, ABS	1 Mbps	Event-triggered
LS-CAN	Body	Doors, Roof, Dashboard, Climate	125 Kbps	Event-triggered
LIN	Body	Electric seats, Windows, Mirrors	20 Kbps	Economical, Low-complexity

2. Sync: It consists of 8 bits of alternate ones and zeros; an additional start bit (zero) and stop bit (one) as delimiters. This field helps Slave nodes in synchronization before the identifier field starts.
3. Identifier: First 6 bits of this field represent identifier and last 2 bits are used for identifier parity.

The Response part of the frame has two fields:

1. Data: It contains up to 8 bytes of data which can be either from the Master or the Slave.
2. Checksum: This field helps in error detection due to bit version during transmission. Cycle redundancy check (CRC) is used over both Identifier and Data.

3.2. Controller Area Network (CAN)

CAN architecture, its applications for automotive systems, its protocol characteristics and its node and frame structures have been discuss here.

3.2.1. Architecture and Applications

CAN networks are used for more sophisticated applications which require higher data-rates than the LIN bus can provide. CAN is by far the most widely used communication technology in the automotive industry. It was developed in eighties by Bosh and first used in vehicles in 1994; sine then there is increasing trend towards adoption of CAN for vehicular networks [9]. Different ECUs in a vehicle can communicate with each other reliably via an expensive CAN network. A large number of ECUs can be connected to the network using CAN interfaces; this not only decreases wire harness but also weight in the vehicle. In practice, $3-10$ nodes are used per CAN bus.

CAN data is transmitted in the form of frames which are labeled by identifiers to specify the frame priority on the bus. Two versions of the CAN exist based on the size of the identifier: standard CAN and extended CAN. Standard CAN having a 11 bit identifier is considered to be the base frame format. Due to limitations and problems of base frame format for some applications, extended CAN having a 29 bit identifier is specified. All the nodes on the network have a CAN controller that intelligently transmits and receives messages. As the CAN network is message broadcast system, all the nodes see the messages on the bus but only the relevant messages are filtered and received by each node.

Three different flavors of CAN are being used in the vehicles:

1. High-speed CAN: It is the most popular and extensively used version of the CAN due to the fastest speed and a simple cable connection. It can provide a data-rate of about 1 Mbps. HS CAN is mainly used for real-time control in powertrain and chassis domains (500 Kbps) of the vehicle.
2. Fault-tolerant CAN: If fault-tolerant bus is used, CAN communication does not interrupt if wiring failure occurs due to open, short or incorrect load on bus lines. On the other hand, data-rate on this bus limits (125 Kbps) due to its ability of fault-tolerance. FT CAN is used for electronics in the body domain (125 Kbps).
3. Single-wire CAN: It is primarily used to decrease the cost as only a single-ended CAN data wire is required. However, it is not immune to noise and due to this only a limited data-rate ($33.3-83.3$ Kbps) is achieved; used in body and comfort devices.

3.2.2. Protocol Characteristics

CAN protocol allow nodes on the network to start transmitting if the bus is idle. A process called bus arbitration is used to avoid conflicts if multiple nodes start transmitting at the same time. Arbitration process is based on the priority of the frames, which is transmitted in the arbitration field or identifier of the frame.

3.2.3. CAN Node Structure

The structure of a CAN node consists of three layers: Application, Object, Transfer and Physical. The interface with the Application layer, i.e. the application running on the hardware, is provided by the Object layer. Moreover, the Object layer mainly performs message filtering and processing. The processing of messages involves determining the messages to be used among the ones received from the lower layer. The Transfer layer provides a large number of functionalities such as message framing and validation, bus arbitrations, fault confinement, error detection and time management for bus transfer. Physical layer specifies the actual transfer medium properties and electrical signal levels for transmission of bits between nodes.

3.2.4. CAN Frame Structure

The frame structure of the CAN protocol contains the following fields:

1. Start of Frame: A single dominant bit is transmitted in this field to start an exchange process.
2. Arbitration: It consists of 11 bits of an Identifier and an RTR bit; Identifier bits are transmitted in reverse order and RTR bit is dominant.
3. Control: 6 bits of this field represent Reserved bits (2 bits for extended CAN frame) and Data Length (4 bits to indicate data bytes).
4. Data**:** It can contain total 8 bytes of Data, i.e. useful information to be transmitted.
5. CRC**:** 15 bits are used in this field for error detection and correction purposes.
6. Acknowledgement**:** Uses a single bit each for ACK (Acknowledgement) slot and ACK delimiter subfields. ACK slot is used to acknowledge the successful reception of the message frame.
7. End of Frame: Used to indicate the end of frame utilizing a fixed sequence.

3.3. FlexRay

Flexray architecture, its applications for automotive systems, its protocol characteristics and its node and frame structures have been discussed here.

3.3.1. Architecture and Applications

Flexray is usually referred to as the next generation vehicular network technology. It not only possesses the capability of high speed data transfer but also higher degree fault-tolerance and reliability [1]. In order to achieve an open bus standard for time-triggered distributed control of safety-critical systems in the vehicles, a consortium of automobile companies was formed in 2000. In 2005, Flexray consortium released protocol and physical layer specifications and it was introduced first time in the BMW X5 in 2006 [10]. Since then it was used in many vehicles and today it may be considered as a de-facto standard for fault-tolerant and deterministic high speed vehicular networks [11].

Flexray introduced a flexible, hybrid and a dual channel communication network for automotive systems which is very different from the priority driven approach used by CAN network [2]. Flexray network has been designed to transmit information supporting two channels and two approaches; event triggered approach and time triggered approach. In event triggered approach, a message has to be transmitted before the occurrence of any event while in time triggered approach, a message has to be transmitted before specific time. Event triggered approach is used for diagnosing any fault in vehicle while time triggered approach is used for in-vehicle communication. Flexray protocol has been developed to communicate at the data rate of 10Mbits/s per channel or total of 20Mbits/s per two channels which is greater than 20 times of CAN network data rate [12].

Flexray supports three network topologies; passive bus topology, active star topology and active star topology combined with passive bus topology [12]. These are discussed below.

1. Passive bus topology: The configuration of the network in this topology is like a dual bus which means that a node can be connected to both channels (A and B) and single channel (A or B) individually. It can also be a single bus communication network depending upon it is connected to single channel only.
2. Active star topology: This topology is a multiple star topology which means that it can support redundant or multiple channels as well as single channel. When we create a single, non-redundant star topology it is same as we created a single bus communication network in passive bus topology. Similarly, when we create a single but redundant star topology it is same as we created a dual bus communication network in passive bus topology. But when we built a communication network with cascaded stars then each node must have a p2p connection with one of the two stars while the stars are connected to each other directly. We can also build a network with redundant cascaded stars in which a node can be connected to one or more stars.
3. Active star topology combined with passive bus topology: This topology is hybrid of passive bus and active star topologies which means that for communication, nodes can build a point to point connection, a bus topology connection and a star coupler connection at a time.

3.3.2. Protocol Characteristics

Flexray utilizes recurring communication cycles that consists of static, dynamic segments, symbol window and network idle time to implement media access control [11]. The static and dynamic segments arbitrate transmissions on the bus using time-division multiple access and mini-slotting based multiple access schemes which allow nodes to transmit data statically and dynamically respectively. Moreover, the arbitration grids that form the backbone of FlexRay media arbitration are used by each segment at the lower level. The arbitration grids of static segments are called static slots and of the dynamic segments are called minislots. Each static slot has the same duration and a specific message is assigned to a specific time slot while a minislot has to come around to that node which wants to communicate and if minislot does not come around it has to wait for it. If no communication takes place after a specific time then minislot counter has to incremented and then next message has to be transmitted. Next to arbitration grid level, there are macrotick and microtick levels. Boundaries of macroticks are action points which tells that transmission start from here and end at here. Macrotick and microtick are actually timers which counts microticks and macroticks respectively [13].

3.3.3. FlexRay Node Structure

Node is responsible for sending and receiving frames or messages. As Flexray is dual channel network so in flexray a node must have two bus drivers for each channel, a communication controller, a host and a power supply unit [12]. Bus guardians are used for each channel for monitoring the improper transmissions with respect to time. There are different interfaces in a node architecture i.e. host-communication controller interface (H-CC), host-bus guardian interface (H-BG), communication controller-bus guardian interface (CC-BG), communication controller-bus driver interface (CC-BD), bus driver-host interface (BD-H), bus driver-bus guardian interface (BD-BG) and last one, which is optional, bus driver-power supply interface (BD-PS). All interfaces handle different applications. H-CC and H-BG share a lot of information within the interface. Host shares configuration data information with CC and BG while CC and BG share status information with host. CC-BG interface allows BG to supervise all the transmissions of the CC. In CC-BD interface, CC shares transmit data (TxD) and transmit data enable not (TxEN) information with bus driver and BD shares receive data (RxD) information with CC. In BD-H interface, BD shares status information and read error conditions with host and host controls the operation modes of the BD. In BD-BG interface, BD shares receive data enable (RxEN) information with BG and BG shares bus guardian enable (BGE) information with BD. In BD-PS interface, BD shares inhibit signal (INH) with PS which can be used for controlling the modes of ECU.

3.3.4. FlexRay Frame Structure

In Flexray, data is transmitted in the form of frames. Each frame has 3 segments; header segment, payload segment and trailer segment [1, 14]. Header segment consists of frame ID, payload length, header CRC and cycle count [15]. Header segment is used for protecting length in Flexray while CAN doesn't have this. Each frame ID, which is unique for each channel consists of 11 bits, is used to check the priority of any frame in communication cycle. Each higher frame identifier represents low priority. In header CRC, HD=6 is used for detecting error in header data. Each payload segment consists of 0 – 254 byte of data which is a static segment and contains the whole data while in trailer segment 24 bits are reserved for Cyclic Redundancy Check (CRC) which it checks over header and payload segments [14].

3.4. Media Oriented Systems Transport (MOST)

MOST technology can be referred to as a de-facto standard for multimedia applications in vehicular networks [16]. A partnership of automotive companies called MOST Cooperation, founded in 1998, defined this common protocol for high data-rate multimedia applications. The specifications provide standard application program interfaces (APIs) to access and integrate with different multimedia devices. Physical layer of the MOST is usually implemented using plastic optical fibers (POF) due to its unique advantages such as flexibility, lighter weight and no interference radiation. Though other network topologies are possible, MOST devices are mostly connected in a ring topology. MOST provides connectivity among typical multimedia devices and applications like audio-video transfer, GPS navigation and video displays. MOST protocol has been developed to communicate at the data rate of 24.8Mbits/s and this is a single channel communication network [16].

3.4.1. MOST Node Structure

As mentioned earlier MOST is a unique topology whose nodes form a ring shape which means that all preceding nodes are forwarding information or data to the next nodes in a cycle [17]. There are some special nodes in the MOST which manage management or synchronization of ring [17]. When any node comes in a ring then some functional address is assigned to that node. There are three interfaces in a node. Human Machine interface which is responsible for user's communication with MOST, program user interface which are controllers for other devices and slaves or actuators which are responsible for responding functional block's commands.

3.4.2. MOST Frame Structure

In MOST, data is transmitted in the form of blocks. Each block has 16 frames and each frame has 512 bits of data [15]. In a frame 4 bits are reserved for preamble, 4 bits are for boundary descriptor, then 480 bits are for synchronous or asynchronous data, 16 bits are for control frame, 7 bits are for frame control and status information and last bit is parity bit for error detection [18].

Table 2. *Comparison of wireless technologies.*

Technology	Blue tooth	ZigBee	Wi-Fi	UWB
Data rate	1-3 Mbps	25-250 Kbps	54 Mbps	27.24 Mbps
Bandwidth	2.4 GHz only	2.4 GHz, 915 MHz, and 868 MHz	2.4, 3.6 & 5 GHz	7.5 GHz

3.5. Wireless Technologies

All the networks discussed above are wired; there are also many wireless technologies for both in-vehicle and vehicle-to-vehicle (V2V) communication. These wireless technologies are Bluetooth, ZigBee, Wi-Fi and Ultra Wide Band (UWB) which can be used successfully in automotive systems for communication [2]. Bluetooth also known as IEEE 802.15.1 standard is actually designed to replace cables in devices such as mouse, keyboard and printer etc. [19]. It can be used for exchanging data for short distances wirelessly with a maximum network speed of 3Mbps. There are two network topologies in Bluetooth; the piconet and the scatternet. In piconet, there are two configurations; one master, one slave and one master, multiple slaves. Each device can serve as master or a slave. In one master and multiple slaves configuration, there can be maximum of seven slave devices. One or more piconets form the scatternet which can support more than 8 devices. ZigBee also known as IEEE 802.15.4 standard is designed for exchanging data for long distances with low data rate but with secure networking. This application is known as low rate wireless personal area network (LR-WPAN) [19, 20]. The band width and data rate comparison of all these wireless technologies is given in Table 2.

A network consists of different nodes, each node has a microcontroller, a transmitter and an antenna which can perform different applications like detection, controlling and monitoring. A node can be either a full function device (FFD) or a reduced function device (RFD). FFD can perform all the tasks which are in ZigBee network while RFD can perform only limited tasks. Wireless fidelity (Wifi) is a technology for wireless local area networking (WLAN) is known as IEEE 802.11 standard [21]. In wi-fi, communication operates in ad hoc mode, this network technology uses less power but provides high bandwidth, and also there is no need to talk to access point before talking to other devices. Wifi is based on cellular architecture in which each cell is a Basic Service Set (BSS) which means that there are collections of stations for communicating to each other within the network. BSS are mainly of two types; independent BSS and infrastructure BSS. In independent BSS, there are no access points and they cannot connect to any other BSS because these are ad hoc networks. While infrastructure BSS can connect to any other BSS through access points. UWB is actually alternative to Bluetooth and is suitable for short distance communication with high speed and its bandwidth is up to 480 Mbps [22]. It can fulfil the multimedia applications like audio and video transmission to multiple channels at a time. The main advantage of UWB is that it has the ability to resist the interference so it can be used for collision avoidance [23].

4. Conclusion

The state-of-the-art wired network technologies for in-vehicle communication such as LIN, CAN, FLEXRAY and MOST and wireless network technologies for in-vehicle and vehicle-to-vehicle (V2V) communication such as Bluetooth, ZigBee, Wi-Fi and UWB have been discussed in this paper and compared in terms of architecture, development, applications, protocol characteristics, node structure, frame structure and data rate. LIN protocol is used extensively for low level communication system with a bandwidth of around 20k Bits/s, CAN protocol fulfils requirements of soft real time systems with a bandwidth of around 500 kBits/s, whereas flexray is an excellent network topology for in-vehicle communication which is hybrid, very flexible and higher degree fault tolerant. Flexray can be used for hard real time systems with a bandwidth of 10Mbits/s per channel while MOST can be used for multimedia and telemetric with a bandwidth of 24.8 Mbits/s. Wireless technologies i.e. Bluetooth and ZigBee are designed to exchange data with low data rates; 1-3Mbps and 25-250kbps respectively while Wi-Fi and UWB are designed for high data rate; 54Mbits/s and 27.24 Mbps respectively. All these technologies satisfy the requirements of diagnostics and multimedia communication for in-vehicle and vehicle to vehicle communication and can be used for advanced autonomous driving systems but the bigger challenge can be to integrate all these protocols in an autonomous vehicle with real time constraints.

References

[1] N. Nicolas, Y. Song, F. Simonot-Lion, C. Wilwert, "Trends in automotive communication systems", Proceedings of the IEEE, vol. 93, no. 6, pp. 1204-1223, 2005.

[2] N. Thomas, H. Hansson, L. Lo Bello, "Automotive communications-past, current and future", In Emerging Technologies and Factory Automation, 2005, 10th IEEE Conference on, vol. 1, pp. 8-pp. 2005.

[3] AUTOSAR, Technical Overview v. 2.2.2.

[4] C. Gianluca, A. Valenzano, and S. Vitturi, "Advances in automotive digital communications." Computer Standards & Interfaces, vol 27, no. 6 pp. 665-678, 2005.

[5] C. Samarjit, M. A. Al Faruque, W. Chang, D. Goswami, M. Wolf, and Q. Zhu. "Automotive cyber–physical systems: A tutorial introduction." IEEE Design & Test 33, no. 4, pp. 92-108, 2016.

[6] S. Maria, M. Simic, and H. Schmidt. "From automotive to autonomous: Time-triggered operating systems." In Intelligent Interactive Multimedia Systems and Services, pp. 347-359. Springer, Cham, 2016.

[7] Novais, P., and S. Konomi. "A survey on vehicular communication technologies." In Intelligent Environments 2016: Workshop Proceedings of the 12th International Conference on Intelligent Environments, vol. 21, p. 308. IOS Press, 2016.

[8] Tuohy, Shane, Martin Glavin, Ciarán Hughes, Edward Jones, Mohan Trivedi, and Liam Kilmartin. "Intra-vehicle networks: A review." IEEE Transactions on Intelligent Transportation Systems 16, no. 2 (2015): 534-545.

[9] K. Seung-Han, S. Seo, J. Kim, T.-M. Moon, C.-W. Son, S.-H. Hwang, and J. W. Jeon. "A gateway system for an automotive system: LIN, CAN, and FlexRay", In Industrial Informatics, 2008. 6th IEEE International Conference on, pp. 967-972. IEEE, 2008.

[10] F. Andreas, and J. Hedberg, "Comparison of flexray and can-bus for real-time communication", IEEE transactions on industrial electronics vol. 58, no. 3, 2012.

[11] S. Yung-Hoh, C.-M. Ku, Z.-W. Wang, "A FlexRay/CAN vehicle communication network for an anti-lock braking system based on X-by-wire technology", In ICCM 2013, 9th International Conference on Computing Technology and Information Management. 2013.

[12] S. Robert, B. Jackman, "An introduction to FlexRay as an industrial network", In Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on, pp. 1849-1854. IEEE, 2008.

[13] H. Florian, T. Wagner, C. Horst, F. Bailer, M. Ihle, "FlexRay communication controller", U.S. Patent 8, 484, 383, issued July 9, 2013.

[14] Li, Wenchao, Marco Di Natale, Wei Zheng, Paolo Giusto, Alberto Sangiovanni-Vincentelli, Sanjit A. Seshia, "Optimizations of an application-level protocol for enhanced dependability in FlexRay", In Proceedings of the Conference on Design, Automation and Test in Europe, pp. 1076-1081. European Design and Automation Association, 2009.

[15] R. B. Abdul, S. Krishnaveni, "Comparison of CAN, TTP and Flexray Communication Protocols‖", International Journal of Innovative Research in Computer and Communication Engineering 2.

[16] K. Ugur. "In-vehicle communication networks: a literature survey", Computer Science Report 10, 2009.

[17] S. Andrzej, M. Buczaj, "The evolution of "Media Oriented Systems Transport protocol" Teka Komisji Motoryzacji i Energetyki Rolnictwa, vol 14, no. 3, 2014.

[18] S. Andrzej, M. Buczaj, "New elements in vehicle communication "Media Oriented Systems Transport" protocol". Teka Komisji Motoryzacji i Energetyki Rolnictwa, vol. 12, no. 1, 2012.

[19] L. Jin-Shyan, Y.-W. Su, C.-C. Shen, "A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi", In Industrial Electronics Society, 2007. 33rd Annual Conference of the IEEE, pp. 46-51, 2007.

[20] B. Nick, "ZigBee and Bluetooth: Strengths and weaknesses for industrial applications", Computing and Control Engineering, vol. 16, no. 2, pp. 20-25, 2005.

[21] F. Erina, F. Potorti, "Bluetooth and Wi-Fi wireless protocols: a survey and a comparison", IEEE Wireless Communications vol. 12, no. 1 pp. 12-26, 2005.

[22] R. Paul, W. Xiang, W. Stark, “Modeling of ultra-wideband channels within vehicles” IEEE Journal on selected areas in communications, vol. 24, no. 4, pp. 906-912, 2006.

[23] Khairnar, Mrs, D. Vaishali, SN Pradhan. “V2V communication survey wireless technology”. arXiv preprint arXiv:1403. 3993, 2014.