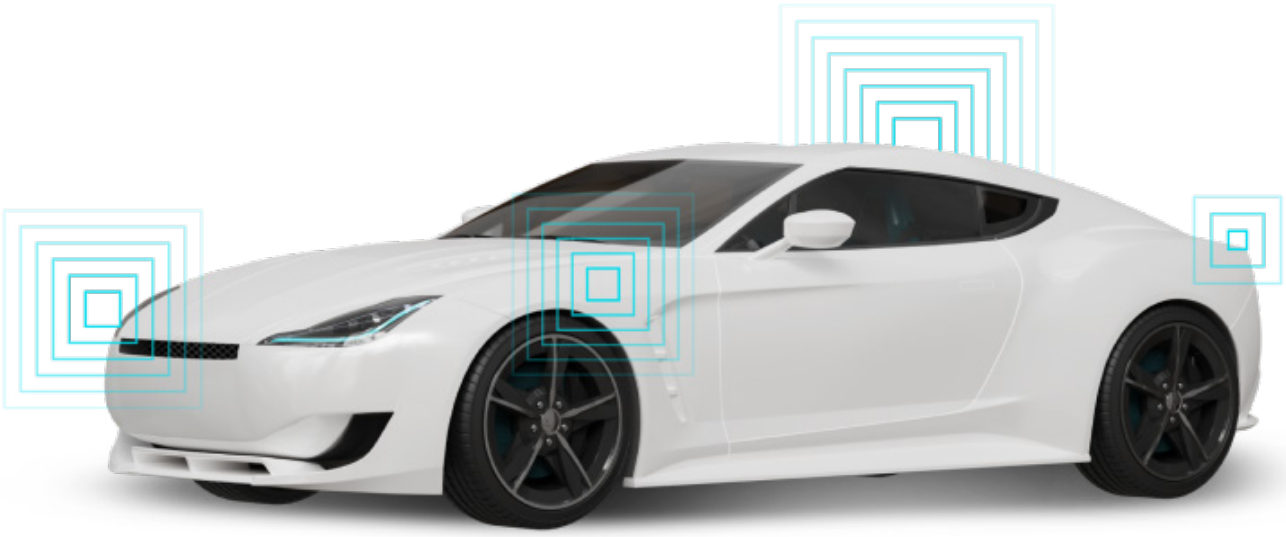
PLAXIDITYX

GO EVERYWHERE

A2B Network Security:

Protecting Automotive Audio Bus from Remote I2C Threats



Author:

Robbie Galfrin, Security Researcher

Table of Contents

Abstract

This whitepaper explores the security challenges associated with the Automotive Audio Bus (A2B) protocol, an advanced communication technology by Analog Devices that facilitates audio, sensor and control data transmission via a single unshielded twisted pair cable. While A2B enhances efficiency and flexibility, the integration of remote I2C (Inter-Integrated Circuit) communication introduces notable security threats. These threats could enable attacks such as lateral movement across nodes, unauthorized access to components, and intentional damage to electronic control units (ECUs). To help automotive developers counter these risks, this paper presents actionable mitigations, including hardware security measures like isolating the I2C bus and employing dedicated subordinate chips, alongside software-level strategies such as treating incoming data as untrusted, rigorous code reviews, and fuzz testing. By adopting robust mitigation strategies, the automotive industry can safely capitalize on A2B's innovative capabilities while maintaining stringent security standards.

Introduction

In recent years, automotive security has surged to the forefront of industry priorities, driven by the dual pressures of increasing cyber-based threats and physical vehicle theft, alongside stringent regulatory demands. While rapid technological advancements integrating new interfaces and connected systems into vehicles have enhanced convenience and functionality, these technologies have also expanded the attack surface. Each new interface brings unique risks, necessitating a meticulous evaluation to ensure security threats are not unintentionally introduced and are properly managed. This holds true for the Automotive Audio Bus (A2B) as well.

The A2B protocol is an innovative automotive audio solution by Analog Devices [1], offering streamlined communication for audio and control data over a single unshielded twisted pair cable throughout the entire vehicle. While A2B provides unmatched efficiency and flexibility, its implementation, particularly when using remote I2C (Inter-Integrated Circuit) communication, introduces security threats that are often overlooked. This paper explores these risks and proposes actionable mitigations.

A2B Overview

The A2B protocol, developed by Analog Devices, is primarily designed for automotive audio applications [2] but is also gaining traction in consumer electronics, industrial automation, and other industries. By enabling multiple nodes to connect on a single bus without requiring separate power lines, A2B simplifies system architecture and reduces costs.

Some common use cases for A2B technology include:

- Advanced Driver Assistance Systems (ADAS) for distributing audio and control signals
- Noise-canceling systems in vehicles
- Distributed microphone arrays
- Industrial automation for sensor integration

Method of Operation

The A2B network has a Single Main-Multiple Subordinates topology. It is mainly used to transfer audio in the form of I2S/TDM or PDM streams, as well as data & control signals using either the I2C-Over-Distance protocol or A2B Mailboxes. Additionally, the GPIO-Over-Distance protocol is supported, which enables the A2B Main to directly and remotely control GPIO pins on the A2B Sub Transceiver.

Communication is performed in time-divided cycles, where each cycle consists of a frame of data going downstream from the A2B Main to the A2B Subordinates, and then a frame of data going upstream from the last A2B Sub in the chain towards the A2B Main.

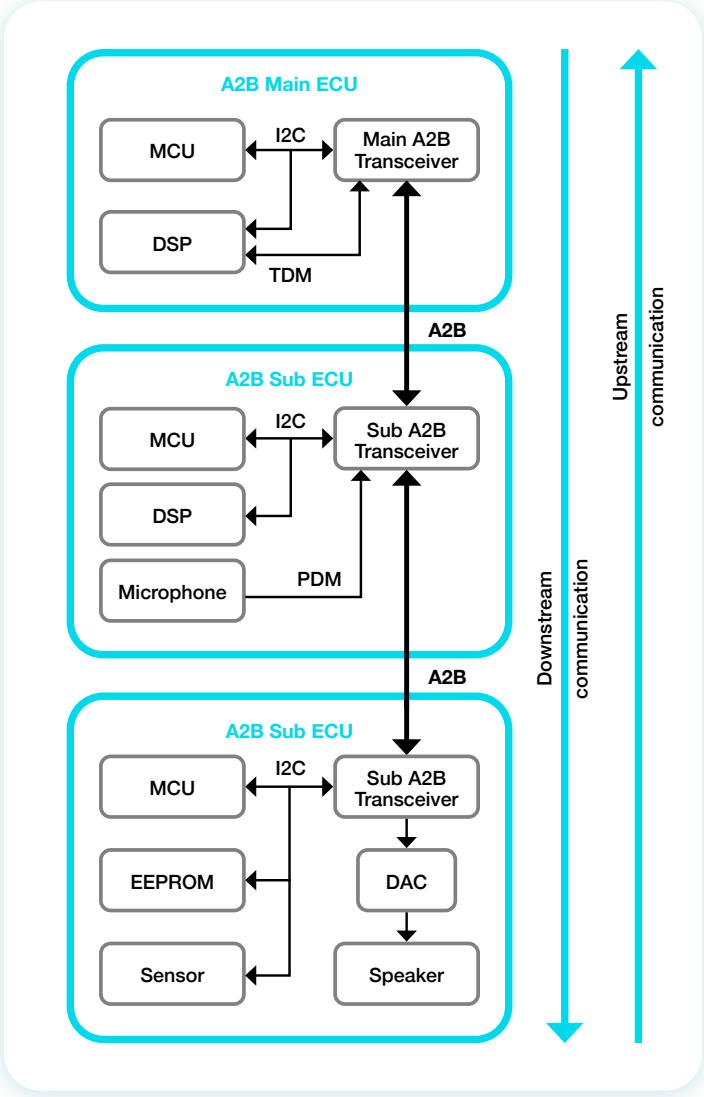


Figure 1. A2B Bus high level block diagram

Upon initialization, the Main performs network discovery & configuration and the network becomes active. Over this network, audio streams are defined (either statically prior to the start of operation or dynamically) by assigning TDM slots in either the upstream or downstream A2B superframes:

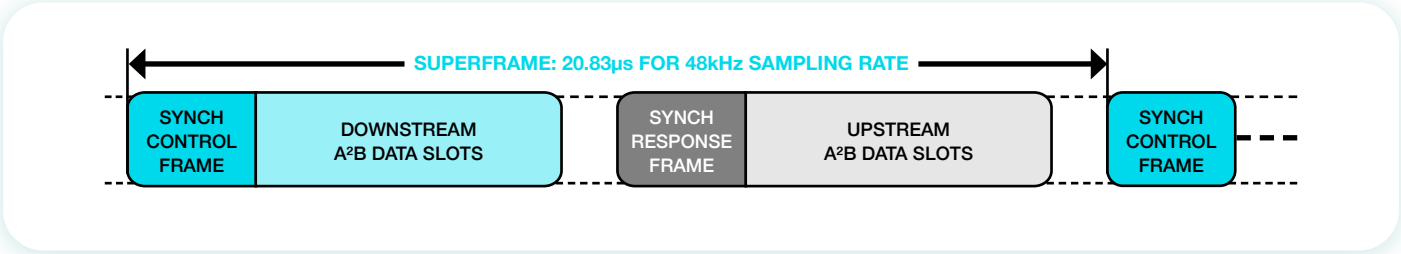


Figure 2. A2B superframe structure. Source: AD242x reference manual [4]

While its benefits are undeniable, remote I2C communication over the A2B protocol presents a double-edged sword, as it introduces another potential remote attack vector that should be considered carefully.

Remote I2C Communication: Expanding the Use of a PCB-level Communication Protocol

I2C-Over-Distance is used to transport data over A2B in a manner that can be easily adopted and interfaced. This method is used to transparently perform standard I2C communication remotely, transferring data over the A2B bus as part of the SYNCH control frames.

It allows the A2B Main MCU to remotely configure the A2B Subnodes, to communicate with the A2B Subordinate MCU, and to directly program EEPROM memories or configure sensors on the subnode.

There are two ways in which a host MCU connected to an A2B Main can communicate with components connected to the remote A2B Sub: I2C-Over-Distance and A2B Mailboxes.

I2C Over Distance

In I2C-Over-Distance, the A2B chips act as an I2C Bridge, such that components on each side “believe” they are communicating with a standard on-board I2C component. This method is often used to allow an MCU connected to the A2B Main node to communicate with peripheral devices (I2C Subordinates) connected to the A2B Sub node.

For example, if an MCU that is connected to the A2B Main wants to read some sensor data or program an I2C EEPROM on the A2B Sub, the MCU will communicate as an I2C Main and this communication will be transferred downstream over the A2B to the relevant A2B Sub.The A2B Sub will also act as an I2C Main and will mirror the message in the relevant address. The response from the component will be transferred back upstream to the A2B Main, which will act as an I2C Sub and mirror the response back to the MCU.

A2B Mailboxes

The A2B MailBox is a lightweight, asynchronous, bidirectional message pipe that each side can use to pass up to 4 bytes of data per transfer and then trigger an interrupt that will signal the other side to read the respective Mailbox over I2C.

This method is often used for communication between MCUs connected on both ends of the A2B Bus. Since this method is interrupt-based, it can be done asynchronously where each MCU acts as an I2C Main to read or write Mailboxes following an interrupt.

On each side there are two Mailboxes - one configured for reception and one for transmission. When an MCU wants to transmit data to the other side, it will write the data to the Tx Mailbox and then configure a register that will trigger an interrupt on the other side. The other side will receive the interrupt and can choose when to read the data. Once it reads the data, it notifies the sending side that the data was read so that additional data frames can be put in the Mailbox.

A2B Mailbox Communication Stack Example

In order to better understand the potential risks of using the A2B Mailbox method, let's take a closer look at the structure of a typical communication stack.

Mailbox registers are limited to 4 bytes. Thus, for a mailbox to be used for any meaningful application-layer communication, a transport layer must be implemented on the MCU.

An example of such implementation can be seen in the image below:

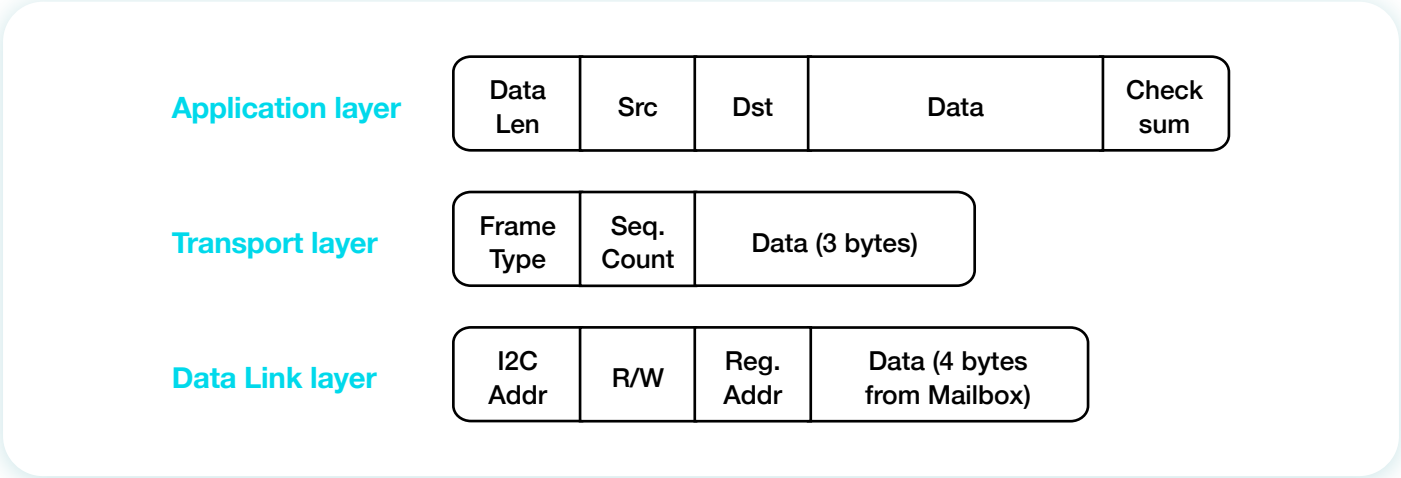


Figure 3. An example of I2C Mailbox communication stack on an MCU

The Data Link layer is the data in the mailboxes themselves. The MCU will implement some sort of Transport layer to defragment Mailbox content into messages longer than 4 bytes. An Application layer could then possibly be added in order to address logical source and destination or to add data integrity checks such as checksum.

It's important to emphasize that such an implementation is on the MCU that is connected to the A2B Node, and is not a feature of the A2B protocol itself. This means that the risks stemming from such an implementation are not inherent to the A2B protocol, but rather are related to the handling of I2C data on the MCU. Any implementation faults are specific to the MCU's software and not to the A2B.

A2B Security Threats

Since A2B allows I2C communication to be performed remotely, it effectively extends the I2C bus beyond the physical confines of a single PCB. This blurs the security boundaries inherent to this protocol and introduces new risks that should be carefully considered.

One of the key security threats related to A2B is the possibility that once an attacker gains control of an A2B node, they could then use that node to compromise other ECUs in the A2B network.

Here are a few examples of potential attack scenarios involving lateral movement.

Lateral Movement Downstream from an A2B Main ECU

Let's consider a case where an attacker gains code execution ability on the Main A2B MCU. In such a case, the attacker can use the A2B bus to attempt to access additional ECUs:

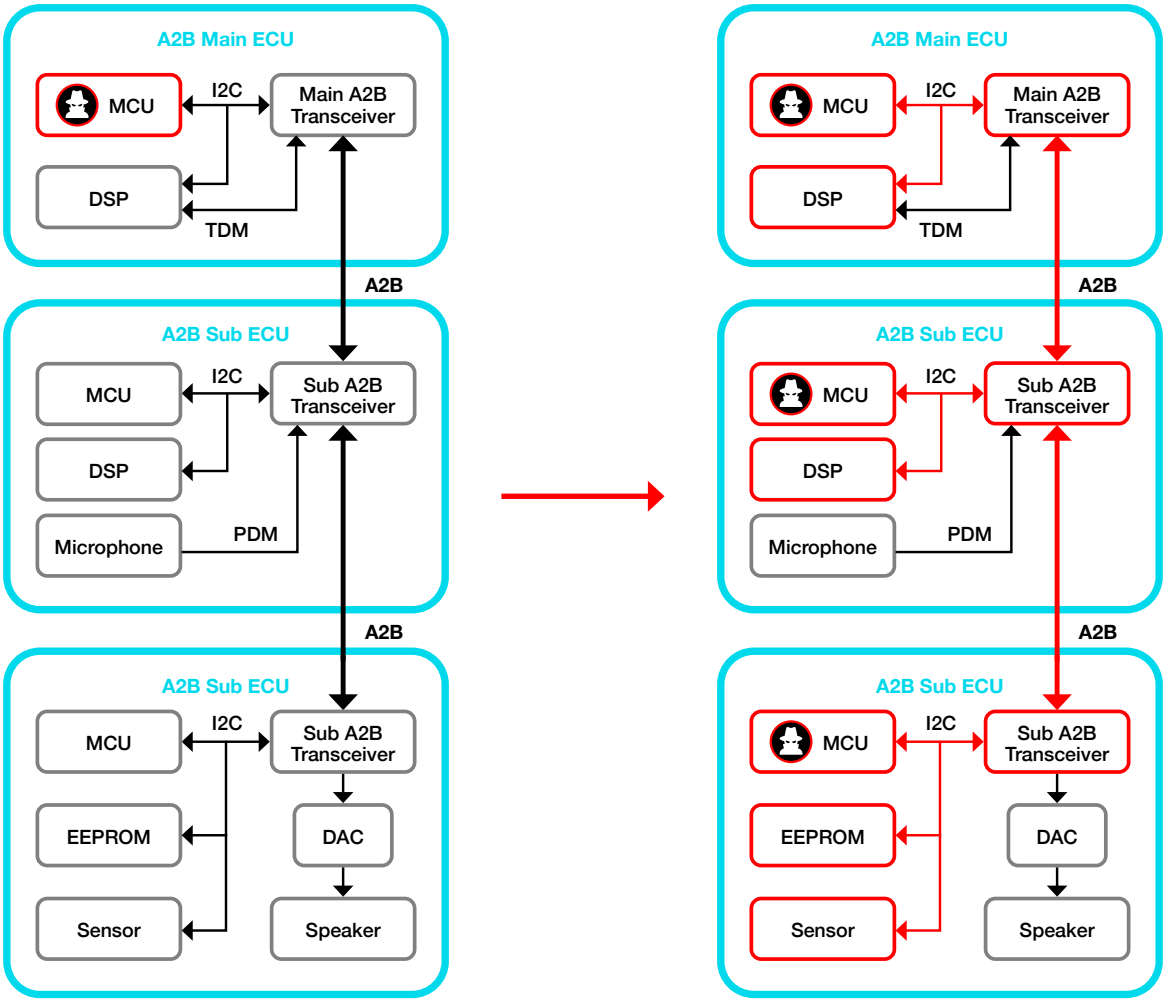
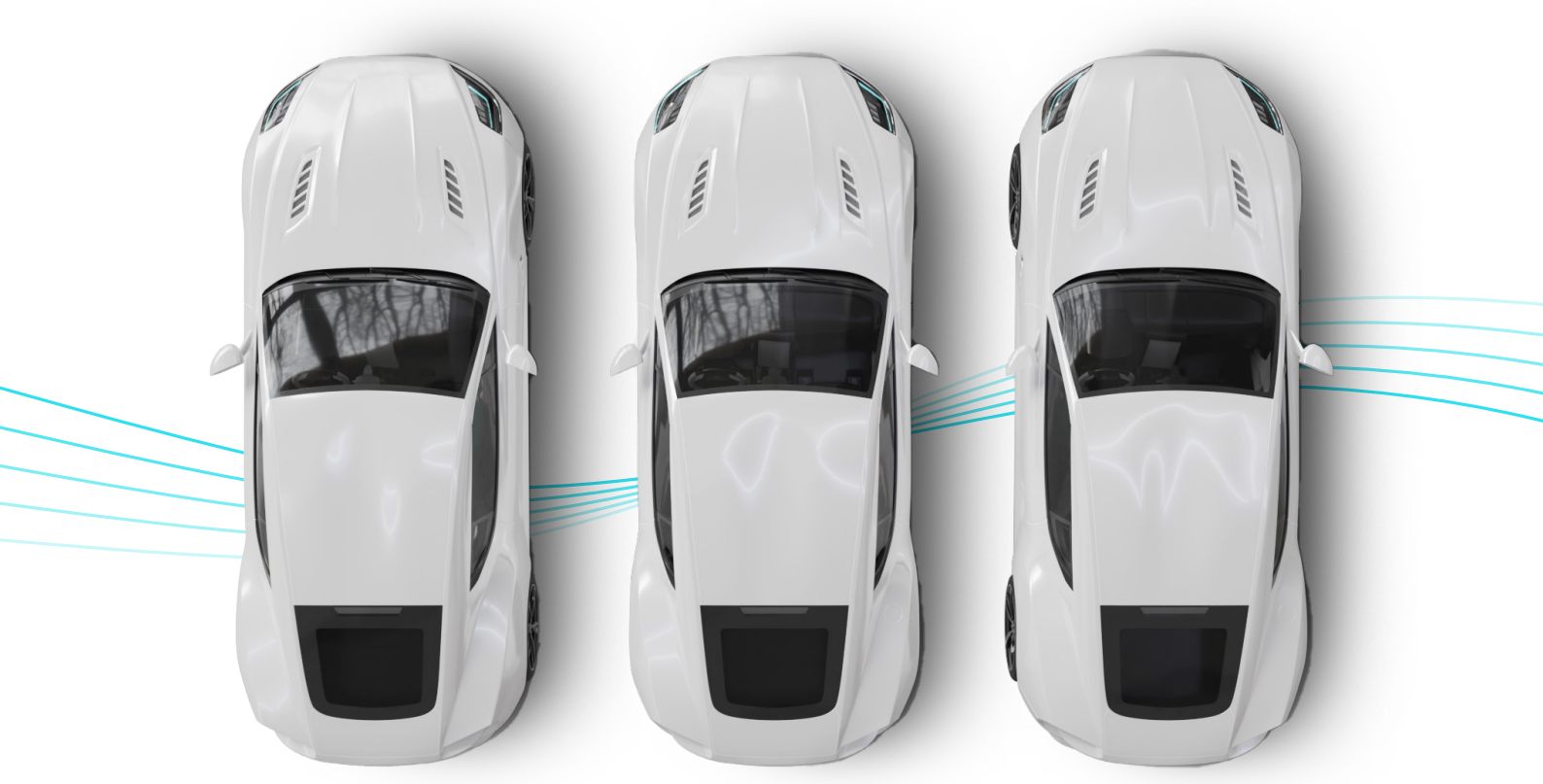


Figure 4. A compromised A2B Main MCU is used to further compromise A2B sub nodes

Possible attack vectors might include the following:

- Since the attacker controls the A2B Main, he can initiate I2C communication to any A2B-Sub on the bus. For example, if any of the A2B-Sub MCUs implement a transport & application layer to communicate using Mailboxes, the attacker could exploit potential vulnerabilities in the implementation of the I2C Mailbox communication stack on connected MCUs to gain **remote code execution on the A2B Sub MCU**.
- Another potential attack vector could be **remote access to other A2B-Sub I2C bus components**. For example, assuming the A2B sub MCU uses the same I2C bus to communicate with the A2B Transceiver as well as other unrelated components such as EEPROMs or sensors, the attacker could communicate with them from remote (e.g., overwrite EEPROM firmware, configure or read sensor data, etc.).
- Additionally, as previously stated, GPIO-Over-Distance allows the A2B Main to **remotely toggle physical GPIOs on the A2B-Sub PCB**. So under certain circumstances (pending PCB layout design of the A2B Sub ECU), it may be possible to maliciously cause an electric shortage on GPIO lines on the A2B Sub, possibly bricking it and **causing permanent DoS** of the ECU.



A2B Subordinate Node Role Switch

As common A2B chips can be configured as both Main and Sub (e.g., AD2428, AD2430W, AD2437), if an attacker has control of an A2B Sub MCU, they can reconfigure the A2B chip as an A2B Main, thus effectively taking control of all downstream A2B nodes.

In such a scenario, the attacker would need to assume control of the Sub A2B DSP in order to reconfigure some A2B control lines that usually originate from the DSP, most notably the SYNC line.

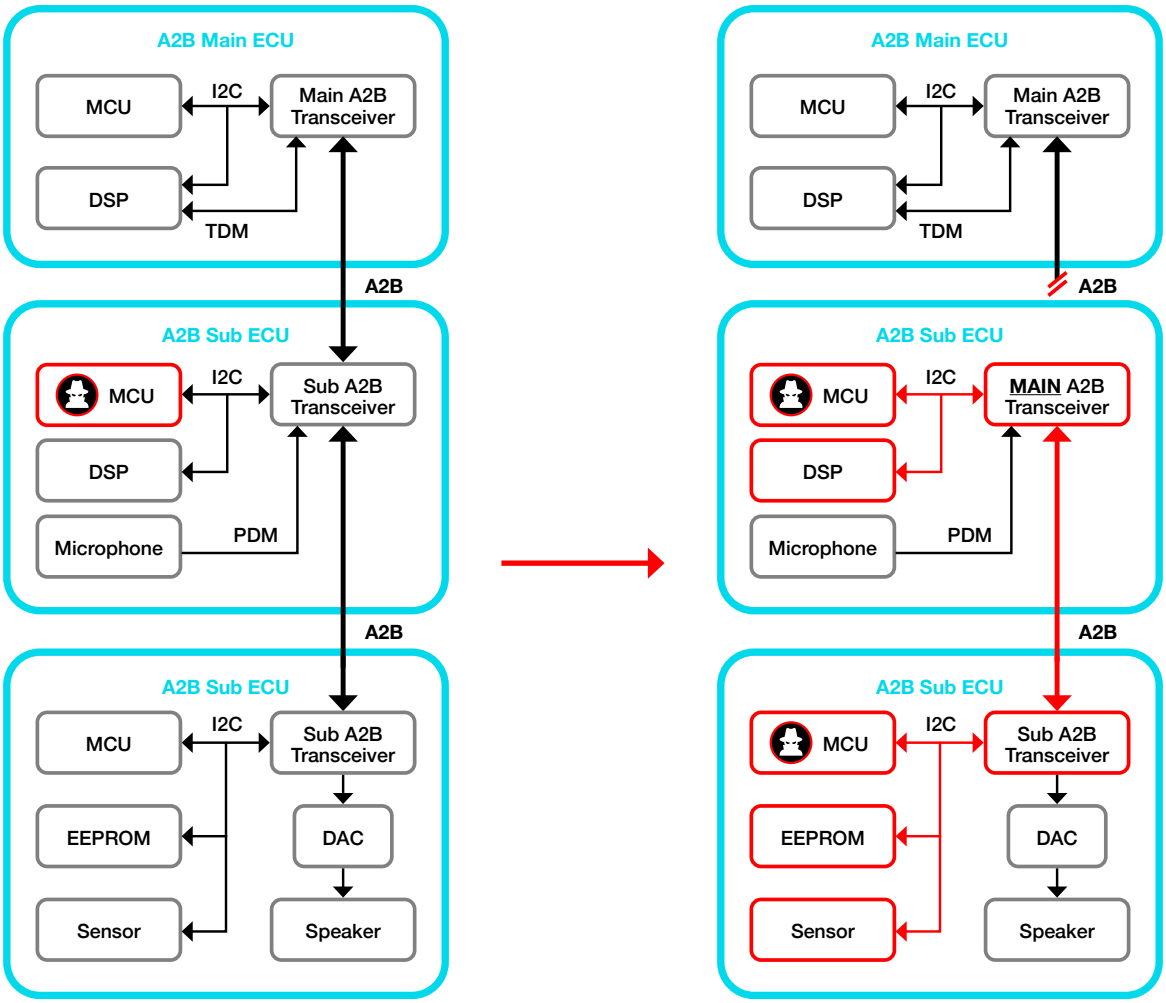


Figure 5. A compromised A2B Sub node MCU is used to compromise other A2B Sub nodes downstream

This will allow the attacker to impersonate an A2B Main, and perform **all attacks detailed in the previous section on all A2B Sub ECUs downstream** from the compromised Sub node.

Lateral Upstream Movement from an A2B Sub ECU

In the scenario where an attacker compromises an A2B Sub node, potential upstream lateral movement should also be taken under consideration.

In case of I2C communication between the A2B Main and the A2B Sub MCUs, the A2B Sub MCU responds to commands initiated by the A2B Main, and these commands are handled on the A2B Main MCU communication stack.

Thus, the attacker could exploit potential vulnerabilities in the A2B Main MCU's code that handles the A2B Sub MCU responses to gain remote code execution on the A2B Main MCU.

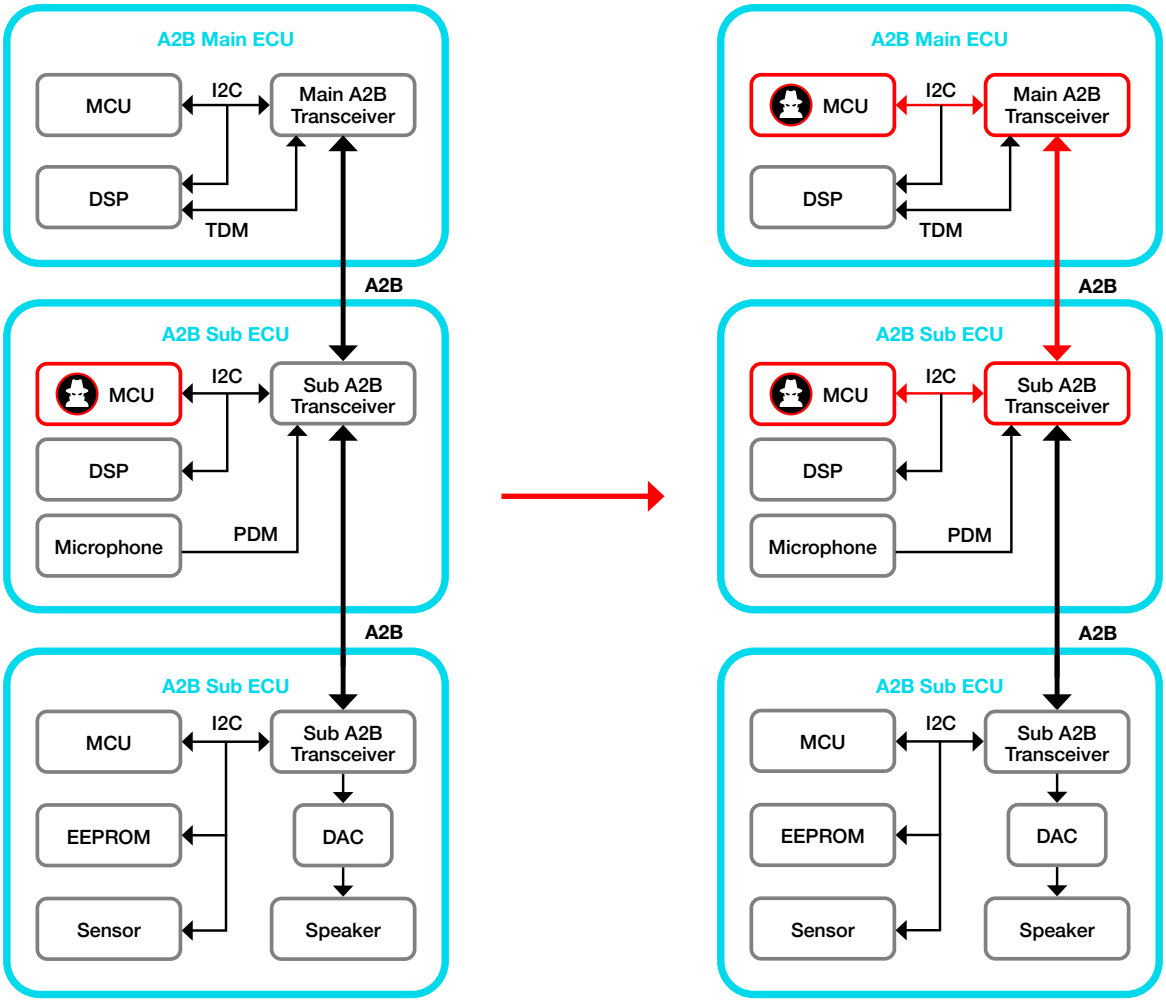


Figure 6. A compromised A2B Sub node MCU is used to compromise the A2B Main Node MCU

Mitigations & Preventive Measures

In order to mitigate the threats described above, several steps can be taken on both the hardware design and the software development levels.

A2B Sub ECU Implementation

- On the hardware design level:
  - Separate the I2C physical bus that is used for A2B communication from other components to prevent remote access in case of a compromised A2B Main
  - Use an A2B chip that is A2B Sub only to prevent an attacker from converting it to an A2B Main
  - Verify that the PCB layout of remotely controlled GPIOs is resistant to modifications of the GPIO direction.
- On the software implementation level:
  - Treat I2C communication coming from the A2B Main as untrusted, and verify that its handling cannot be used to compromise the MCU.
    To mitigate this risk, perform a security code review of the I2C communication stack or alternately perform A2B interface fuzzing to verify no inputs cause unintended behaviour of the system. This can be done using interface fuzzing tools, such as the PlaxidityX AutoTester.

A2B Main ECU Implementation

- Treat I2C data that arrives from any A2B Sub as untrusted and verify that its handling cannot be used to compromise the MCU.
  In order to mitigate this risk, perform a security code review of the I2C communication stack or alternately perform fuzz testing to verify no inputs cause unintended behaviour of the system. This can be done using interface fuzzing tools, such as the PlaxidityX AutoTester.

Summary

While the A2B protocol offers groundbreaking simplicity and efficiency in automotive audio systems, its ability to allow remote I2C communication introduces a new attack vector. Automotive manufacturers should take this threat under consideration as part of overall system security when implementing a communication stack using this interface.

The risks associated with A2B implementations include lateral movement attacks (upstream or downstream on the A2B bus), as well as access to shared resources and remote physical damage.

However, with proper mitigations in place - such as hardware-level protections including isolation of I2C buses and rigorous software testing - these risks can be addressed.

By prioritizing security measures during system design, developers can ensure that A2B's innovative capabilities are realized without compromising vehicle security.

It should be noted that this paper was shared with Analog Devices Product Security Response Team and their comments were taken into consideration and incorporated into the paper prior to its publication.

About PlaxidityX Cyber Security Research and Solutions Department

The PlaxidityX Cyber Security Research and Solutions department is a leader in protecting the automotive industry. With a deep understanding of vehicle architectures, protocols, and standards, we provide comprehensive cybersecurity services to our clients.

Our team, backed by decades of expertise in both cybersecurity and the automotive sector, has partnered with major OEMs and Tier 1s on dozens of penetration testing and research projects. Our goal is to verify and strengthen our customers' cybersecurity posture, helping them meet and exceed key industry regulations like UNR 155 and ISO-21434.

Whether it's a dedicated research project or the deployment of our advanced PlaxidityX products, we deliver the solutions and insights needed to stay ahead of evolving threats and ensure vehicles are secure throughout their lifecycle.

References

1. A2B Solutions by Analog Devices
2. An article about A2B Use Cases by Analog Devices
3. Software over the A2B - How A2B Is Changing the Game for SOTA in Automotive Applications
4. AD242X Reference Manual