A Sophisticated Solution to Automotive Amplifier System Testing over Ethernet

Rishi Divakar

***Abstract* — This paper provides an overview of a sophisticated solution to system testing an automotive amplifier over an Automotive Ethernet interface.**

***Keywords* — Automotive Ethernet; SOME/IP (Scalable Service-Oriented Middleware over IP), AVB (Audio Video Bridging), DoIP (Diagnostics Over IP), PTP (Precision Time Protocol)**

I. INTRODUCTION

A car amplifier is an electronic device that increases the power of a car audio system, improving its sound quality and volume. It takes the low-level signal from the Head-Unit or other source and amplifies it to drive the speakers, producing a stronger and clearer audio output. Amplifiers typically have multiple channels to support various types of speakers (such as subwoofers and tweeters) and may feature tone controls and other features to enhance sound quality. Some car amplifiers can be integrated into the car's sound system, while others can be added as aftermarket upgrades.

In-Car audio has been part of the automotive industry since the car went into mass production. Car audio systems have moved from traditional mono systems to stereo systems and surround-sound solutions. The initial approach was following the Home Audio System with a central amplifier powering speakers within the car. Digital car amplifiers, on the other hand, use digital signal processing (DSP) to process and amplify the audio signal. They are more efficient, producing less heat, and they can be smaller in size. They also offer more control over the audio signal, with features such as adjustable crossover points, equalization, and time correction. They can also be programmed to deliver specific audio processing options, making them more versatile than analog amplifiers.

Automotive Ethernet is a type of computer networking technology that is used in the automotive industry to connect various electronic components in a vehicle. It is based on the standard Ethernet protocol and allows for high-speed communication between devices, such as cameras, sensors, infotainment systems, and advanced driver-assistance systems (ADAS).

The use of Automotive Ethernet helps to address the increasing need for high-speed data transfer and connectivity in modern vehicles. It supports the growing number of electronic systems in vehicles and allows for the integration of advanced features, such as in-car entertainment, telematics, and autonomous driving.

One of the benefits of Automotive Ethernet is its ability to handle multiple streams of high-bandwidth data, enabling high-quality multimedia entertainment, real-time traffic information, and other services.

Automotive Ethernet supports the Transmission Control Protocol/Internet Protocol (TCP/IP) stack which is a suite of communication protocols used to interconnect network devices and transfer data over a network. It is the most widely used communication protocol suite on the Internet and is the standard for communicating between computers and other devices over the internet.

Support for specific application needs is provided by extensions to the basic Ethernet Internet Protocol (IP) standard by Audio Video Bridging (AVB), Scalable Service-Oriented MiddlewarE over IP (SOME/IP), and Diagnostic Over IP (DoIP). This allows automotive OEMs to move to Ethernet for a variety of applications, using a single networking technology and wiring system. In this paper, we will be covering how an Automotive Ethernet-based amplifier test setup is built.

The proposed Ethernet simulator for building the test system is CANoe and VN5620 hardware – both from Vector, the leading provider of software tools and services for the automotive industry. The VN5620 is a compact and powerful interface facilitating the simulation, testing, and validation of Ethernet networks and its protocols. And CANoe is a comprehensive tool that provides a wide range of features for Ethernet communication, including automotive protocols simulation, data analysis, diagnostics, and customization.

Amplifier testing is not complete without verifying its output by a sound acquisition system. The proposed sound acquisition system is built on National Instruments (NI) platform which is a company that provides hardware and software solutions such as Laboratory Virtual Instrument Engineering Workbench (LabVIEW) for automated test and measurement systems, including data acquisition and control systems.

Problem Statement — Over the past 10 years, automotive applications are growing in an exponential manner which has demanded higher bandwidth requirements. Automotive OEMs are also adding more and more computer-based systems, applications, and connections. Ethernet deployment can and will reduce the cost to setup harness in terms of cabling, network interfaces, and on-board computing power. Recent technology advancements have made Ethernet viable for use in cars.

There are already devices that enable the simulation of SOME/IP, DoIP, and AVB protocols. Available solutions require additional hardware which causes performance issues in Time Sensitive Network (TSN) based amplifier testing. The proposed solution with Vector hardware and customized software CANoe configuration will be able to simulate 3 protocols, especially maintaining the consistent frame rate in AVB streaming with different gain and standard frequency ranges (20-20000 Hz).

II. AMPLIFIER SYSTEM

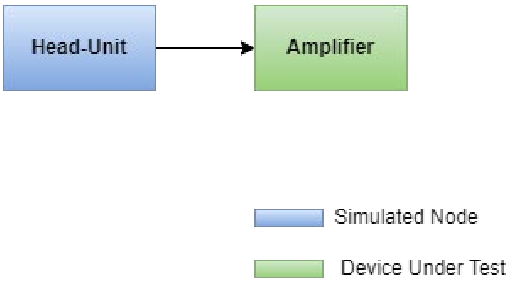

Figure 1: Car Audio System

Complete Audio system for passing the audio in a car network includes a Head-Unit and an amplifier. Head-Unit sources audio stream as AVB packets to the amplifier based on the connected media devices. The amplifier receives the AVB packets over the Ethernet network, post-process, amplifies, and plays the audio over the connected speakers. In order to perform a system testing of the audio amplifier, a simulation setup is needed which is able to stream audio precisely and send control and diagnostics commands to the amplifier via a single Ethernet interface.

The industry is looking for a solution that is capable of streaming audio of standard frequency ranges and different gains. This paper details this question.

The amplifier supports the following:

- 1 AVB Class A Stream
- 32-bit per sample
- 48 kHz Sampling rate
- 32-bit PCM integer stream with 24bit PCM content + zero pad

III. TOOLS & HARDWARE

A. Hardware and Software tools used:

No	Item Description	Version	Purpose
1	Rosenberger No. E6K10A-1xxZ5-y	-	The physical interface to the amplifier
2	ix connector to Rosenberger No. E6K10A-1xxZ5-y conversion	-	Cable with the physical conversion from HIL simulator to amplifier interface
3	Vector VN5620	-	HIL simulator interfacing to Amplifier network (HIL Simulator hardware)
4	NI PXIe-4464	-	Hardware to perform sound data acquisition
5	NI PXIe-1095	-	Medium to insert PXIe-4464 card
6	NI LabVIEW	2021	A software tool to develop sound data acquisition application
7	NI Sound & Vibration Toolkit	-	A software tool to measure and calculate gain, voltage level etc.
8	Vector CANoe	15 PRO	A software tool providing the HIL Ethernet protocol functionality (HIL Simulator tool)

Table 1: Tools & Hardware list

B. Test System Overview

HIL simulator simulates a traditional Head-Unit which sends SOME/IP, and DoIP commands and stream AVB packets to amplifier. The proposed HIL-Simulator is using Vector hardware VN5620 with Automotive Ethernet capabilities and Vector CANoe 15 PRO tool to design and develop the simulation. HIL Simulator uses Communication Access Programming Language (CAPL) scripts to interact with the Ethernet network and perform specific tasks such as simulation of specific automotive protocols.

Measurement setup records the sound data from speaker output and performs basic measurements such as voltage, gain, etc. The proposed measurement setup is using National Instruments-powered PXIe-4464 card with LabVIEW 2021 and Sound & Vibration toolkit.

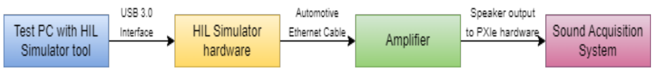

Figure 2: Test System Overview

IV. TEST SYSTEM DESIGN

HIL-Simulator simulates SOME/IP, DoIP, and AVB protocols and passes them to an amplifier and the Sound acquisition application acquires sound output from the amplifier's speaker's side for measurement applications.

To enable SOME/IP, DoIP commands, and AVB streaming to the amplifier via HIL-Simulator, custom scripts must be written in the software tool to design HIL-Simulator which frames the header and payload for respective protocols.

A. arxml File

An AUTOSAR XML file or ARXML for short is just a normal XML file with the file extension ".arxml". The root XML element is called <AUTOSAR>. Inside the AUTOSAR (XML), there will be a collection of AUTOSAR packages each containing elements. I.e. arxml file shall be a Data Base (DB) file for HIL-Simulation system.

B. SOME/IP Simulation

Message ID (Service ID/Method ID) (32 bit)			
Length (32 bit)			
Request ID (Client ID/Session ID) (32 bit)			
Protocol version (8 bit)	Interface version (8 bit)	Message Type (8 bit)	Return Code (8 bit)
Flags (8 bit)	Reserved (24 bit)		
Length of Entries Array (32 bit)			
Entries Array			
Length of Options Array (32 bit)			
Options Array			

Figure 3: SOME/IP Service Discovery Message Formatting

The SOME/IP Service Discovery messages must be assembled according to the format shown and sent to the amplifier. To do this, the remote peer must be configured with the appropriate properties. The following section lists how to configure a peer for a subscribe by manual steps. However, the peer might also be configured automatically according the arxml if the database node is assigned to a simulation node.

1. *Open a TCP connection from the client's IP address and source port.*
2. *Creating a service instance is based on the TCP connection handle in the previous step. This step enables setting properties such as Major version, Minor version, etc.*
3. *Create an event group for the service instance defined in the previous step*
4. *Add notification method ID's to the event group created in the previous step.*
5. *Send subscription requests to the server. The service subscription request is sent to the server through a UDP port defined by the server's service offer message*

An overview of the connection process is given in Figure 4.

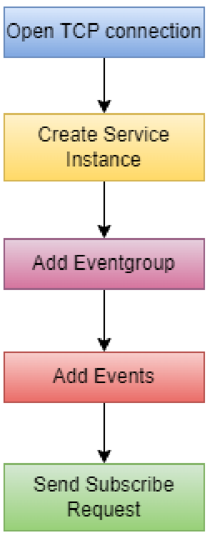

Figure 4: SOME/IP Subscription

Service instance configures SOME/IP header parameters. Based on that, the AUTOSAR Interaction Layer DLL has APIs that shall configure payload data. The message is sent to the server IP address and TCP port.

While framing the payload, the test system developer needs to make sure that the header values and payload bytes are in the expected format. Otherwise, the amplifier may not accept this packet or HIL-Simulator shall show the packet as corrupted.

C. DoIP Simulation

The Unified Diagnostic Service (UDS) protocol (ISO 14229) is standardized across both manufacturers and standards. Further, UDS is today used in amplifiers across all TIER1 Original Equipment Manufacturers (OEMs). In practice, UDS communication is performed in a client-server relationship.

DoIP (Diagnostics over Internet Protocol) facilitates the use of automotive diagnostic services exposed through UDS over TCP/IP on an Ethernet network. Compared to conventional CAN-based diagnostics, DoIP allows for much faster data transfer rates at low hardware costs. This makes DoIP interesting for today's car manufacturers. In short, DoIP is a wrapper around the UDS protocol that transfers diagnostic information through the TCP protocol.

The Open Diagnostic eXchange format (ODX or PDX - packed ODX) is an XML-based ASAM standard which specifies data model, communication parameters and diagnostic capabilities of ECUs. The diagnostic description is used throughout development, testing and production process as well as for after-sales and service.

PDX file is added in the HIL-Simulator tool to enable DoIP communication. Once the file is loaded, the validation engineer needs to configure the interface, variant, diagnostic tester, and simulation by options to enable the TCP/IP communication.

DoIP connection using the HIL-Simulator tool is processed in the below steps:

1. *Load the PDX file in HIL-Simulator*
2. *Open TCP connection between Server and Client*
3. *TCP 3-way handshake shall be performed*
4. *Send Vehicle Identification request from the client to the server: A UDP message with no payload shall be sent to the DoIP server. This is to identify which node is taking part in DoIP communication.*
5. *Receive Vehicle Identification response: A UDP message with no payload shall be received from the server.*
6. *Send Routing activation request: A TCP message with payload type 0x0005 shall be sent from the client to the server.*
7. *Receive Routing activation response: A TCP message with payload type 0x0006 shall be sent from the server to the client.*
8. *DoIP connection is established once the above steps are completed. A bi-directional communication between server and client where payload type is 0x8001 shall be transferred, is established.*

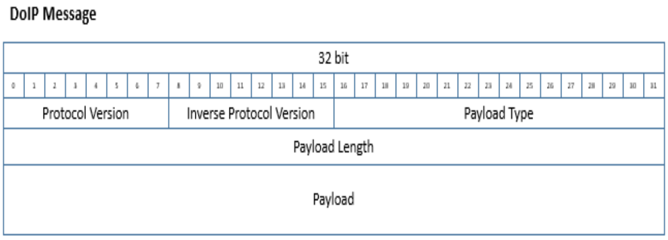

Figure 5: DoIP Message Format

Above steps are summarized as a flowchart in Figure 6.

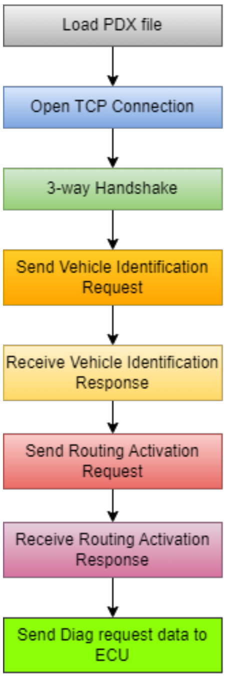

Figure 6: DoIP Connection Process

The Payload of the DoIP Message shall have the first 4 bytes as the logical source address and logical amplifier address. TCP connections must be reconnected quickly as soon as they are disconnected due to a battery cycle or Hard Reset service.

The connection process is handled by HIL-Simulator where the amplifier shall act as a server in DoIP communication. However, if a scenario arises where the amplifier is configured as a server, a CAPL script needs to be developed which shall initiate a server socket at the HIL-Simulator side and accept the TCP connection request from the amplifier. Followed by sending the DoIP header and UDS payload as TCP packet. UDS payload can be captured directly from the Diagnostic console by adding an event filter in the measurement setup and store the event data to the HIL-Simulator software tool's system variables so that data can be added directly into the execution report.

D. Precision Time Protocol

Precision Time Protocol as defined in IEEE 1588 provides a method to precisely synchronize computers connected to a Local Area Network (LAN). PTP is capable of synchronizing the multiple clocks to the nano second scale in a network specifically designed for IEEE 1588.

The protocol defines the synchronization messages used between Master and Slave clocks. The slave has to synchronize to the master clock eventually. The messages in the protocol include master sync message, master delay response message and slave clock delay request messages. The Best Master Clock (BMC) algorithm allows multiple masters to negotiate the best clock for the network.

IEEE 1588 protocol does not define how exactly to implement PTP into a master or slave. While taking into consideration of Hardware-in-the-Loop development, a developer does not need to worry about the implementation of this protocol since the simulator can configure itself as either a master or a slave and clock synchronization would be carried out by clocks in simulator and amplifier.

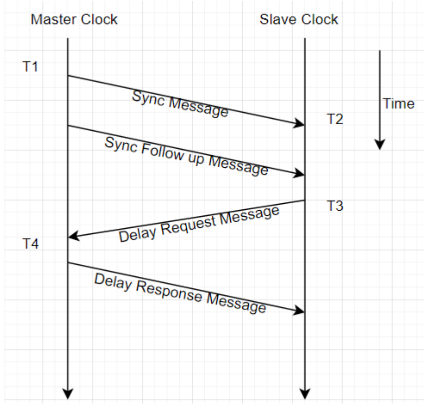

Figure 7: PTP Sync Cycles (simplified)

Figure 7 explains how time synchronization happens between a Master and a Slave.

1. *Four timestamps are captured between master and slave clocks.*
2. *A sync message with precise timestamp is sent from master to slave clocks.*
3. *This timestamp is sent in the follow-up message since the time T1 was sampled when sync message was transmitted.*
4. *T2 is the precise timestamp of the sync message as it is received at the slave.*
5. *Master to slave difference = T2' - T1 (with T2' being T2 expressed in units of the master clock)*
6. *T3 is the precise time of the delay request message from the slave and T4 is the precise time of delay request message when received at the master.*
7. *Slave to master difference = T4 – T3' (with T3' being T3 expressed in units of the master clock)*
8. *One way delay = (master to slave difference + slave to master difference)/2*
9. *The offset is used to correct the slave clock. i.e. Offset = ((T2' – T1) + (T4 – T3'))/2*

E. AVB Streaming Simulation

Audio Video Bridging (AVB) is a set of standards for audio and video transmission over Ethernet networks. It provides a low-latency, deterministic, and reliable method for transmitting audio and video data over Ethernet networks. AVB is designed to meet the demanding requirements of professional audio and video applications, such as live performance and broadcasting.

AVB implements time-sensitive networking (TSN), which ensures that audio and video data is transmitted with a predictable and consistent delay. TSN enables multiple audio and video streams to be transmitted over a single Ethernet network without interfering with each other. AVB also includes a control protocol called the Stream Reservation Protocol (SRP), which allows devices to reserve bandwidth for audio and video streams, ensuring that the required bandwidth is available when it is needed.

AVB has been adopted by several organizations, including the Audio Engineering Society (AES), the European Broadcasting Union (EBU), and the Institute of Electrical and Electronics Engineers (IEEE). It is becoming increasingly popular for professional audio and video applications and is supported by a growing number of hardware and software vendors.

HIL-Simulator shall simulate a sine wave in the different frequency ranges (20-20000 Hz). Before writing a CAPL script for AVB streaming, the developer must know the following parameters of AVB streaming. These parameters are critical such that, if any of those parameters are incorrectly configured in the CAPL script, the amplifier shall not be able to decode the AVB stream packets and no audio shall be produced at the speaker end.

1. *Audio Samples per block*
2. *Sampling Rate*
3. *Bit Depth*
4. *Stream ID*
5. *Source MAC Address*
6. *Destination MAC Address*
7. *VLAN (if any)*
8. *AVTP Subtype*
9. *Channels per Frame*
10. *Stream Data length.*

AVTP (AVB Transport Protocol) uses a MAC-to-MAC sending approach. i.e. no IP addresses or ports are required for streaming AVB packets. HIL-Simulator software tool allows setting the above parameters. Also, an option to render the audio input from the PC can be configured in the script so that any audio that plays on Test PC can be played in the amplifier speaker end.

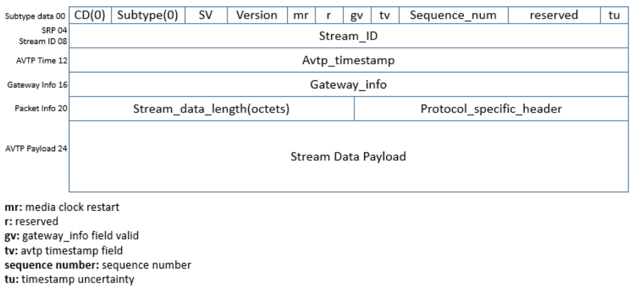

Figure 8: AVTP Message Format

An overview of the AVB connection process is given in Figure 9.

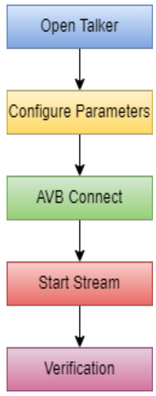

Figure 9: AVB Connection Process

1. *Open Talker: AVB_IL.dll enables to open of a talker/streaming end MAC address from which AVB packets are streamed.*
2. *Configure Parameters: Certain parameters need to be set in the AVB streaming packets using MediaSet/AVBSet functions so that AVB packets can be decoded at the receiving end MAC address.*
3. *AVB connect: AVB connect function shall enable a connection between MAC addresses.*
4. *Start Stream: Sine generation payloads are transferred to the AVB payload and AVB packets shall be streamed to the device at the other endpoint.*
5. *Verification: AVB packets can be performed by either verifying AVB packets in the HIL-Simulator software tool's trace or by connecting speakers to amplifier.*

F. Sound Data Acquisition

Sound data acquisition is the process of collecting and recording sound data from a specific source, such as a microphone, instrument, or machine. The process involves capturing the analog sound waves and converting them into digital form, which can then be stored, processed, and analyzed. The quality of sound data acquisition depends on various factors, such as the type of microphone or instrument used, the environment in which the sound is recorded, and the hardware and software used for the data acquisition process.

Sound output from an amplifier can be verified physically by connecting appropriate speakers. However, amplifier verification is not complete without measurement of gain, signal-to-noise ratio, voltage level, etc. The proposed sound data acquisition system is using National Instruments (NI) powered PXIe-4464 cards along with NI LabVIEW (2021 version) and Sound & Vibration toolkit.

The sound acquisition system is able to acquire the sound output from the amplifier's speaker's end and perform measurements such as gain, voltage Signal Noise Ratio (SNR), etc. Also, the system has the capability to store the measurement file with Technical Data Management Streaming (TDMS) extension.

V. IMPLEMENTATION AND RESULTS

The proposed HIL-Simulator is built on Vector's VN5620 hardware and CANoe 15 PRO. HIL-Simulation is achieved by writing custom CAPL scripts. The built-in DLL for Automotive Ethernet simulation makes it easier for early deployment. Being a solution for 3 different protocol simulations through a single hardware interface, the current approach is scalable across different modules since the basic simulation concept remains the same.

Compared to the available AVB streaming Automotive Ethernet development board, the proposed solution shall maintain the expected frame rate throughout the CANoe measurement. I.e., no noise will be measured on the acquisition side. The below graph depicts the expected results.

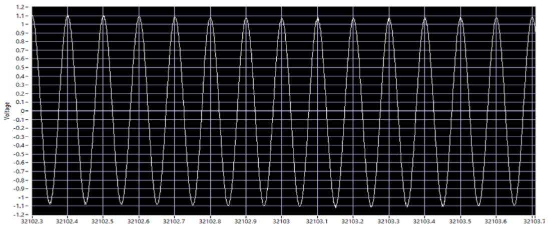

Figure 10: Sound output measured at PXIe-4464 where the source is the CANoe tool.

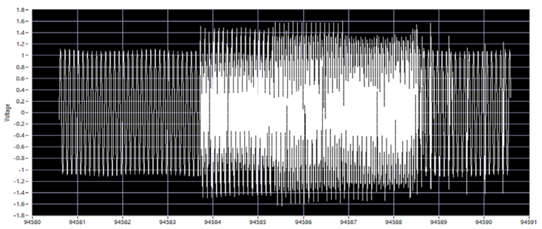

Figure 11: Sound output measured at PXIe-4464 where the source is other mediums of AVB generation.

Since the frame rate is consistent throughout CANoe measurement, it does not produce any noise in any of the frequency ranges. Figure 10 shows the sound acquisition graph where no noise is observed and where the source of AVB streaming is customized CAPL script via the CANoe tool. However, if the frame rate is not consistent, it will cause a small glitch in the speaker output as shown in Figure 11.

VI. CONCLUSION

In this study, a sophisticated solution for automotive amplifier testing over Ethernet is presented. The proposed HIL-Simulator solution using the Vector CANoe tool with VN5620 hardware has been found easily scalable and provides high throughput compared to other available solutions. 4 different protocols (SOME/IP, DoIP, AVB, PTP) simulation is achieved through a single VN interface.

ACKNOWLEDGMENT

The author would like to thank Vector Informatik GmbH for providing timely support while building the test system as well their insightful comments and intriguing discussions on Automotive Ethernet future use cases.

REFERENCES

[1] F. L. Soares, D. R. Campelo, Y. Yan, S. Ruepp, L. Dittmann and L. Ellegard, "Reliability in automotive ethernet networks," *2015 11th International Conference on the Design of Reliable Communication Networks (DRCN)*, Kansas City, MO, USA, 2015, pp. 85-86, doi: 10.1109/DRCN.2015.7148990.

[2] *J. -d. Park, B. -m. Cheoun and J. -w. Jeon, "Worst-case analysis of ethernet AVB in automotive system,"* 2015 IEEE International Conference on Information and Automation*, Lijiang, China, 2015, pp.1696-1699, doi: 10.1109/ICInfA.2015.7279559.*

[3] AUTOSAR document, "SOME/IP Protocol Specification"

[4] AUTOSAR document, "Specification of Diagnostics over IP"

[5] Rosenberger Product Catalogue "System for Frequencies up to 20 GHz and Data Transmission up to 56 Gbps H-MTD® Products High-Speed Modular Twisted-Pair Data".

[6] White paper 'Precision Time Protocol' (PTP/IEEE-1588), End Run Technologies