# HoliSec

*Holistic Approach to Improve Data Security*

## *AUTOSAR Secure Onboard Communication*
## *Testing SecOC with Various Communication Methodologies*

**Presenter**: Kaushik Naik, Brian Katumba

March 26, 2019

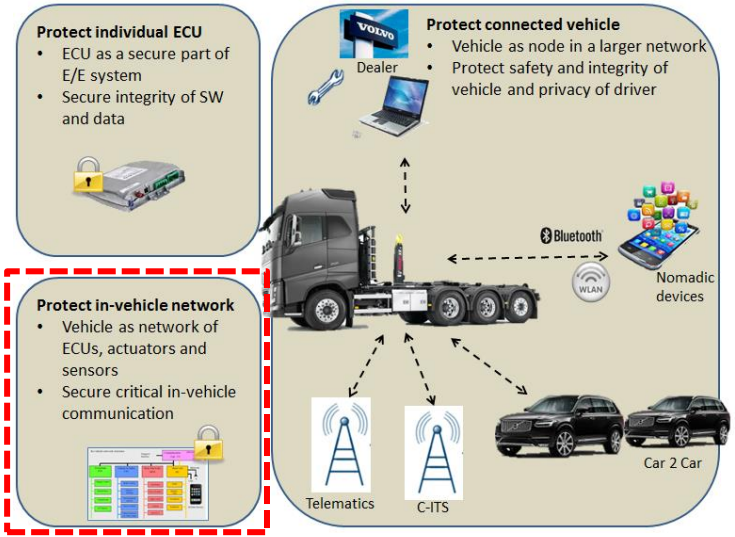
VINNOVA FFI VOLVO VOLVO ARC CORE CHALMERS ASSURED SECURITY CONSULTANTS RI.SE Research Institutes of Sweden

# Outline

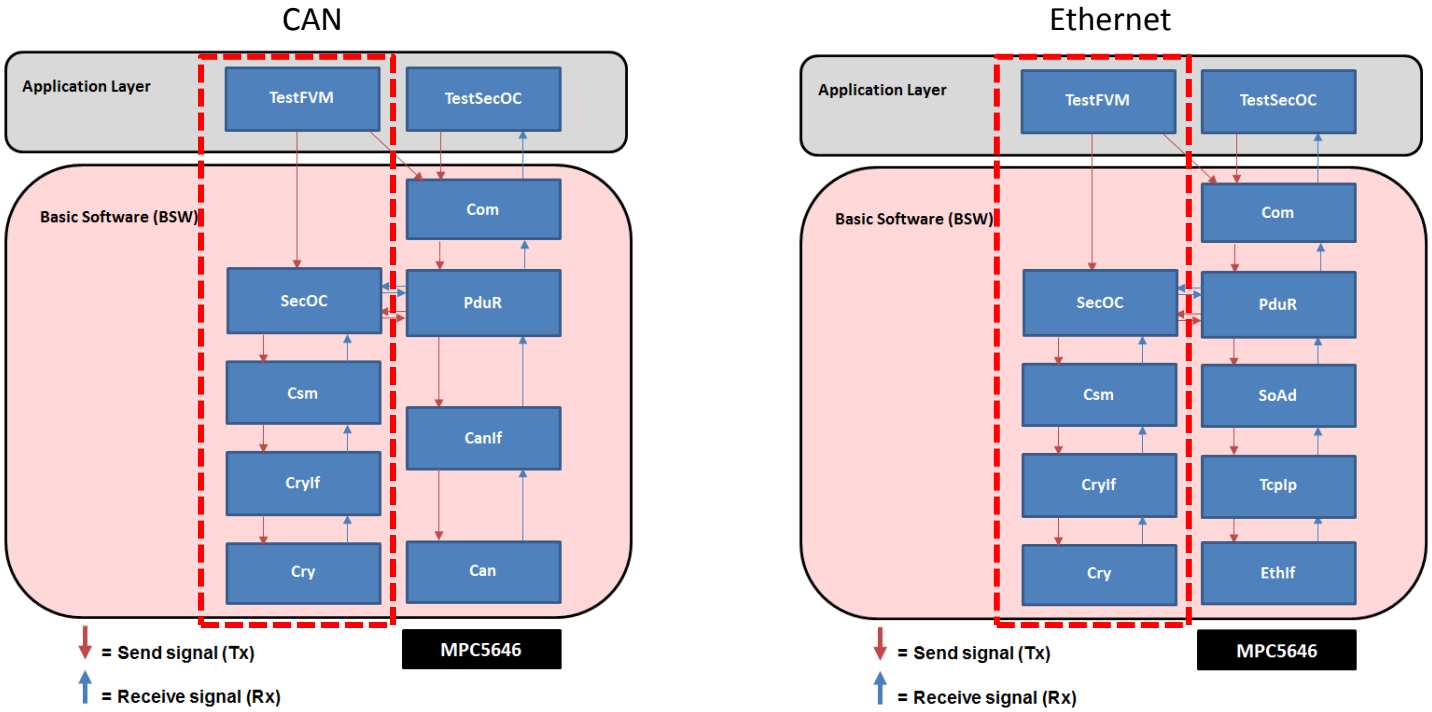
- Objective
- AUTOSAR BSW Message Flow with SecOC and FVM
- Message Configurations
- Freshness Value Manager (FVM)
- Practical Implementation of SecOC
- SecOC Evaluations

# Objective

- This work is part of WP3.2.
- The objective is to implement AUTOSAR SecOC to the different communication setups: CAN and Ethernet.
- Freshness Value Manager (FVM) implementation to support AUTOSAR SecOC for synchronization between the Master and Slave ECUs.
- Evaluating the AUTOSAR SecOC module:
  - Message Tampering
  - Re-Synchronization
  - Performance (with a real truck use case : Brake light function)

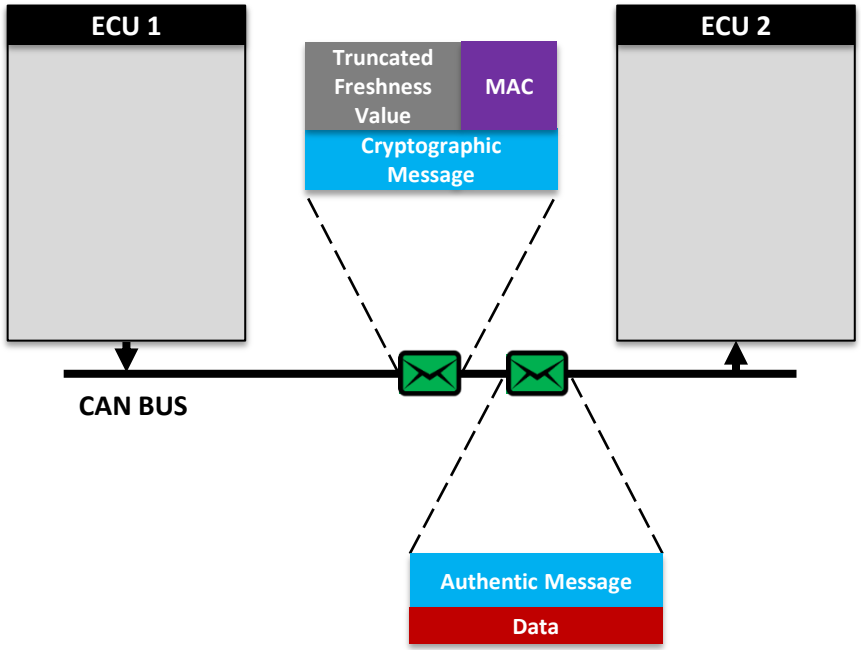


# AUTOSAR BSW Message Flow with SecOC and FVM

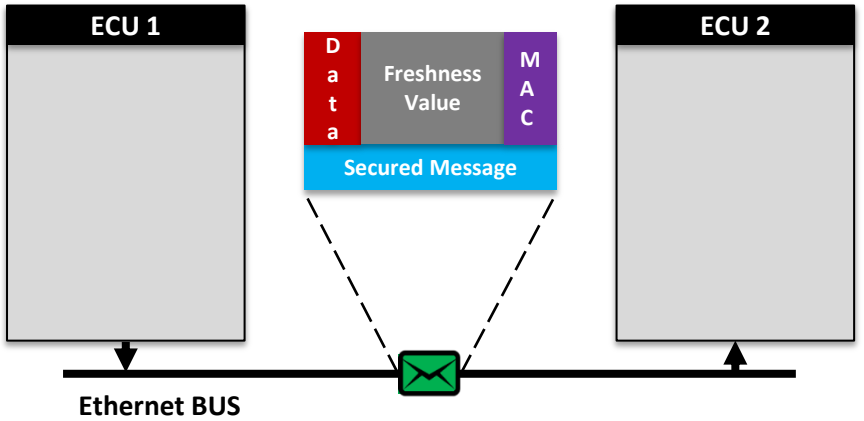


# Message Configurations

Configuration for CAN communication

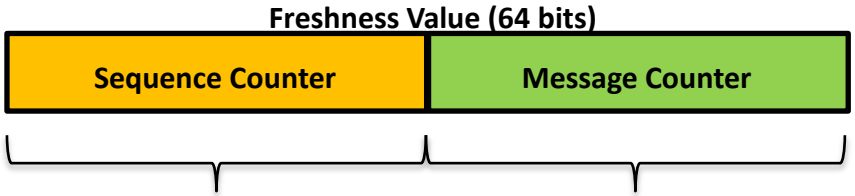



Configuration for ETH communication



# Frehness Value Manager (FVM)

- Freshness handling is a monotonic counter to prevent replay attacks.
- Freshness implementation is left to OEMs for their specific use.
- Freshness Value Manager (FVM) supports SecOC module to have freshness counters for each message been sent and also provides Sync Messages to have synchronization between ECUs.



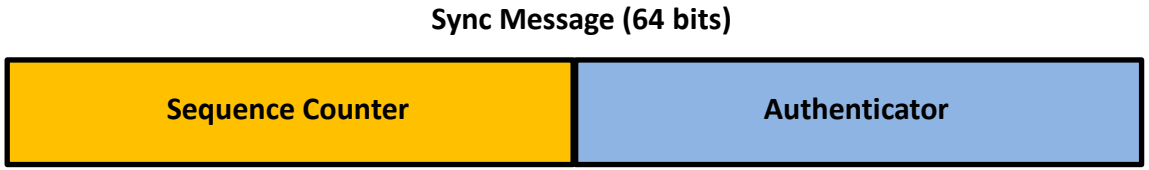

- **One sequence counter per sender node**
- **Incremented once every time the SyncMessage is transmitted.**
- **The size of the Sequence Counter is taken as 32 bits.**

- **One message counter per secured PDU sent by the sender node**
- **Incremented once per each ECU sent.**
- **It is set to 0 every time the Sequence Counter has been incremented.**
- **The size of the Message Counter is taken as 32 bits.**

# Frehness Value Manager (FVM)

## Sync Message

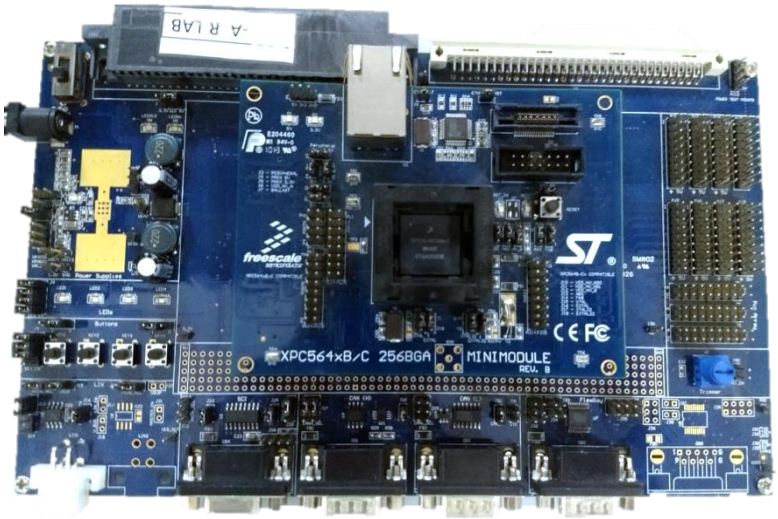
- The purpose of the SyncMessage is for the Sender and Receiver(s) of data to be in sync.
- The periodicity of the SyncMessage state the maximum time that the communication may be out of sync from freshness perspective in the absence of faults.




- The FVM module in the Sender is responsible for generating and sending the SyncMessage at the **start-up, periodically** and **on-demand** from receiver ECUs.

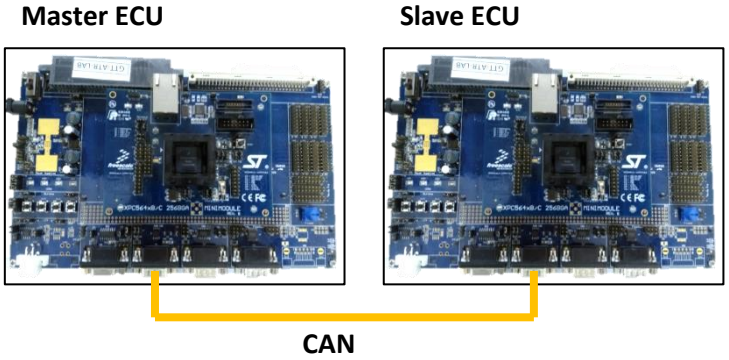
# Practical Implementation of SecOC: Test Rig-Hardware Specification

- Freescale MPC5646C Evaluation board.
- Freescale MPC5646c supports CSE (Cryptographic Service Engine) which is a peripheral module that implements the security functions.
- The CSE (Cryptographic Service Engine) has the following features:
  - Secure storage for cryptographic keys
  - AES-128 encryption and decryption
  - AES-128 CMAC (message authentication code) authentication
  - Random number generation
  - Secure boot mode

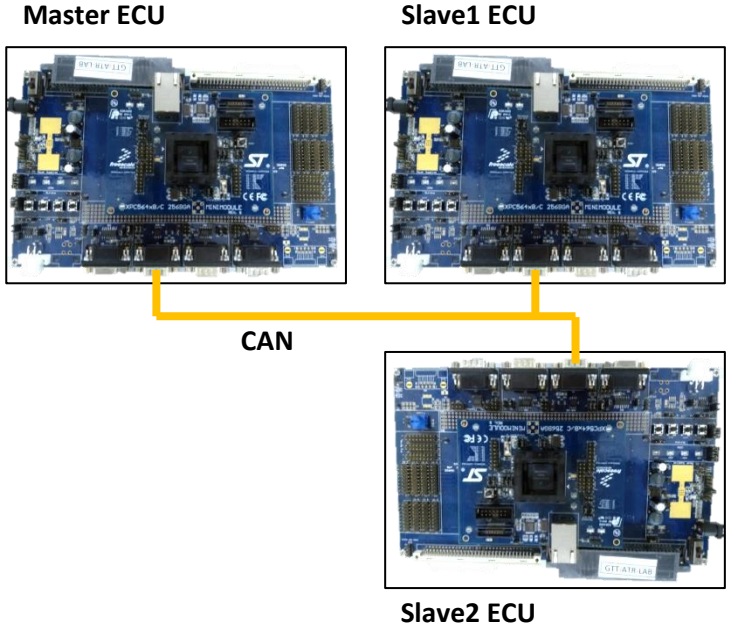


# Practical Implementation of SecOC: Scenarios

**Scenario 1** is a single sender and receiver setup, where both communicating peers communicate over a CAN bus.

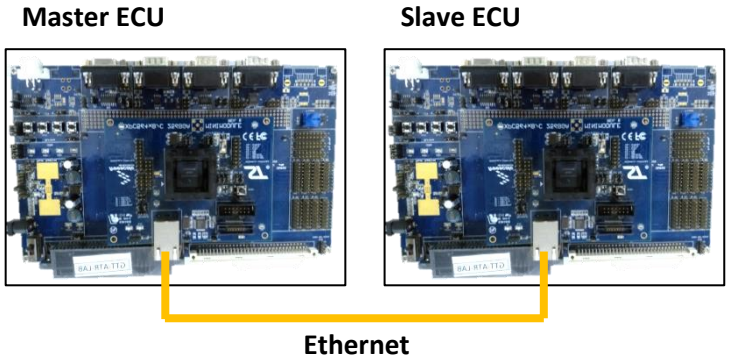



**Scenario 2** is a single sender and two receiver setup, where all communicating peers communicate over a CAN bus.

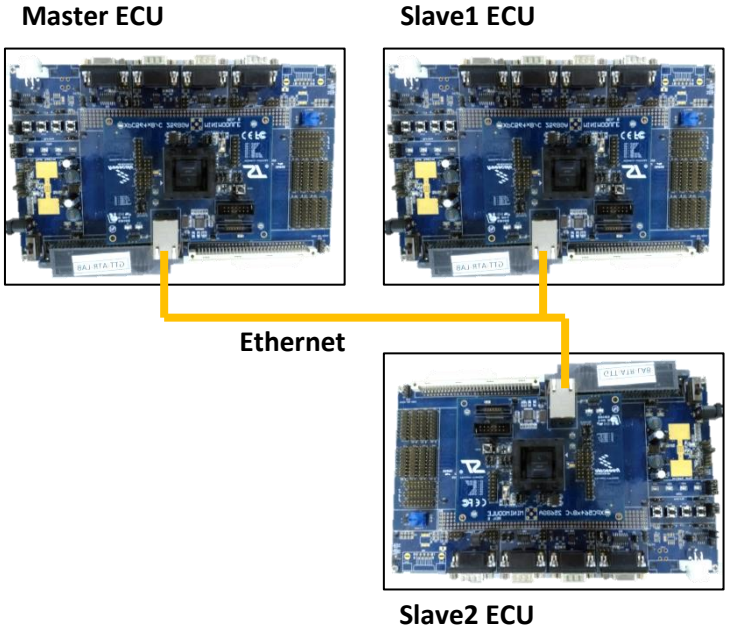


# Practical Implementation of SecOC: Scenarios

**Scenario 3** is a single sender and receiver setup, where both communicating peers communicate over a Ethernet bus.



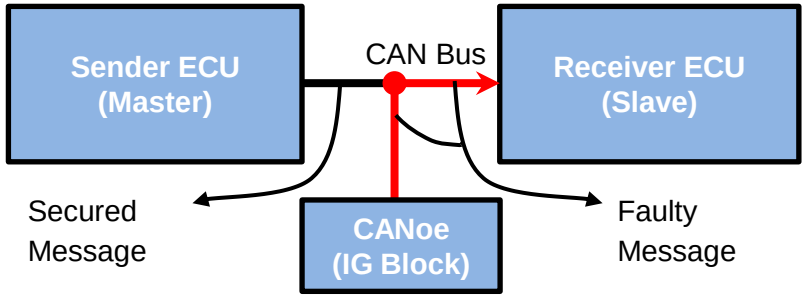

**Scenario 4** is a single sender and two receiver setup, where all communicating peers communicate over a Ethernet bus.



# SecOC Evaluation - Message Tampering

- **Message Tampering**
  - In this test we are trying to evaluate how the SecOC module of the receiver ECU responds to a faulty message.
  - The actual message sent from the sender ECU was tampered and a faulty message was sent with same CAN Id using CANoe.

**Test Observation:** The MAC verification fails at the SecOC module and the data is not sent to the upper layers.




IG – Interactive Generator

# SecOC Evaluation - Re-Synchronization

- **Receiver ECU reset test**
  - In this test we reset the receiver ECU so that it goes out of sync with the Sender ECU.
  - This is to check the Sync Request feature.

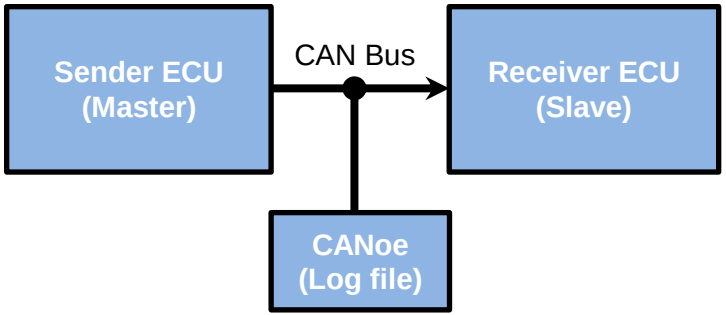
**Test Observation:**

- The Receiver ECU detects a sync lost and sends a sync request.
- The sender ECU receives the sync request and immediately sends a sync message.
- Then the receiver ECU regains sync.

**Test Result:**

| Sl.No. | Time | CAN ID | Data |
|---|---|---|---|
| 1. | 4.098995 | 18300314x | 9C 00 00 00 D3 3C 80 8B |
| 2. | 4.299039 | 18300314x | 9D 00 00 00 65 25 EC C8 |
| 3. | 4.310876 | 18300315x | 01 00 00 00 00 00 00 00 |
| 4. | 4.328947 | 18300314x | 9E 00 00 00 A8 9D 8D 85 |
| 5. | 4.499043 | 18300314x | 9F 00 00 00 3C 22 FB 7E |

18msec

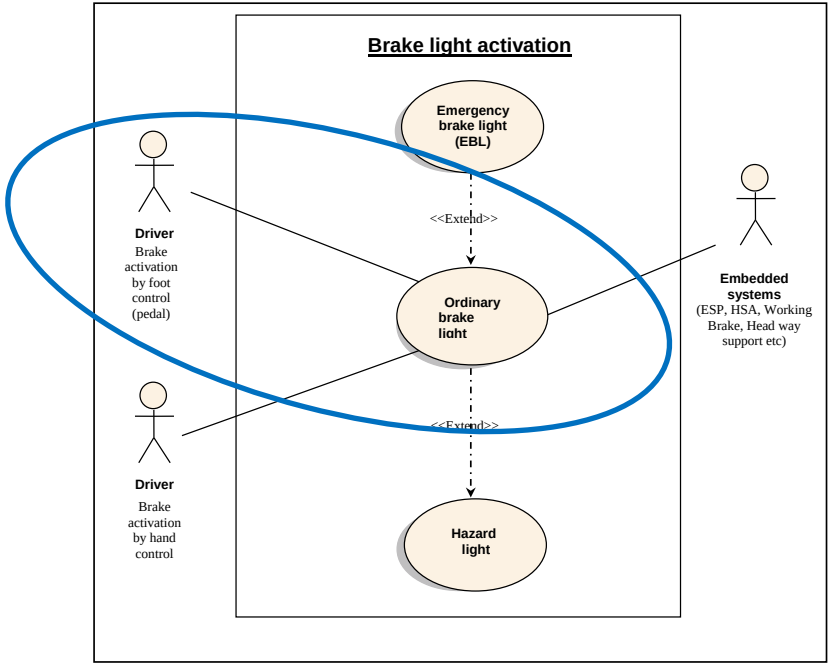



**<-- Sync Request from receiver**

**<-- Sync Message sent after a sync request**

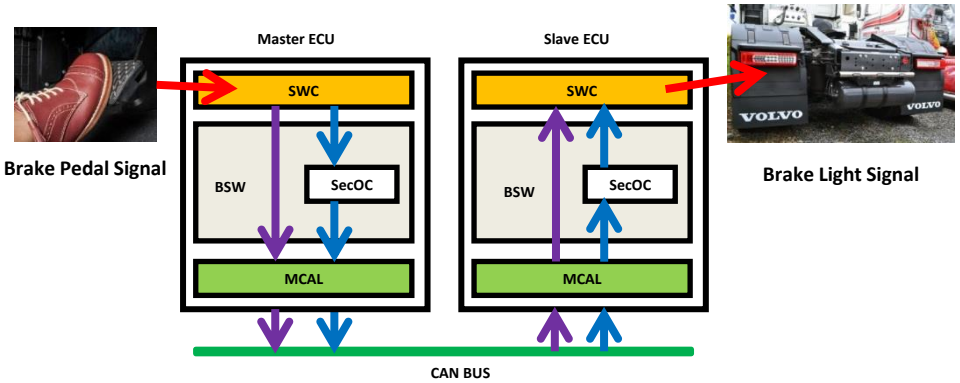
**Sync Message ID - 18300314x**
**Sync Request ID - 18300315x**

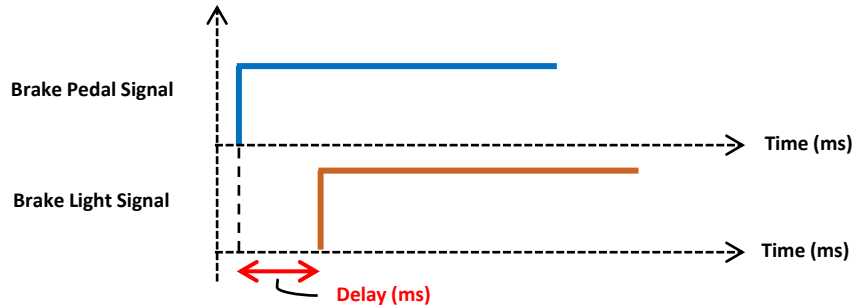
# SecOC Evaluation- Performance : Brake Light Function




- Brake Light can be activated by: Foot control, hand control and Embedded Systems such as ESP (Electronic Stability program), HAS (Hill Start Aid), working brake headway support etc.
- For the purpose of this work, we focus on Brake activation by Brake Pedal: Depressing the brake pedal shall activate the brake light with a periodicity of **10ms** and within **100ms**.
- **Brake light is a high-priority function**, it is important that the light is activated with a **minimum** of delay time from brake activation.

# SecOC Evaluation- Performance : Brake Light Function




- The sender ECU will send the brake pedal signal through CAN to another Receiver ECU which will glow the brake light.. Periodicity: 10, 20, 50, 100
- The push button on the sender ECU replicates a brake pedal and a LED on the receiver ECU replicates the brake light.
- When a push button is pressed in the sender ECU, the signal will be sent to receiver ECU which will glow a LED.
- The time latency with and without data security are measured using Oscilloscope

# SecOC Evaluation- Performance : Brake Light Function

- **Message latency measurement:**

| Periodicity → | 10ms | 20ms | 50ms | 100ms |
|---|---|---|---|---|
| Secured Message (msec) → | 12 | 12 | 48 | 48 |
| Non-Secured Message (msec) → | 12 | 12 | 48 | 48 |

RQ: Periodicity of 10ms and time from brake pedal signal to the brake light activation at the receiver ECU should be less than 100msec

- Considering the requirements of the brake Light function, with or without Security (SecOC), the results did not exceed the time bounds of 100ms hence making SecOC an ideal security solution for onboard communication of the light function
- Note that the brake signal is routed via multiple ECUs and the signal periodicity could change in the in-between nodes.

# HoliSec

*Holistic Approach to Improve Data Security*

## Questions?

# Live Demo During the Break!