CAN-MM: Multiplexed Message Authentication Code for Controller Area Network Message Authentication in Road Vehicles

CAN-MM: Multiplexed Message Authentication Code for Controller Area Network Message Authentication in Road Vehicles

Franco Oberti, *Member, IEEE*, Alessandro Savino, *Senior Member, IEEE*, Ernesto Sanchez, *Senior Member, IEEE*, Paolo Casasso, Filippo Parisi, and Stefano Di Carlo, *Senior Member, IEEE*

***Abstract*—As the automotive industry adopts more technology, the threat of cyberattacks on vehicles grows. Electronic Control Units (ECUs) operate in a hostile environment, raising safety concerns for drivers and passengers. Initiatives from both industry and government bodies aim to address these risks. The primary communication protocol used in the automotive industry, the standard Controller Area Networks (CANs) protocol, is a target for cybercriminals due to its limitations in ensuring communication integrity. This paper proposes CAN Multiplexed MAC (CAN-MM), using frequency modulation to multiplex Message Authentication Code (MAC) data with standard CAN communication. CAN-MM enables the transmission of MAC payloads at reduced time cost while maintaining backward compatibility with old CAN protocol versions. The solution is also compatible with modern evolutions of the CAN protocol and advanced algorithms resorting to MAC as part of the security infrastructure.**

***Index Terms*—Automotive, CAN-bus, multiplexed MAC, secure CAN network, secure embedded system.**

I. Introduction

MODERN road vehicles, striving for improved comfort, sustainability, environmental friendliness, and safety [1], feature intricate onboard control systems, especially in real-time safety-critical domains [1], [2]. The increased interconnectivity of electronic components exacerbates this complexity. However, this sophistication also makes the automotive industry an attractive target for attackers [3], with ECUs vulnerable to cyberattacks in hostile environments [4].

Manuscript received 31 October 2023; revised 15 March 2024 and 15 May 2024; accepted 16 May 2024. Date of publication 20 May 2024; date of current version 17 October 2024. This work was supported by the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU through Project SERICS under Grant PE00000014. The review of this article was coordinated by Dr. Gedare Bloom. *(Corresponding author: Stefano Di Carlo.)*

Franco Oberti is with the Department of Control and Computer Engineering, Politecnico di Torino, 10129 Torino, Italy, and also with DUMAREY Softronix S.r.l., 10129 Torino, Italy (e-mail: franco.oberti@dumarey.com).

Alessandro Savino, Ernesto Sanchez, and Stefano Di Carlo are with the Department of Control and Computer Engineering, Politecnico di Torino, 10129 Torino, Italy (e-mail: alessandro.savino@polito.it; ernesto.sanchez@polito.it; stefano.dicarlo@polito.it).

Paolo Casasso and Filippo Parisi are with the DUMAREY Softronix S.r.l., 10129 Torino, Italy (e-mail: paolo.casasso@dumarey.com; filippo.parisi@dumarey.com).

Digital Object Identifier 10.1109/TVT.2024.3402986

To mitigate these risks, carmakers and governments are endorsing initiatives to bolster cybersecurity in the automotive sector (i.e., the ISO/SAE 21434:2021 standard for road vehicles cybersecurity engineering [5], and the ISO/PAS 5112:2022 guidelines for auditing cybersecurity engineering [6]). Additionally, the UN Economic Commission for Europe (UNECE) has introduced new regulations for vehicle cybersecurity and software updates, delivered through the WP.29 package [7], [8]. The automotive industry is working harder to make their products more secure and to research ways to address serious security threats that take advantage of communication between modules [9], [10], [11].

The CAN protocol is central to automotive communication. Therefore, ensuring robust security measures within CAN communication is crucial to uphold the integrity and safety of modern vehicles [12]. Detailed insights into potential CAN threats and related countermeasures are provided in [13], [14], [15]. Country-specific regulations mandate specific CAN messages accessible through an On-Board Diagnostics (OBD) port in every vehicle [16], [17]. Ensuring the integrity (i.e., immunity to tampering) and authenticity (i.e., originating from an authorized source) of CAN messages is, therefore, critical to prevent unauthorized access and ensure the safety and operational efficiency of essential functionalities of the vehicle [18], [19], [20]. To achieve this, the Secure Onboard Communication (SecOC) and Crypto Stack defined in Automotive Open System Architecture (AUTOSAR) require the incorporation of a MAC digest within the payload of each data frame [21]. However, integrating a MAC digest in a CAN frame presents compatibility issues, feasible only for specific CAN protocol versions and resulting in back-compatibility challenges [22].

This paper proposes a technique named CAN-MM, offering a novel approach to MAC transmission. This technique enables the multiplexing of the MAC alongside data transmission without altering the original frame format, ensuring full compatibility with all versions of the standard CAN protocol. The main objective of CAN-MM technology is to integrate a System-on-Chip (SoC) compatible MAC in the CAN version 2.0 to enable achieving a security level that matches the most recent advancements, such as SecOC utilizing MAC with Controller Area Network Flexible Data-Rate (CAN FD). Moreover, this approach addresses

the authentication timing challenges identified by Ikumapayi et al. [22]. Eventually, by freeing data bytes from the CAN frame, it offers a novel approach to incorporate the MAC in signature schemas, authentication protocols, or key exchange mechanisms, such as [23].

The article is organized as follows: Section II gives some background on the CAN network, including vulnerabilities and common attacks, while Section III reports the state-of-the-art literature on CAN security. Section Section IV describes the CAN-MM architecture. Section VI provides experimental results, and Section IX summarizes the main contributions and concludes the paper.

II. Background

On-board ECUs play a crucial role in automotive applications by managing subsystems and facilitating real-time communication with sensors and actuators [24]. The CAN bus, a primary vehicle network, adheres to safety guidelines, ensuring reliable communication in noisy environments. The CAN electrical signal, transmitted differentially through CAN high line (CANH) and CAN low line (CANL), minimizes noise impact from motors, ignition systems, and switching contacts. High-speed (HS) (ISO 11898-2 [25]) and Low-Speed (LS) (ISO 11898-3 [26]) interfaces provide varying throughput capabilities based on different voltage levels. In HS CAN, dominant bit transmission (logic 0) raises CANH to 3.5 V and lowers CANL to 1.5 V, creating a 2 V voltage difference. Recessive bit transmission (logic 1) maintains both CANH and CANL at 2.5 V with minimal voltage difference. A differential voltage above 0.9 V indicates a dominant level (logic 0), while below 0.5 V denotes a recessive level (logic 1), ensuring reliable communication in noisy environments. Twisted-pair conductors are commonly used for physical transmission lines to mitigate magnetic interference.

Multiple CAN protocol variants exist, each supporting different transmission speeds and frame payload sizes. CAN FD and CAN 2.0 protocols differ in maximum transmission speed and payload size, with CAN 2.0 limited to 8 bytes and CAN FD extending to 64 bytes. Despite CAN FD supporting larger payloads, many applications still use 8-byte payloads to ensure compatibility with existing vehicle CAN database [27], [28], [29]. Controller Area Network Extra Long (CAN XL), a newer version meeting ISO/TC 22/SC 31 Data communication standards [30], offers features like extended data payload capacity (up to 2,048 bytes) and higher communication speeds ranging from 500 kbit/s to 5 Mbit/s, with potential speeds reaching 12 Mbit/s in the CAN SIC XL FAST configuration. The CAN SIC XL FAST baud rate is comparable to the 10BASE-T1S technology, also known as Vehicle Ethernet, providing 10 Mbit/s bandwidth over a single-pair physical layer. The original CAN protocol includes no built-in security features. Additionally, country-based regulations require the provision of an OBD port [16], [17], commonly located within vehicles, enabling access to legislative diagnostic messages. These messages, transmitted in plaintext to comply with legislative mandates, introduce considerable security vulnerabilities.

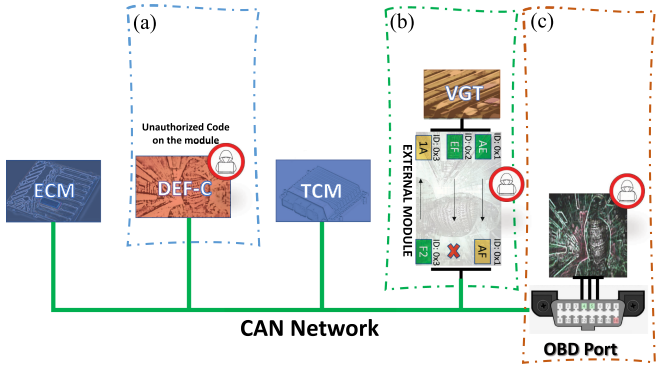

Fig. 1. Vehicle CAN network surface attack scheme. A small CAN vehicle network scheme composed of 4 modules: Engine Control Module (ECM), Transmission Control Module (TCM), Diesel Exhaust Fluid Controller (DEFC), and Variable Geometry Turbine (VGT). These ECUs communicate with sensors and actuators in real-time, making integration essential for their operation. (a) Corrupted vehicle CAN node runs unauthorized code. (b) Attack vector through external CAN module plugged upstream to CAN victim node. (c) The external CAN module directly accesses the OBD port inside the vehicle cabin.

In an endeavor to mitigate these risks, the SecOC framework, explicitly designed for CAN FD, along with CAN secure (CANsec) for CAN XL, has been promulgated. These methodologies expressly elevate the principles of data integrity and authenticity over confidentiality [31]. The critical role played by the CAN bus in the domain of automotive communications mandates a comprehensive investigation into its security weaknesses, potential avenues for attack, and the methods by which such attacks may be carried out [32], [33], [34].

The attack surface of a CAN presents numerous potential vulnerabilities attackers could exploit. This encompasses strategies for unauthorized access, undermining data integrity, data breaches, executing hijacking maneuvers, or hindering the system. Despite the variety of attack vectors against CAN networks, two main types of attacks have been reported in the literature: (i) Man in the Middle (MitM) [35] and (ii) Replay Attacks [36].

Fig. 1 illustrates three prevalent automotive attack settings that target the CAN protocol. Each setting is effectively utilized in MitM and Replay Attacks. Fig. 1(a) demonstrates an attack through a compromised CAN node, where unauthorized software takes control. This can occur via the corruption of the CAN controller's firmware or by exploiting software module vulnerabilities, such as a buffer overflow. In Fig. 1(b), an attack is facilitated by a hardware module that isolates the victim node from the rest of the vehicle network, enabling the interception and manipulation of CAN traffic. The final scheme, depicted in Fig. 1(c), involves connecting an external module to the vehicle's OBD port, granting direct access to the CAN bus. Various commercially available, low-cost CAN modules that feature Bluetooth connectivity support this approach, allowing for programmability via mobile applications. These settings are crucial in laying the groundwork for advanced CAN attacks, exemplified by the Janus Attack [37] and the Cloak Attack [38].

The Janus Attack, a new and sophisticated threat in CAN protocol [37], leverages the CAN protocol synchronization rules and targets devices with different sample points. It involves

transmitting a single CAN frame with dual payloads, causing targeted devices to interpret divergent data compared to others in the network. This undermines the atomic multicast principle of CAN, critical for system integrity. It operates by coercing all CAN controllers to synchronize simultaneously, then manipulating the CAN bus level after the first one has sampled the bus but before another does, resulting in valid frames with differing payloads as it exploits the characteristics of the two different payloads to have the same size.

A cloak attack in cybersecurity involves manipulating bit signals to deceive networked ECUs [38]. The main idea is that the attacker leverages the different sampling times of two receivers to craft two different frames (FrameA and FrameB). The difference is represented by a selection of bits the attacker alters after the first receiver samples the frame (FrameA). Appropriately crafted, the bit-changes in the second frame (FrameB) can avoid triggering re-synchronization mechanisms, aiming for an optimized bit-string with minimal detection and errors in the Cyclic Redundancy Check (CRC) field (as the CRC code will be based on the original content of FrameA). If the attacker achieves such duplication, it can generate out-of-sync data in ECUs.

The Replay Attack shares similarities with MitM attack. To execute this attack, the attacker must perform a learning phase by monitoring the network and collecting a certain amount of CAN frames. Later, the attacker replays these previously collected frames on the network to achieve a target behavior. Unfortunately, this attack does not require the attacker to possess specific skills, expertise, or advanced knowledge about vehicle CAN networks.

These clusters of attacks can be successfully mitigated by linking a CAN Frame payload to a unique MAC that is directly derived from the frame data. Yet, the MAC alone is insufficient for replay attacks due to the CAN payload with identical data producing the same digest. Hence, adopting a rolling counter tied to the data is advised to achieve different digests while maintaining data parity.

The MAC effectively mitigates threats but may also introduce weaknesses in the framework system. This is especially significant in safety-critical, hard real-time systems like ECM, TCM, etc. Ikumapayiet al. formalizes the impact that authentication schemes have on the real-time performance of messages over CAN, CAN FD, and CAN XL based on response time analysis. A CAN frame is schedulable if its Worst-Case Response Time (WCRT) is less than or equal to its deadline. Message deadlines may be implicit, i.e., equal to their period, or explicit (constrained). In particular, Ikumapayi et al. [22] demonstrated that adding a MAC to the payload of CAN, CAN FD, and CAN XL messages might impact the schedulability and the meeting of deadlines based on the percentage of utilization. In particular, on classical CAN, after 70% of utilization, almost all messages fail to meet the deadlines. On the other hand, CAN FD and CAN XL exhibit higher schedulable resilience (it drops when the percentage of bus utilization rises to 80–90%) thanks to the faster bit rate. Nevertheless, pushing such high bus utilization can be malevolent.

When the CAN frames include the MAC in their payload before utilizing the data, the MAC shall be verified as a success. Modern ECUs are generally equipped with a Hardware Secure Module (HSM), a dedicated SoC module that manages all cryptographic and security functions, including verifying MACs. The host system is momentarily suspended during the verification process by the HSM. In the context of real-time systems, an attacker might take advantage of this by injecting or flooding the CAN vehicle network with secure CAN frames that possess a legitimate ID but include counterfeit data and MAC. This situation leads to the HSM being overwhelmed with MAC verification requests that fail, while the host system is forced into repeated waiting periods, causing abnormal delays [39]. These delays can significantly disrupt the system's capacity to adhere to its real-time deadlines, necessitating the initiation of safety system recoveries to address the failure to meet these critical timing constraints.

III. Related Works

As the original version of CAN protocol did not include any security support, researchers have come a long way to support it on top of the existing infrastructure or by proposing enhanced versions.

First attempts to improve the security of the CAN protocol and improve resistance to attacks involved including a MAC digest for integrity and authenticity assurance [40], often employing Cipher-based Message Authentication Code (CMAC) or keyed-Hash Message Authentication Code (HMAC) signatures, depending on hardware support. The CAN+ protocol, introduced by Ziermann et al. in [41], aimed to enhance CAN data rates by relaxing constraints during specific transmission time slots. While the CAN application can benefit from the increased speed, its assessment lacked consideration for Electromagnetic Compatibility (EMC) and disturbance handling, which is crucial in the automotive domain. Furthermore, CAN+ relies on media access characteristics not present in the latest CAN FD and CAN XL protocols, which offer higher payload sizes and data rates. Despite advancements, minimizing latency in MAC signature reception and checking remains essential in CAN FD and CAN XL, which offer increased payload size and data rates.

Significant advancements have been made to enhance broadcast authentication mechanisms, capitalizing on the increased data rate of CAN+. Van Herreveg et al. introduced CanAuth [42], a backward-compatible broadcast message authentication protocol for the CAN bus. This protocol meticulously follows CAN specifications, prioritizing ID-oriented authentication while addressing authentication delays and time synchronization concerns. However, Groza et al. [23] point out that CanAuth's drawback lies in managing many keys associated with message IDs, raising security concerns. In response, they propose the LIBrA-CAN protocol as an alternative. Both LiBrA-CAN and CanAuth share the goal of enhancing CAN communication security but adopt distinct approaches and mechanisms. LiBrA-CAN emphasizes decentralized broadcast-based arbitration and lightweight implementation, ensuring resilience against replay attacks and flexibility in configuration. On the other hand, CanAuth focuses on message authentication and verification, providing robust protection against unauthorized access and

tampering. To preserve the integrity of the physical layer, Hazem et al. [43] put forth LCAP, a Lightweight CAN Authentication Protocol for Securing In-Vehicle Networks.

All previous works point out that the MAC size can significantly impact the resistance to attacks, i.e., the MAC size and the time required to elaborate it. To tackle the time constraints, authors in [44] proposed a truncated MAC, justified by the average data size of 15,768 messages from a 2010 Toyota Prius during a 12.27-minute use case. They noted that only a part of the 8 bytes available in the CAN frames were used, making room for a short MAC. Following a similar direction, to further reduce the schema complexity and support all possible CAN protocols, very recently, Luo et al. [45] proposed a lightweight schema based on the introduction of the MAC in place of the CRC field in the 2.0 version of the protocol. While the authors demonstrated the capability of their approach, the back compatibility with standard hardware is not guaranteed, as they will check a CRC value that is not correct.

In general, both approaches go against National Institute of Standards and Technology (NIST) guidelines, stating that a truncated MAC digest below 4 bytes compromises cyber resilience [46]. Ikumapayi et al. [22] have explored the impact of adding authentication codes as separate messages, noting potential strain on timely delivery, especially given size constraints. As the authors noted, the effect of reserving more than four bytes in CAN 2.0 (i.e., 24Bit-CMAC-8Bit-FV) limits data interchangeability as it requires adding an extra frame to contain the remaining bytes that do not fit into the original frame. However, secure CAN FD and CAN XL protocols support MAC digest sizes from 4 to 16 bytes, accommodating complex protocols like authentications as demonstrated by [23]. Yet, upgrading an entire vehicle network to these protocols involves benefits and extra costs [47], which are left to the manufacturer to evaluate.

Eventually, it is worth mentioning that some recent works support authentication and confidentiality without resorting to MAC [48]. They include only cryptography techniques in the handshake phase, leading to a tiny increase in the latency, limited to hundreds of μs, paying with reduced security if compared with schemas resorting to MAC [23], [49], [50].

Modulation techniques are not new in the security of the CAN protocol; recent efforts by Michaels et al. [51] introduced modulation techniques to enhance the security of the CAN protocol. Their proposal incorporates a rolling secret (watermark) aligned with primary bus messages through multiplexing based on Binary Phase Shifting Keying (BPSK) modulation. While this multiplexed watermark significantly improves security by ensuring transmitted message authentication, it solely addresses this aspect, leaving incomplete coverage to attacks such as MitM, as the watermark can be forged.

IV. CAN-MM Technology

CAN-MM technology offers a non-intrusive solution for implementing MAC-based message authentication and integrity checks without compromising payload capacity or backward compatibility with all CAN standard versions. This approach is especially relevant for CAN 2.0 applications, enabling the development of a secure CAN network with an large MAC digest size. Additionally, CAN-MM enhances response time and performance of MAC digest computation across all CAN versions.

Essentially, the underlying concept of CAN-MM involves utilizing digital modulation techniques (i.e., On-Off Keying (OOK)) to multiplex the transmission of the MAC digest with the original CAN frame payload. The OOK is a simple digital modulation scheme based on Amplitude-Shift Keying (ASK) commonly used in telecommunication [52], [53]. OOK transmits a logical one by sending a carrier wave signal, while the absence of the carrier wave represents a logical zero.

The MAC information is encoded by switching the carrier wave on and off. A logical zero is transmitted on the bus by generating the original CAN signals, while in the case of a logical one, a wave is added to the standard CAN electric signals (in both CANH and CANL). This wave acts as a carrier. Its amplitude is a configured parameter, with a value of $Vpp = 300\,mV$ in this study, to ensure sufficient margins when reconstructing the original signal at the receiver's side.

To combine the signals from the CAN frame and MAC digest, the CAN-MM system necessitates appropriate synchronization, as depicted in Fig. 2. The Identifier Extension (IDE) bit of the CAN Control field initiates the synchronization procedure. During this procedure, a synchronization sequence of logic "1" and "0" is introduced on the MAC CODE RX line for the entire duration of the Control field. These values are modulated with the content of the Control field. Subsequently, the MAC digest is modulated onto the data payload. Finally, to enhance the reliability of the system, the CRC of the MAC digest is modulated onto the CRC slot of the payload. The CRC is a specialized checker to detect transmission errors. Multiplexing the MAC digest directly with the message ensures a strong link between the MAC code and the corresponding message, bolstering security by minimizing vulnerabilities such as message and code separation.

Fig. 3 depicts the effect of using the CAN-MM approach on the CAN 2.0 and CAN FD frames. In both cases, modulating the MAC helps maintain the full payload capacity of the frame; when the frame is long enough, e.g., the CAN FD, it reduces the necessary size of the frame while retaining the same amount of information. This reduction limits the need for the extra transmission time caused by appending the MAC to the data payload or as extra frames [22] when the selected MAC length is above 64 bits. It also might help optimize the system's real-time performance and the CAN bus load of the entire vehicle network.

The CAN-MM architecture consists of two main blocks: a transmitter and a receiver module.

In the left part of Fig. 4, the original transmitter (CAN controller and CAN transceiver blocks) is coupled with the additional functional components required to implement the CAN-MM schema in the bottom left. A multiplexer block is employed to multiplex the MAC-related information. This block includes a diverter switch [54] with two inputs, namely a carrier supplied by an internal generator and ground. The modulated CAN signal is applied to both CANH and CANL. The multiplexer is controlled by the MAC bitstream to provide

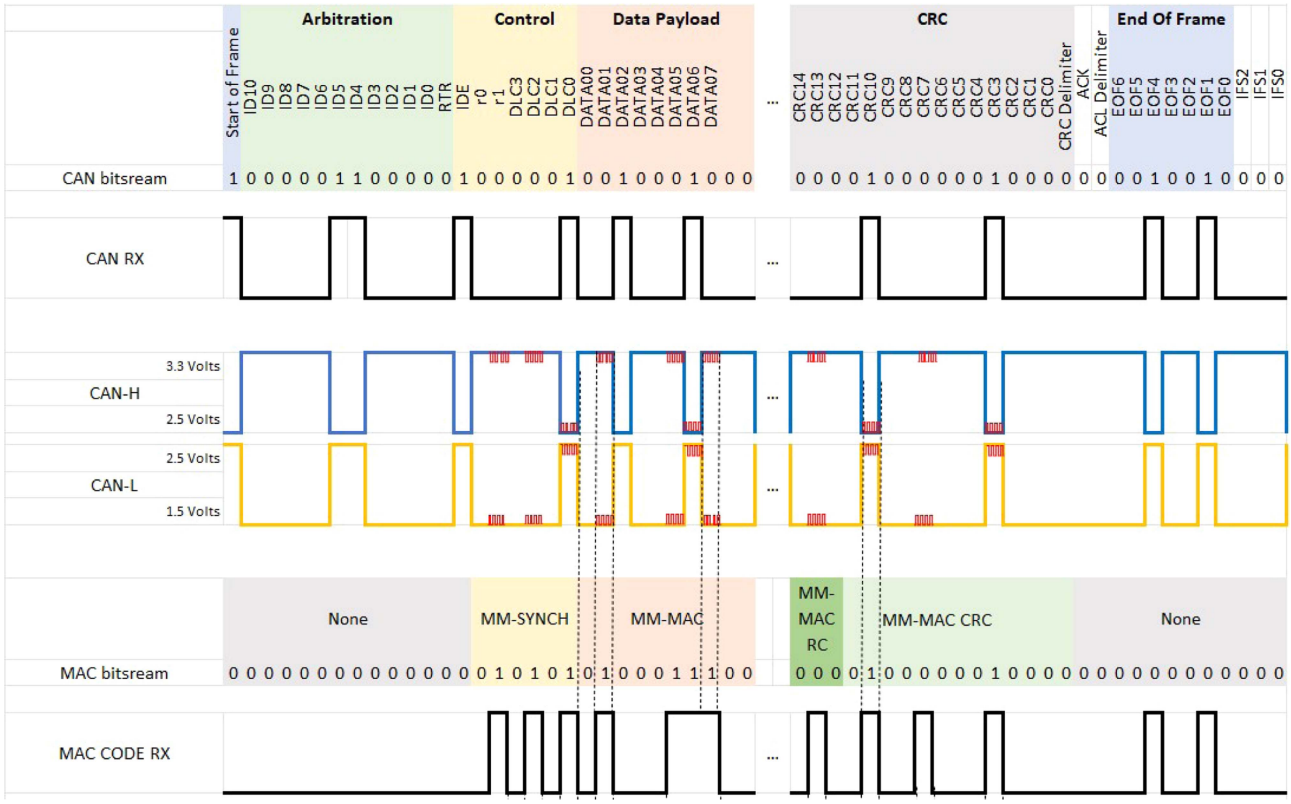

Fig. 2. Physical electrical CAN-MM signal scheme.

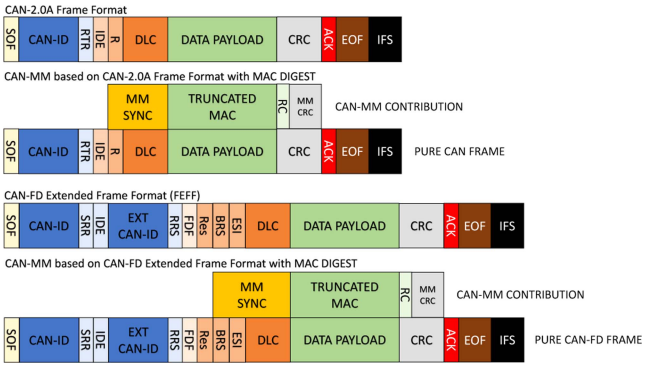

Fig. 3. Application of the CAN-MM technology to both CAN 2.0 and CAN FD frames.

a carrier as output when the corresponding MAC bit is one and no contribution when the corresponding bit of the MAC is zero. The multiplexer control line is synchronized with the CAN controller to multiplex the MAC information with the CAN payload.

In the right part of Fig. 4, the receiver includes a decoder block that is responsible for extracting the multiplexed MAC bitstream sequence from the payload of the CAN frame. The MAC decoder block utilizes a hybrid analog-digital electronic network to extract the correct CAN-MM contributions encapsulated in the CAN physical signal. The standard CAN receiver and the MAC decoder operate in parallel, eliminating the necessity of extra computation time for MAC extraction. While the transceiver processes the CAN frame, the decoder reconstructs the MAC bitstream. This allows the ECU to receive the CAN data payload and its MAC code in a shorter time window compared to existing solutions [12].

The custom CAN-MM components are situated downstream of the standard CAN interface to ensure full electrical compatibility with existing CAN interfaces.

To further explain the CAN-MM decoder, Fig. 5 shows the complexity of the analog-digital electronic required. The decoder is composed of four stages, which are as follows:

- *Filtering:* This stage is replicated for both CANH and CANL. It includes a band-pass filter with a center frequency *fc* at the frequency of the carrier signal.
- *Comparing:* This stage is a threshold comparator that operates on both CANH and CANL to identify the specific area where the carrier signal is present.
- *Conjunction:* This stage combines the analog data from both CAN lines into a single digital signal stream.
- *Counter:* The final stage is a logical network that identifies the area of the carrier signal in the digital domain.

These stages work together in a highly coordinated fashion to accurately extract the modulated information from the CAN channel.

V. Validation Model

A. Experimental Setup

The validation of the CAN-MM architecture considers a typical application scenario, specifically, a standard automotive CAN 2.0 network operating at a speed of 500 kbps. A

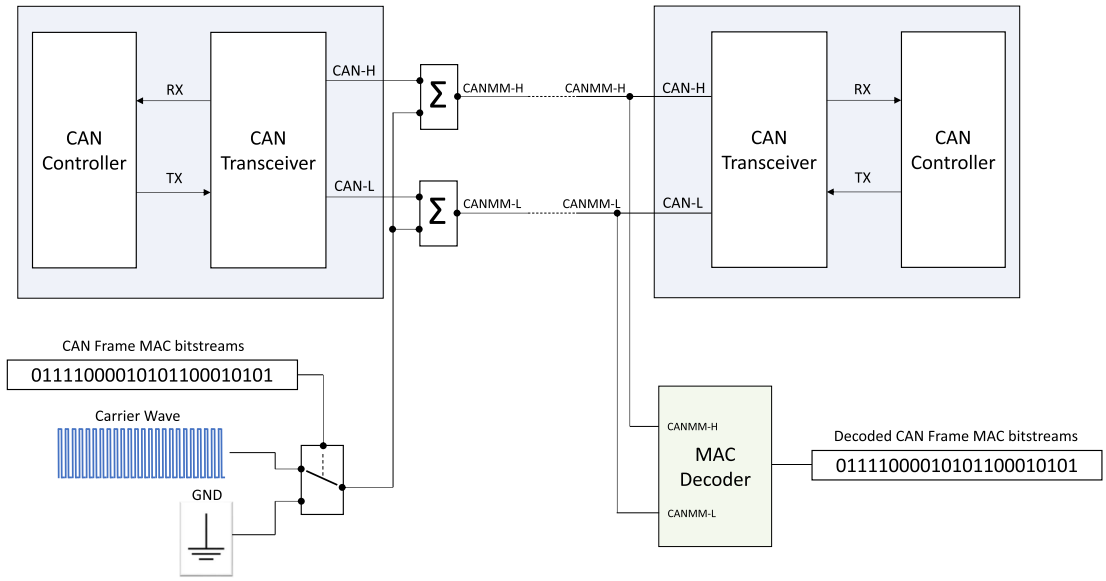

Fig. 4. CAN-MM Transmitter & Receiver block scheme.

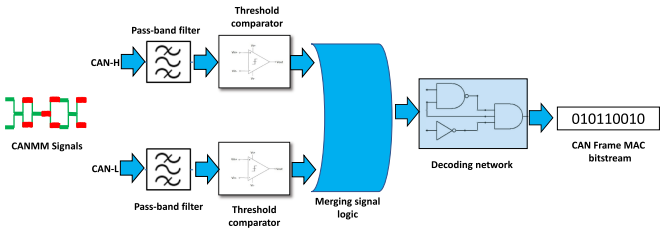

Fig. 5. CAN-MM MAC decoder Type-A block scheme.

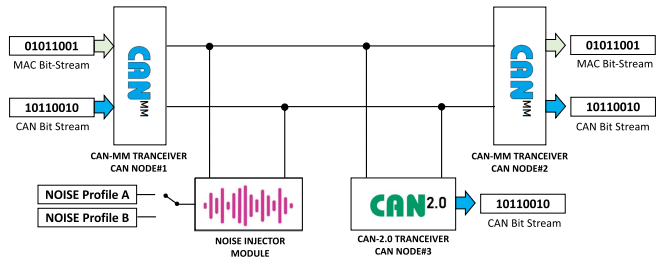

Fig. 6. Block scheme of the CAN-MM validation setup.

hybrid automotive CAN network comprising three CAN nodes was designed and simulated using the LTSpice [55] simulation environment to validate the architecture. Two nodes were CAN-MM transceivers, one serving as a transmitter and the other as a receiver. The third node was a standard CAN version 2.0 receiver. This setup enabled the validation and verification of the CAN-MM functionality and its backward compatibility with standard CAN transceivers. The complete block diagram for this configuration is presented in Fig. 6.

B. Noise and Interference Analysis

CAN systems boast a robust immunity to ground noise and electromagnetic interference, thanks to differentially transmitted

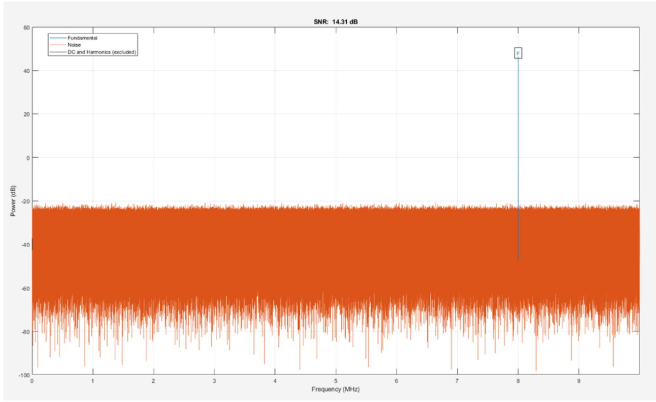

Fig. 7. SNR graph for real CAN recorded signals.

information, independent ground reference, usage of twisted-pair cabling, and balanced differential transceivers.

Since the CAN-MM technology is modifying the original profile of the CAN signals, evaluating it under realistic noisy environments is crucial. A validation environment simulated standard vehicle noise to assess noise and interference effects on CAN-MM technology. The noise profile is acquired using a multi-protocol vehicle interface device connected to an actual vehicle's OBD port. The device, programmed to transmit a specific CAN frame to the ECM, captures the physical CAN signal via an oscilloscope. Direct access to the CAN bus input of the ECU is facilitated through a break-out box. The noise profile is obtained during engine idle, aligned with specifications from various research papers [56], [57], [58]. Noise signals, recorded from both CAN lines with the same phase, cover frequencies from 10 kHz to 10 MHz, with amplitudes between −100 mV and 100 mV. Signal-to-Noise Ratio (SNR) calculations involve computations on two identical carrier signals with a peak-to-peak amplitude of 300 mV. The SNR for this scenario was calculated to be approximately 14.31 dB (Fig. 7). This value provides

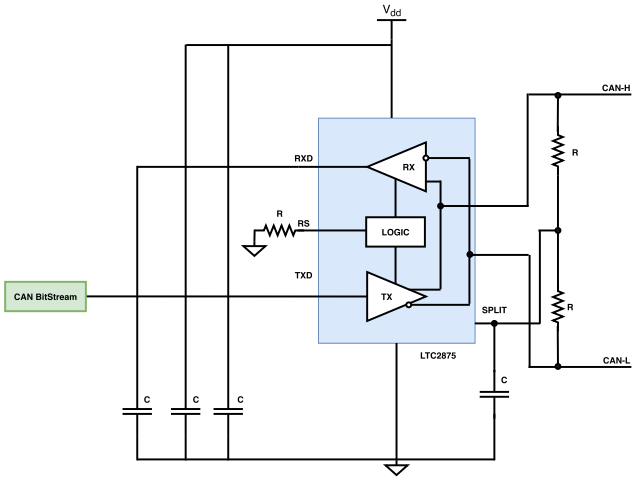

Fig. 8. CAN-MM transceiver - Stage 1 - SPICE block.

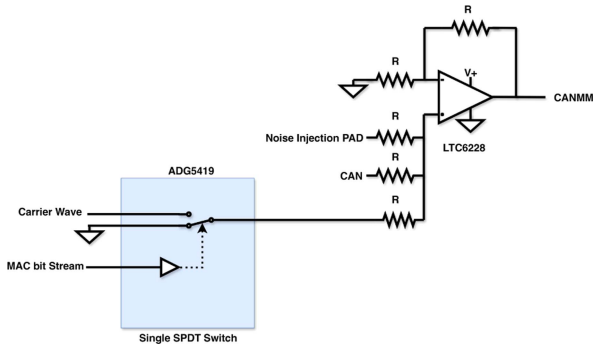

Fig. 9. CAN-MM transceiver - Stage 2 - SPICE block.

insight into the signal's quality relative to the background noise with the current parameters.

C. SPICE Model

The SPICE simulation incorporates input signals, such as the CAN bitstream and its associated MAC, generated from Piecewise Linear (PWL) files. Supplementary signals, including noise profiles, follow the same method with their respective PWL files. Standard library parts provided by the tool are utilized for the remaining design components.

In this setup, the LTC2875 standard CAN transceiver (refer to Fig. 8) is employed, as depicted in Fig. 6. The CAN-MM added part features a High Voltage Latch-Up Proof and a Single pole double throw (SPDT) Switch. Depending on the control value, this block outputs either the carrier wave or zero, subsequently added to the CANH and CANL signals provided by LTC2875, along with the noise contributions (see Fig. 9).

Unlike the transmitter, the custom part of the CAN-MM receiver processes data in parallel to the standard transceiver (refer to Fig. 10). The receiver includes a pass-band analog filter with a cutoff frequency set to the carrier frequency, followed by a voltage comparator with a voltage reference set to the absolute value of the noise (in this case, 100 mV). These stages form the first decode chain for CAN-MM and are identical for both CAN lines (see Fig. 10).

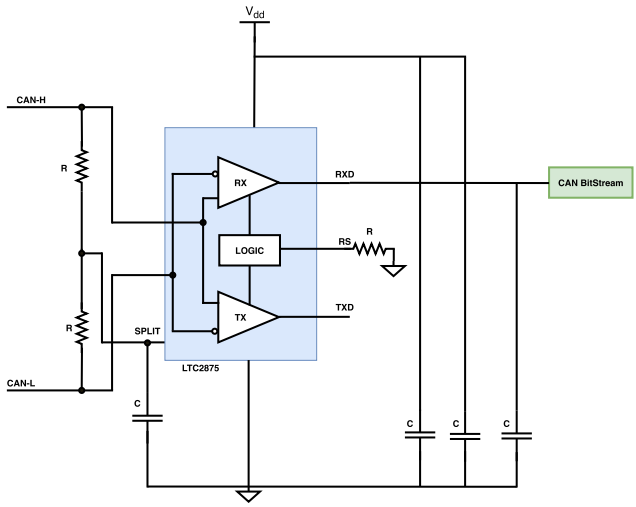

Fig. 10. CAN-MM receiver - Stage 1 - SPICE block.

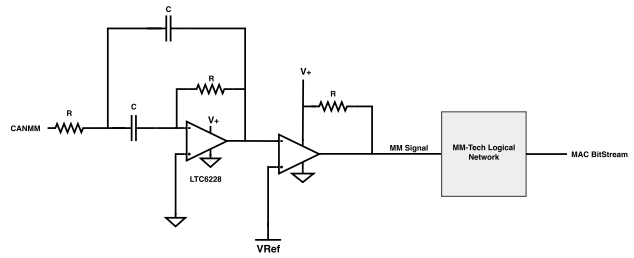

Fig. 11. CAN-MM receiver - Stage 2 - SPICE block.

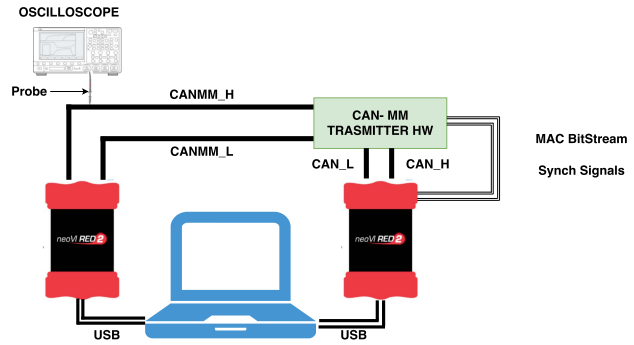

Fig. 12. CAN-MM hardware concept scheme.

In the second stage of the CAN-MM receiver, the contribution on the two CAN lines is collapsed together through a NOR port. Downstream of the NOR port is a custom logical network based on flip-flop counters, which is used to extract the MAC contribution (refer to Fig. 11).

D. Preliminary Hardware Implementation

A hardware prototype was created to enhance the validation of the CAN-MM technology. The prototype is specifically designed to assess the functionality of the CAN-MM transmitter. It is implemented within a compact In-Loop CAN network, as illustrated in Fig. 12. The primary goal of this validation is to confirm the capability of a standard receiver to receive the CAN-MM conditioned signal accurately.

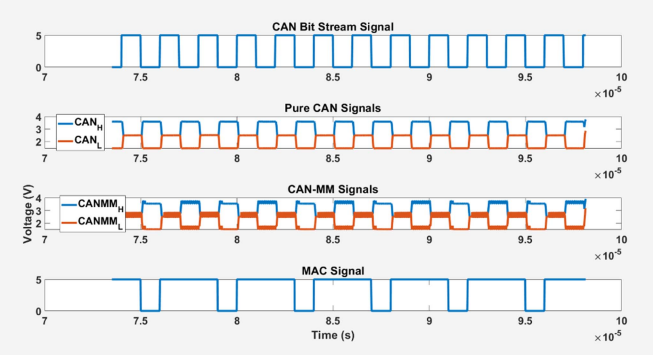

Fig. 13. CAN-MM transmitter output.

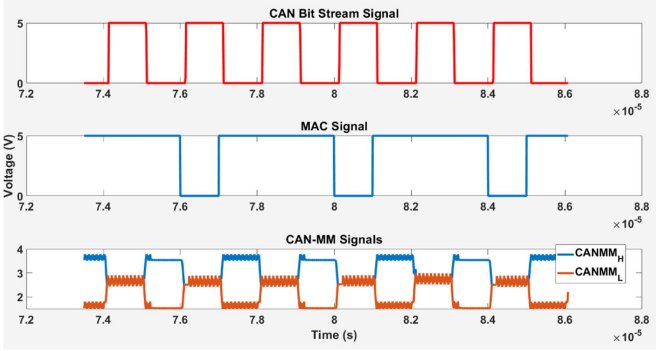

Fig. 14. CAN-MM receiver signals.

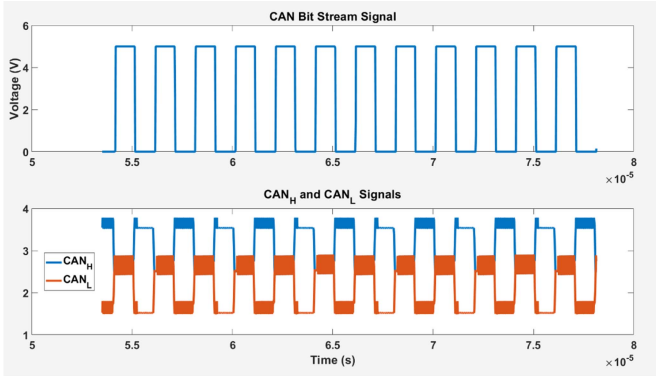

Fig. 15. CAN 2.0 transceiver.

The experimental setup involves a laptop connected to a Neo VI Multi-Protocol Vehicle Interface, which oversees a custom hardware board designed for CAN-MM operation. This board is crucial for converting the incoming CAN signal, received through the Neo VI interface, into a CAN-MM frame. The conversion process is directed by control signals continuously managed by the Neo VI device. Additionally, the hardware board is linked to another Neo VI device via the CAN-MM bus, set up to function under the standard CAN protocol. This configuration creates a closed loop with the laptop, facilitating seamless communication.

Notably, the CAN-MM bus is deliberately designed to be open-access, enabling the intentional introduction of noise and permitting data acquisition with an oscilloscope. In the second stage of the loop-back scheme, a programmable noise source was also added to simulate the noise profile acquired during the idle operation of the engine, as previously used in the LT-Spice simulations.

VI. Experimental Results

The collected signal diagrams, illustrated in the following figures, show the electrical signals generated by each module depicted in Fig. 6. The output signals generated by CAN-MM node #1 are illustrated in Fig. 13, which depicts four subplots. The blue line in the first subplot illustrates a section of a transmitted CAN bitstream, while the second one displays the differential electrical signals. The third subplot shows the CAN-MM electrical signals that are transmitted on CANH and CANL, where MAC signal in the fourth subplot is multiplexed.

Fig. 14 depicts the functionality of the CAN-MM receiver in node #2. It shows how the receiver manages the physical signal generated by the CAN-MM transmitter and transmitted on the bus. The bottom subplot displays the received CAN-MM physical signal through the CAN-MM transceiver, which is identical to the signal transmitted by node #1 in Fig. 13. The subplot in blue color is the MAC bitstream extrapolated by the CAN-MM decoder in node #2, and it is the corresponding MAC of the subplot in red color.

To demonstrate the complete compatibility of CAN-MM with the standard CAN 2.0 protocol, node #3 simulates a standard CAN 2.0 transceiver. As shown in Fig. 15, the backward compatibility is guaranteed, as the transceiver can reconstruct the correct CAN bitstream when it receives a CAN frame modulated under CAN-MM specifications. However, a standard CAN transceiver lacks the extended hardware required to demodulate the MAC bitstream, making it impossible to extract it.

To support a timewise analysis of the CAN-MM to understand the potential benefits of the parallel transmission of the MAC alongside the data payload, we computed the MAC transmission Extra Time (MET), introduced by the transmission of the MAC digest. It depends on the MAC's length in bits ($MACsize$) and the selected CAN protocol transmission time of a data bit (τ_{dbit} [22]), as shown in (1).

$$MET = MACsize * \tau_{\text{dbit}} \tag{1}$$

Aligning with the experimental setup in [22], we computed MET using τ_{dbit}=0.00025 (ms) for the CAN FD and τ_{dbit} equal to 0.0001 (ms) for the CAN XL.

In a CAN FD the MET required to transmit the 64-bit MAC digest is 16 μs, as per (2)

$$MET = MACsize * \tau_{\text{dbit}} = 64 * 0.00025 = 16(\mu s) \tag{2}$$

Adopting a more traditional baud rate on CAN FD, 500 kbps, we calculate a $\tau_{\text{dbit}} = 0.002(ms)$. In this condition, the extra transmission time required by MAC appended to the payload is 128 μs (see (3)).

$$MET = MACsize * \tau_{\text{dbit}} = 64 * 0.002 = 128(\mu s) \tag{3}$$

Keeping the MAC's size constant, adopting the CAN XL protocol with a speed rate of 10 Mbps, the MET would be reduced to

6.4 μs, which represents the best possible transmission performance by SecOC and CANsec, as per equation (4), demonstrating that a broad adoption fo CAN XL would introduce faster performance.

$$MET = MACsize * \tau_{\text{dbit}} = 64 * 0.0001 = 6.4(\mu s) \quad (4)$$

Opting for CAN-MM instead highlights a key benefit: the negligible impact on transmission times due to MAC. This capability to maintain consistent transmission times, with or without MAC, offers a solution to the schedulability challenges discussed by Ikumapayi et al. [22]. Moreover, CAN-MM supports countermeasures on the schedulability noted by the authors of [39]. The systems described in the paper adopt Rate-monotonic scheduling (RMS), a deterministic scheduling algorithm for real-time operating systems that assign priorities to tasks based on their period; the shorter the period, the higher the priority. A pivotal aspect of RMS is its CPU utilization bound for n periodic tasks, which can be calculated using the Liu & Layland formula, (5), where C_i is the computation time of task i, T_i is the period of task i, and U is the total CPU utilization. This formula ensures that if the total CPU utilization is below a certain threshold, all tasks can be scheduled to meet their deadlines, making RMS particularly efficient for systems with hard real-time constraints.

$$U = \sum_{i=1}^{n} \frac{C_i}{T_i} \leq n \left(2^{\frac{1}{n}} - 1\right) \quad (5)$$

The transmission time of the CAN and the MAC might significantly contribute to C_i, the computational load. By reducing the transmission time, CAN-MM directly decreases C_i and, consequently, the total CPU utilization. This reduction is crucial for enhancing resilience against certain types of attacks.

To provide a general understanding, the HSM performance metrics published by Pott [59] indicate that more than 300 clock cycles are required for MAC verification. When considering latency, the total time is approximately 5–6 μs, which parallels the time savings achieved by CAN-MM compared to CAN XL. Consequently, this denotes that CAN-MM might theoretically offer a twofold increase in the system's ability to withstand such attacks, in contrast to the conventional CAN XL framework where the MAC is appended to the payload.

The robustness of CAN-MM was further validated through measures performed on the hardware implementation introduced in Section V-D. These results complement the ones produced by the LT-SPICE simulations. The captured data in Fig. 16 portrays the real-time CAN-MM-H bus traffic. The applied noise profile follows what has been captured from a vehicle as described in Section V-B. Within this experimental framework, the CAN-MM transmitter effectively performs the multiplexing of the MAC Bitstream, precisely the bit sequence `00001110101111010111`, over the underlying physical CAN-H signal. This multiplexing process is executed through the OOK modulation technique, closely replicating the observations obtained in the simulated environment, thus confirming the robustness of the CAN-MM system.

Moreover, the BUSMASTER [60] tool reported error-free reception of the transmitted CAN message. This confirms the backward compatibility of the CAN-MM approach with conventional hardware. The multiplexed carrier of the standard transceiver is intelligently filtered out, effectively treating it as noise in the system.

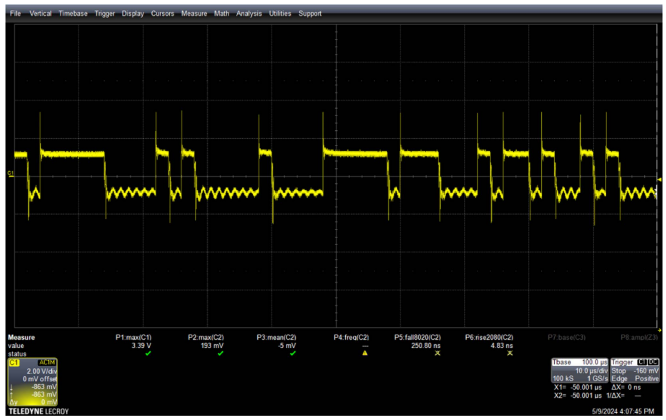

Fig. 16. CAN-MM-H acquired by oscilloscope.

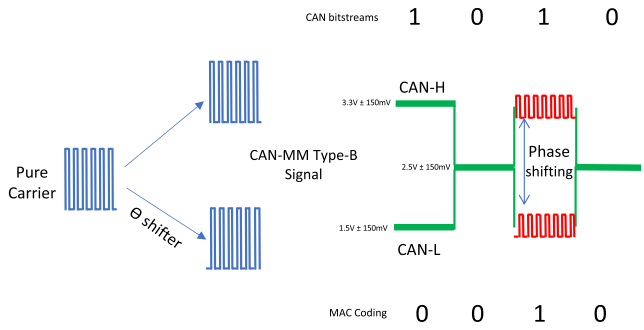

Fig. 17. CAN-MM Type-B physical signals scheme.

VII. CAN-MM Type-B

Section VI highlights a potential limitation in the CAN-MM architecture when the carrier and noise frequencies align, manifesting sporadic failures in demodulating the MAC bit-stream. While this scenario is unlikely to occur in actual situations, considering that noise amplitudes exceeding 100 mV are seldom encountered, this paper introduces an advanced CAN-MM architecture called Type-B, able to withstand scenarios where the carrier signal frequency matches the noise. CAN-MM Type-B ensures additional robustness to noise across all frequency bands without risking data corruption.

The CAN-MM Type-B physical signals scheme incorporates Carrier Phase Shift Modulation (CPSM) [61] as depicted in Fig. 17. The CPSM carrier varies between CANH and CANL, causing a phase shift ranging from 90° to 270°. The proposed design sets the phase modulation to 90° for CANL as depicted in Fig. 20.

The additional phase-shifting can result in incorrect codification, particularly if the differential voltage in the red area depicted in Fig. 19 exceeds the 0.5 V threshold. To overcome this limitation, an additional re-phaser stage represented by the orange area in the receiver reported in Fig. 18 reverses the CPSM applied by the CAN-MM Type-B transmitter. This block

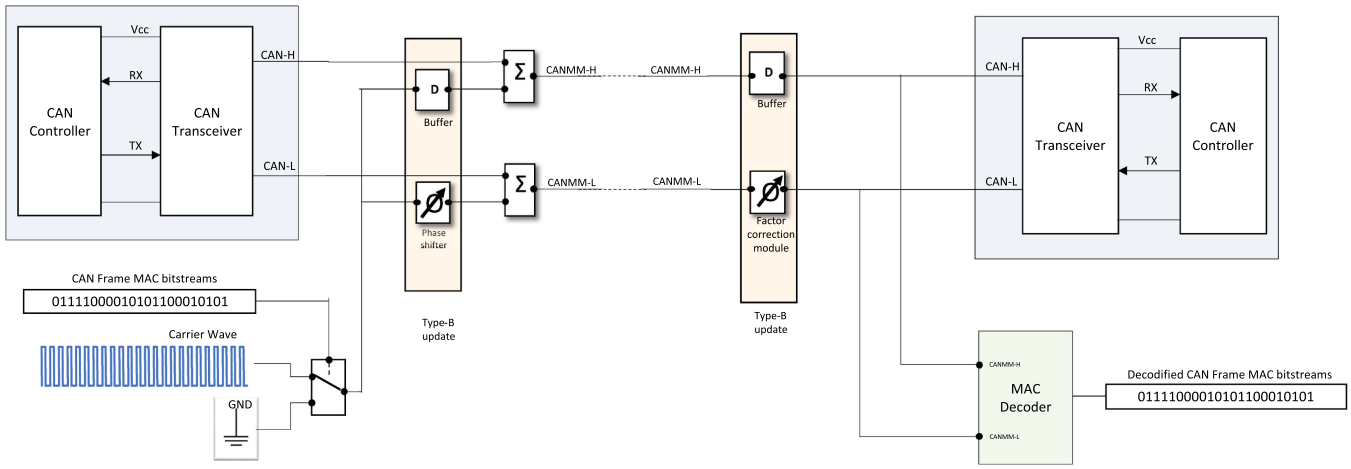

Fig. 18. CAN-MM Type-B transmitter& receiver block scheme.

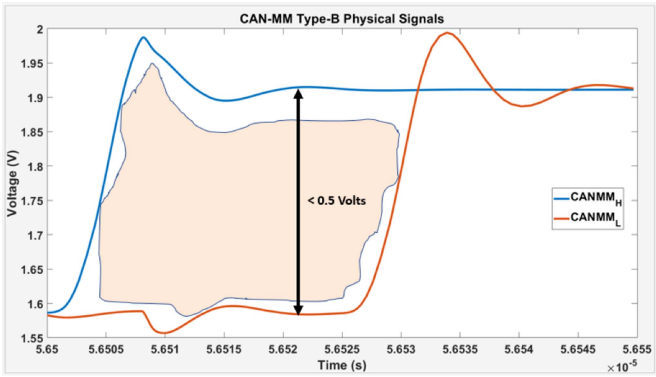

Fig. 19. Critical area due to shifting phase for codification correctness.

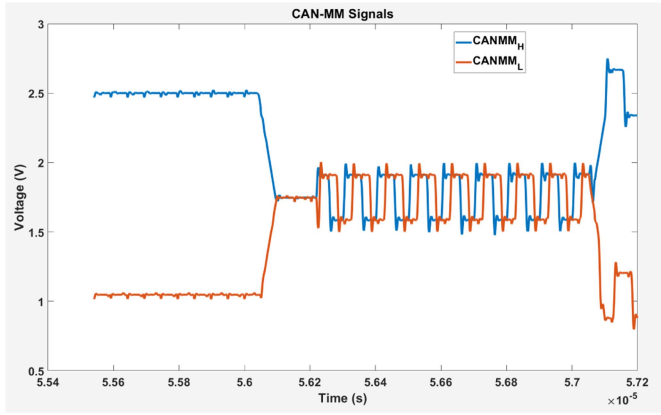

Fig. 20. CAN-MM type-B physical signal with the shifted carrier on CAN-L.

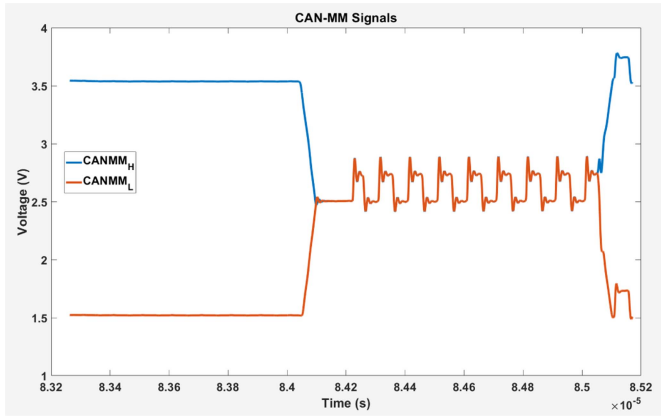

Fig. 21. CAN-MM type-B filter scheme.

is placed at the very beginning of the reception process. Once the re-phasing is completed, the standard CAN-MM receiver, which includes the standard CAN transceiver and the CAN-MM decoder, work in parallel to extract their respective data from the re-phased frame.

The additional protection to noise of CAN-MM Type-B across all frequency ranges comes with the cost of adding an upstream hardware re-phaser block to the CAN transceiver when it functions as a receiver.

An LT-Spice model was developed to validate the robustness of the CAN-MM Type-B architecture (Fig. 20). MAC code 1 is encoded by adding a carrier with a shifting phase on CANL, allowing for greater robustness during decoding activities. However, in certain regions, the phase shifting can cause the differential voltage between these signals to exceed the 0.5 V limit. Thus, as shown in Fig. 21, the signal is shifted back before decoding, obtaining full synchronization.

Fig. 22 presents a comparative comparison between CAN-MM and CAN-MM Type-B to validate the design robustness. This experiment applies a noise signal with a 140 mV amplitude to the original CAN-MM architecture model. The investigation is completed only for completeness since the resulting signal is clearly out of specification. As a result of the high noise level, the receiver could not extract the correct MAC bit-stream, and the output was a MAC bit-stream stuck to 1. However, in the case of CAN-MM Type-B, despite a noise signal with an amplitude of 200 mV, the receiver correctly decoded the MAC stream.

By referring to Fig. 23, we have calculated the signal-to-noise ratio (SNR) for this scenario to be approximately 17.32 dB. This

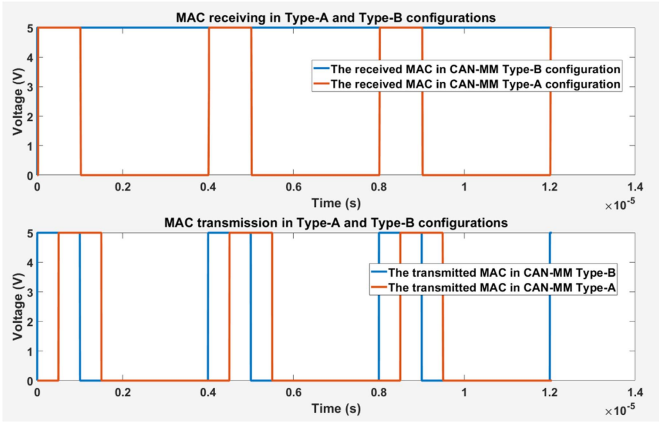

Fig. 22. CAN-MM Type-A vs. CAN-MM Type-B noise capability performances.

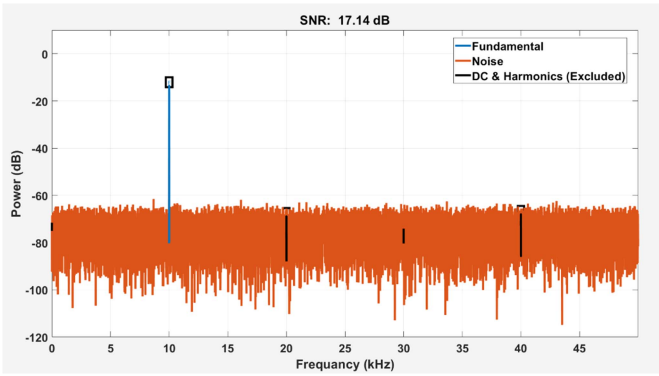

Fig. 23. SNR CAN-MM TypeB graph.

high SNR value underscores the signal's robustness, affirming its clear distinction from the surrounding background noise.

VIII. Security Analysis

This section delves into the security aspects of the CAN-MM architecture, particularly addressing attack models outlined in Section II.

The main objective of CAN-MM is to support a full CAN 2.0 vehicle network security by embedding a SecOC compatible MAC code within each payload frame, matching the same level of protection of CAN FD. Moreover, it supports security against threats such as MitM and replay attacks due to the presence of the MAC mechanism that neutralizes those types of attacks. This capability also includes the more recent Janus attack, as described by the author [37].

CAN-MM may also neutralize Cloak attacks by maintaining payload integrity, even amidst bit modifications. Leveraging the sample rate of two receivers will be more complex if the attacker also must coherently switch the modulated MAC. Such complexity will narrow the timing window where the attack is effective, as discussed in the original paper [38].

When a significant challenge arises when the system is overwhelmed by an excessive number of MACs that need to be validated [39], the validation process demands intensive cryptographic computations, potentially compromising the system's ability to adhere to real-time deadlines. This issue becomes particularly acute with the influx of numerous fraudulent MACs. The CAN-MM system introduces enhanced security measures against those kinds of attacks.

IX. Conclusion

This paper presented an efficient solution to mitigate security concerns within the automotive domain's fundamental communication protocol, the CAN. The proposed solution, CAN-MM, facilitates the transmission of MAC payloads in standard CAN to complement any security schemas based on it efficiently. The support of the MAC transmission also safeguards the automotive communication system against MitM and replay attacks.

The CAN-MM architecture, developed to upgrade communication hardware for upcoming security regulations, maintains compatibility with existing CAN devices, avoiding the necessity for a complete system or vehicle architecture overhaul. This hybrid networking capability offers flexibility to designers, minimizing the requirement for updating electronic components to the new generation and thereby reducing the cost of transitioning a vehicle fleet into the cyber-secure domain.

Additionally, an improved Type-B version of CAN-MM addresses potential demodulation issues without sacrificing backward compatibility. While this modified version may compromise some degree of backward compatibility, the applied modulation technology to the CAN protocol can be extended not only to version 2.0 but also to other existing versions that already incorporate the MAC.

References

[1] U. Abelein, H. Lochner, D. Hahn, and S. Straube, "Complexity, quality and robustness - The challenges of tomorrow's automotive electronics," in *Proc. IEEE Des. Autom. Test Europe Conf. Exhib.*, 2012, pp. 870–871.

[2] C. W. Axelrod, "Cybersecurity challenges of systems-of-systems for fully-autonomous road vehicles," in *Proc. IEEE 13th Int. Conf. Expo Emerg. Technol. Smarter World*, 2017, pp. 1–6.

[3] *Data Bridge*, "Global automotive cyber security market–industry trends and forecast to 2028," 2021. Accessed: Aug. 4, 2024. [Online]. Available: https://www.databridgemarketresearch.com/reports/global-automotive-cyber-security-market

[4] S. Ghosh, *Automotive Cybersecurity: From Perceived Threat to Stark Reality*. Warrendale, PA, USA: Soc. Automot. Engineers, 2016.

[5] *Road Vehicles – Cybersecurity Eng.*," ISO/SAE Standard 21434:2021, International Organization for Standardization, Geneva, Switzerland, 2021. Accessed: Aug. 4, 2024. [Online]. Available: https://www.iso.org/standard/70918.html

[6] *Road Vehicles – Guidelines for Auditing Cybersecurity Engineering*, ISO/PAS Standard 5112:2022, Geneva, Switzerland, 2022. Accessed: Aug. 4, 2024. [Online]. Available: https://www.iso.org/standard/80840.html

[7] UN Economic Commission for Europe, "UN regulation no. 155 - cyber security and cyber security management system," 2021. Accessed: Aug. 4, 2024. [Online]. Available: https://unece.org/transport/documents/2021/03/standards/un-regulation-no-155-cyber-security-and-cyber-security

[8] UN Economic Commission for Europe, "UN regulation no. 156 - software update and software update management system," 2021. Accessed: Aug. 4, 2024. [Online]. Available: https://unece.org/transport/documents/2021/03/standards/un-regulation-no-156-software-update-and-software-update

[9] F. Oberti, A. Savino, E. Sanchez, F. Parisi, and S. Di Carlo, "EXTAURUM P2T: An extended secure CAN-FD architecture for road vehicles," *IEEE Trans. Device Mater. Rel.*, vol. 22, no. 2, pp. 98–110, Jun. 2022.

[10] F. Oberti, E. Sanchez, A. Savino, F. Parisi, and S. Di Carlo, "Mitigation of automotive control modules hardware replacement-based attacks through

hardware signature," in *Proc. IEEE/IFIP 51st Annu. Int. Conf. Dependable Syst. Netw. - Supplemental Volume*, 2021, pp. 13–14.
[11] F. Oberti, E. Sanchez, A. Savino, F. Parisi, M. Brero, and S. D. Carlo, "LIN-MM: Multiplexed message authentication code for local interconnect network message authentication in road vehicles," in *Proc. IEEE 28th Int. Symp. on-Line Testing Robust Syst. Des.*, 2022, pp. 1–7.
[12] C. Lin and A. Sangiovanni-Vincentelli, "Cyber-security for the controller area network (CAN) communication protocol," in *Proc. IEEE Int. Conf. Cyber Secur.*, 2012, pp. 1–7.
[13] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive can networks–practical examples and selected short-term countermeasures," *Rel. Eng. Syst. Saf.*, vol. 96, no. 1, pp. 11–25, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0951832010001602
[14] H. J. Jo and W. Choi, "A survey of attacks on controller area networks and corresponding countermeasures," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 6123–6141, Jul. 2022.
[15] S. Jadhav and D. Kshirsagar, "A survey on security in automotive networks," in *Proc. IEEE 4th Int. Conf. Comput. Commun. Control Automat.*, 2018, pp. 1–6.
[16] U. S. Environmental Protection Agency, "Vehicle emissions on-board diagnostics (OBD)," 2009. Accessed: Aug. 4, 2024. [Online]. Available: https://19january2017snapshot.epa.gov/state-andlocal-transportation/vehicle-emissions-board-diagnostics-obd_.html
[17] *UNECE*, "Transport - vehicle regulations, WLTP task force on on-board-diagnostic (OBD)," 2012. [Online]. Available: https://wiki.unece.org/pages/viewpage.action?pageId=2523184
[18] E. Aliwa, O. Rana, C. Perera, and P. Burnap, "Cyberattacks and countermeasures for in-vehicle networks," *ACM Comput. Surv.*, vol. 54, pp. 1–37, Mar. 2021.
[19] Z.-A. Zhao, Y. Sun, D. Li, J. Cui, Z. Guan, and J. Liu, "A scalable security protocol for intravehicular controller area network," *Secur. Commun. Netw.*, vol. 2021, pp. 1–13, Dec. 2021.
[20] D. Zelle and S. Gürgens, "BusCount: A provable replay protection solution for automotive CAN networks," *Secur. Commun. Netw.*, vol. 2021, pp. 1–26, Nov. 2021.
[21] *AUTOSAR (AUTomotive Open System ARchitecture)*, "Classic platform," 2022. Accessed: Aug. 4, 2024. [Online]. Available: https://www.autosar.org/standards/classic-platform
[22] O. Ikumapayi, H. Olufowobi, J. Daily, T. Hu, I. C. Bertolotti, and G. Bloom, "CANASTA: Controller area network authentication schedulability timing analysis," *IEEE Trans. Veh. Technol.*, vol. 72, no. 8, pp. 10024–10036, Aug. 2023.
[23] B. Groza, S. Murvay, A. V. Herrewege, and I. Verbauwhede, "LiBrA-CAN: Lightweight broadcast authentication for controller area networks," *ACM Trans. Embed. Comput. Syst.*, vol. 16, no. 3, Apr. 2017, Art. no. 90, doi: 10.1145/3056506.
[24] A. Albert, "Comparison of event-triggered and time-triggered concepts with regard to distributed control systems," *Embedded world*, Feb. 2004, pp. 235–252.
[25] Road Vehicles – Controller Area Network (CAN) – Part 2: High-Speed Medium Access Unit," ISO Standard 11898-2:2016, International Organization for Standardization, Geneva, Switzerland, 2016. Accessed: Aug. 4, 2024. [Online]. Available: https://www.iso.org/standard/67244.html
[26] Road Vehicles – Controller Area Network (CAN) – Part 3: Low-speed, Fault-Tolerant, Medium-Dependent Interfaceg, ISO Standard 11898-3:2006, International Organization for Standardization, Geneva, Switzerland, 2006. Accessed: Aug. 4, 2024. [Online]. Available: https://www.iso.org/standard/36055.html
[27] C. Kaiser, A. Stocker, and A. Festl, "Automotive can bus data: An example dataset from the aegis Big Data project," 2019.
[28] Z. Bi, G. Xu, C. Wang, G. Xu, and S. Zhang, "A method for translating automotive body-related can messages based on labeled bits," *Appl. Sci.*, vol. 13, no. 3, 2023, Art. no. 1942. [Online]. Available: https://www.mdpi.com/2076-3417/13/3/1942
[29] A. Gazdag, R. Ferenc, and L. Buttyán, "CrySyS dataset of CAN traffic logs containing fabrication and masquerade attacks," *Sci. Data*, vol. 10, no. 1, 2023, Art. no. 903, doi: 10.1038/s41597-023-02716-9.
[30] *Data Communication*, ISO/TC Standard 22/SC 31, International Organization for Standardization, Geneva, Switzerland, 2022. Accessed: Aug. 4, 2024. [Online]. Available: https://www.iso.org/committee/5383568.html
[31] M. U. Farooq, M. Waseem, A. Khairi, and S. Mazhar, "A critical analysis on the security concerns of Internet of Things (IoT)," *Int. J. Comput. Appl.*, vol. 111, no. 7,pp. 1–6, 2015.
[32] M. Bozdal, M. Samie, and I. Jennions, "A survey on can bus protocol: Attacks, challenges, and potential solutions," in *Proc. IEEE Int. Conf. Comput. Electron. Commun. Eng.*, 2018, pp. 201–205.
[33] G. Bloom, "WeepingCAN: A stealthy can bus-off attack," in *Proc. Workshop Automot. Auton. Veh. Secur.*, 2021, pp. 1–6. Accessed: Aug. 4, 2024. [Online]. Available: https://www.ndsssymposium.org/wp-content/uploads/autosec2021_23002_paper.pdf
[34] H. M. Song, H. R. Kim, and H. K. Kim, "Intrusion detection system based on the analysis of time intervals of can messages for in-vehicle network," in *Proc. IEEE Int. Conf. Inf. Netw.*, 2016, pp. 63–68.
[35] A. Gazdag, C. Ferenczi, and L. Buttyán, "Development of a man-in-the-middle attack device for the CAN bus," in *Proc. 1st Conf. Inf. Technol. Data Sci.*, 2021, pp. 115–130. [Online]. Available: https://api.semanticscholar.org/CorpusID:232361098
[36] P. Noureldeen, M. A. Azer, A. Refaat, and M. Alam, "Replay attack on lightweight CAN authentication protocol," in *Proc. IEEE 12th Int. Conf. Comput. Eng. Syst.*, 2017, pp. 600–606.
[37] K. Tindell, "The janus attack," 2024. Accessed: Aug. 4, 2024. [Online]. Available: https://www.can-cia.org/fileadmin/cia/documents/publications/cnlm/december_2021/21-4_p10_the_janus_attack_ken_tindell_canis_automotive_labs.pdf
[38] L. Yue, Z. Li, T. Yin, and C. Zhang, "CANCloak: Deceiving two ECUs with one frame," in *Proc. Workshop Automot. Auton. Veh. Secur.*, 2021, pp. 1–6. Accessed: Aug. 4, 2024. [Online]. Available: https://www.ndsssymposium.org/wp-content/uploads/autosec2021_23024_paper.pdf
[39] V. K. Kukkala, S. Pasricha, and T. Bradley, "SEDAN: Security-aware design of time-critical automotive networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9017–9030, Aug. 2020.
[40] S. Čapkun, M. Čagalj, R. Rengaswamy, I. Tsigkogiannis, J.-P. Hubaux, and M. Srivastava, "Integrity codes: Message integrity protection and authentication over insecure channels," *IEEE Trans. Dependable Secure Comput.*, vol. 5, no. 4, pp. 208–223, Fourth Quart. 2008.
[41] T. Ziermann, S. Wildermann, and J. Teich, "CAN: A new backward-compatible controller area network (CAN) protocol with up to 16× higher data rates," in *Proc. IEEE Des. Autom. Test Europe Conf. Exhib.*, 2009, pp. 1088–1093.
[42] A. Van Herrewege, D. Singelee, and I. Verbauwhede, "CANAuth–A simple, backward compatible broadcast authentication protocol for CAN bus," in *Proc. ECRYPT Workshop Lightweight Cryptogr.*, 2011, pp. 1–7. Accessed: Aug. 4, 2024. [Online]. Available: https://www.esat.kuleuven.be/cosic/publications/article-2086.pdf
[43] A. Hazem and H. Fahmy, "LCAP–A lightweight CAN authentication protocol for securing in-vehicle networks," in *Proc. 10th Escar Embedded Secur. Cars Conf.*, 2012, pp. 172–182. Accessed: Aug. 4, 2024. [Online]. Available: http://eece.cu.edu.eg/∼hfahmy/publish/escar2012.pdf
[44] J. Schmandt, A. T. Sherman, and N. Banerjee, "Mini-MAC: Raising the bar for vehicular security with a lightweight message authentication protocol," *Veh. Commun.*, vol. 9, pp. 188–196, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209616301619
[45] J. Luo, C.-M. Wu, and M.-H. Yang, "A CAN-bus lightweight authentication scheme," *Sensors*, vol. 21, pp. 1–28, Oct. 2021.
[46] J. Heitz, S. M. Murphy, and R. S. Wilder, "The AES-CMAC Algorithm. RFC 4493," Jun. 2006. Accessed: Aug. 4, 2024. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4493.html
[47] O. Esparza, W. Leichtfried, and F. González, "Transitioning applications from CAN 2.0 to CAN FD," in *Proc. IEEE Int. Conf. Commun.*, 2015, pp. 03-1–03-8. Accessed: Aug. 4, 2024. [Online]. Available: http://s3.eu-central-1.amazonaws.com/cancia-de/documents/proceedings/icc_2015_esparza.pdf
[48] M. D. Pesé, J. W. Schauer, J. Li, and K. G. Shin, "S2-CAN: Sufficiently secure controller area network," in *Proc. 37th Annu. Comput. Secur. Appl. Conf.*, 2021, pp. 425–438, doi: 10.1145/3485832.3485883.
[49] A.-I. Radu and F. D. Garcia, "LeiA: A lightweight authentication protocol for CAN," in *Proc. Eur. Symp. Res. Comput. Secur.*, 2016, pp. 283–300.
[50] T. Sugashima, D. Oka, and C. Vuillaume, "Approaches for secure and efficient in-vehicle key management," *SAE Int. J. Passenger Cars - Electron. Elect. Syst.*, vol. 9, no. 1, pp. 100–106, Apr. 2016.
[51] A. J. Michaels et al., "CAN bus message authentication via co-channel RF watermark," *IEEE Trans. Veh. Technol.*, vol. 71, no. 4, pp. 3670–3686, Apr. 2022.
[52] M. Forzati, "Phase modulation techniques for on-off keying transmission," in *Proc. IEEE 9th Int. Conf. Transparent Opt. Netw.*, 2007, pp. 24–29.
[53] S. Faruque, *Amplitude Shift Keying (ASK)*. Berlin, Germany: Springer, Jan. 2017, pp. 45–55.

[54] D. J. Rogers, T. C. Green, and R. W. Silversides, "A low-wear onload tap changer diverter switch for frequent voltage control on distribution networks," *IEEE Trans. Power Del.*, vol. 29, no. 2, pp. 860–869, Apr. 2014.
[55] Linear Technology Corporation, "LTSpice," Accessed: Aug. 4, 2024. [Online]. Available: http://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html
[56] M. Mizoguchi, H. Mori, N. Maeda, H. Keino, T. Yasuda, and H. Goto, "Alternative technique to estimate the immunity performance for in-vehicle ethernet," in *Proc. IEEE Asia-Pacific Int. Symp. Electromagn. Compat.*, 2016, pp. 703–705.
[57] D. Zhang, S. Zhang, T. Fan, and X. Wen, "Modeling and estimation for conducted common-mode interference of a motor drive system used in electric vehicle," in *Proc. IEEE 21st Int. Conf. Elect. Machines Syst.*, 2018, pp. 831–835.
[58] Z. Li, D. Shouquan, Z. Chengning, and W. Zhifu, "Study on electromagnetic interference restraining of electric vehicle charging system," in *Proc. IEEE 4th Int. Conf. Power Electron. Syst. Appl.*, 2011, pp. 1–4.
[59] C. Pott, P. Jungklass, D. J. Csejka, T. Eisenbarth, and M. Siebert, "Firmware security module," *J. Hardware Syst. Secur.*, vol. 5, no. 2, pp. 103–113, 2021, doi: 10.1007/s41635-021-00114-4.
[60] Robert Bosch Engineering and Business Solutions and ETAS GmbH, "BUSMASTER: An open source software tool for simulation, analysis, and testing of data bus systems," 2024. Accessed: Mar. 11, 2024. [Online]. Available: https://rbei-etas.github.io/busmaster/
[61] S. Agrawal and R. S. Kanchan, "Carrier phase shift modulation for reducing the common mode voltage in a two-level three-phase inverter," in *Proc. IEEE 44th Annu. Conf. Ind. Electron. Soc.*, 2018, pp. 1067–1072.

Franco Oberti (Member, IEEE) received the M.Sc. degree in computer engineering from the Politecnico di Torino, Torino, Italy, in 2007, and the master's degree in advanced cybersecurity from Standford University, Standford, CA, USA. In 2021, he was also an Industry Ph.D. student candidate. In 2007, he started working with Dumarey Softronix (former General Motors Powertrain Europe), where he held different positions. He is currently part of the Product Security Office in Dumarey Softronix. His research focuses on cybersecurity applied to embedded road vehicle systems.

Alessandro Savino (Senior Member, IEEE) received the M.S. and Ph.D. degrees in computer engineering and information technology from the Politecnico di Torino, Torino, Italy, in 2005 and 2009, respectively. He is currently an Associate Professor with the Department of Control and Computer Engineering, Politecnico di Torino, Torino, Italy. His research interests include approximate computing, system reliability, neuromorphic computing, safety-critical systems, software-based self-test, and bioinformatics. He has been part of the program and organizing committee of several IEEE and INSTICC conferences and was a reviewer of IEEE conferences and journals.

Ernesto Sanchez (Senior Member, IEEE) received the master's degree in electronic engineering from Universidad Javeriana, Bogota, Colombia, in 2000, and the Ph.D. degree in computer engineering from the Politecnico di Torino, Torino, Italy, in 2006. He is currently an Associate Professor with the Department of Control and Computer Engineering, Politecnico di Torino. His research interests include microprocessor testing, hardware security, and DNN reliability.

Paolo Casasso received the master's degree in electrical engineering from Politecnico of Turin, Turin, Italy, in 2000. As a Manager in Dumarey Softronix, he leads the hardware development team specializing in automotive embedded control systems. He held several positions in multinational automotive companies, such as FIAT-GM-Powertrain JV and General Motors for more than 20 years.

Filippo Parisi received the master's degree in electronic engineering from Politecnico of Turin, Turin, Italy, in 1992. As the Head of electronics with Dumarey Softronix, he leads the development of electronics, firmware, and virtualization for testing applied to hard real-time, safety-critical automotive embedded control systems. He held several positions in multinational automotive companies such as FIAT Research Center, FIAT-GM-Powertrain JV, and General Motors for more than 30 years.

Stefano Di Carlo (Senior Member, IEEE) received the M.Sc. degree in computer engineering and the Ph.D. degree in information technologies from Politecnico di Torino, Torino, Italy. He is currently a Full Professor with Politecnico di Torino. He has coordinated several national and European research projects. He has authored or coauthored more than 250 papers in peer-reviewed IEEE and ACM journals and conferences. His research interests include computer architecture reliability, safety, and security. He regularly serves on the Organizing and Program Committees of major IEEE and ACM conferences. He is a Golden Core member of the IEEE Computer Society.

Open Access provided by 'Politecnico di Torino' within the CRUI CARE Agreement