Integration and Proving Ground Evaluation of a Connected Highway Pilot System

Author, co-author (Do NOT enter this information. It will be pulled from participant tab in MyTechZone)

Affiliation (Do NOT enter this information. It will be pulled from participant tab in MyTechZone)

Abstract

Highway Pilot (HWP) systems, classified as SAE Level 3 Automated Driving Systems (ADS), represent a potential advancement for safer and more efficient highway drives. In this work, the development of a connected HWP prototype is presented. The HWP system is deployed in a real test vehicle and designed to operate autonomously in highway environments. The implementation presented in this paper covers the complete setup of the vehicle platform, including sensor selection and placement, hardware integration and communication interfaces for both autonomous functionality and Vehicle-to-Everything (V2X) connectivity. The software architecture follows a modular design, composed of modules for perception, decision-making and motion control to operate in real-time. The prototype integrates Vehicle-to-Vehicle (V2V) communication, such as Cooperative Awareness Messages (CAM), to enhance situational awareness and improve the overall system behaviour. The modular structure allows new functionalities to be developed and integrated into the same platform, so it becomes a beta testing platform for the early-stage experimentation of innovative features. To validate the prototype, execution of scenario-based testing on a proving ground is used, which is part of the Safety Assurance Framework (SAF), which provides structured methods to help define test scenarios to cover the system's Operational Design Domain (ODD). Experimental results demonstrate the prototype's ability to operate in varied highway scenarios.

Introduction

In this era of increasing technology, autonomous driving is becoming a force to be reckoned for what appears to be the future of mobility. The major OEMs are currently deploying Advanced Driving Assistance Systems (ADAS) that can cover great varieties of situations, but so far, in most of these systems, responsibility still lies with the driver, which on the SAE driving automation scale is known as level 2 systems [1]. All these new trends are evolving rapidly, and new ones are emerging to increase the benefits that these technologies can bring to vehicle safety. The increased reliability and safety of these systems will ultimately result in responsibility being transferred to the vehicle itself and only require the driver to take control in specific situations, i.e. moving to level 3 on the SAE driving automation scale. This milestone could mark a paradigm shift in what we understand by personal mobility today, allowing monotonous journeys on high-speed roads with the risk of distraction and loss of concentration of the driver to be replaced by safe and comfortable journeys.

In this regard, it is necessary to create development environments for these functionalities to occur effectively and safely, where new implementations and improved concepts can be tested in situations that are close to reality.

On the one hand, there is the entire virtual environment, where functionalities can be developed, tested and validated at relatively low cost in a wide variety of complex situations. However, transferring these simulation tests to the real world can sometimes prove to be a major challenge, as there are particularities and small details that may be overlooked in the models used in virtualization, preventing the discovery of implications for the behaviour and safety of the functionality itself. For this reason, it is necessary to have test instances at all levels in order to build a reliable validation environment that ensures that a system is secure in its entirety.

For all these reasons, it is essential to develop early prototypes, where all these new features and concepts can be evaluated in real environments, enabling the testing of sensors, perception, and control algorithms in real conditions without needing mass production with all the implications that this entails.

In this work we present a connected Highway Pilot (HWP) prototype, a production vehicle transformed into a research prototype instrumented with sensors and equipped with computing devices to deploy the new concepts related to highway pilot systems, including V2V communications to investigate potential benefits this technology can bring to enhance situational awareness and cooperative driving in the field of automated driving. It serves as a modular development lab: using distributed hardware and software, it enables rapid prototyping of new algorithms without the need for excessive resources.

The development and validation of the HWP prototype has been carried out within the Sunrise project [2], which aims to accelerate a safe deployment of Cooperative, connected and automated mobility (CCAM) systems. In this project, a harmonized and scalable Safety Assurance Framework (SAF) is developed to stablish a method and toolchain for virtual and physical evaluation of CCAM systems.

We detail the vehicle build-out (sensor set-up, computing modules, communications integration), the modular software architecture (perception, localization, planning, control, communication), and the V2V connectivity using, for instance, the Cooperative Awareness Messages (CAM) from the European Telecommunications Standards Institute (ETSI) [3]. We then describe a practical example on how the SAF [2] would be applied to the scenario-based testing execution on proving grounds to evaluate the safety of the system's behaviour on its Operational Design Domain (ODD).

Prototype Vehicle Platform

The prototype vehicle platform is named CAVKit, referring to the fact that it serves as a development and testing kit for new Connected and Automated Vehicle (CAV) concepts. This platform consists of various systems fully integrated, including the perception, planning and control systems, which are responsible for perceiving the operational environment and controlling the CAVKit's movement. The integration is intentionally modular: new sensors (e.g. an additional radar) can be added in the system, and any compute node can be upgraded. This design turns the vehicle into a mobile testbed for new features.

The base vehicle is a production Toyota RAV4 from 2022. This vehicle does not have advanced autonomous driving systems, but it does have basic ADAS functionalities (AEB, LKA and ACC; i.e. SAE level 1). The fact that it has a system capable of controlling the dynamic behaviour of the vehicle already present in the vehicle's Electrical/Electronic (E/E) architecture makes it possible to command the vehicle movement via Controller Area Network (CAN) messages. Thus, by integrating an external CAN gateway between the Electronic Control Units (ECU) responsible for acceleration/deceleration and steering angle requests and the rest of the vehicle, movement can be controlled by manipulating these signals, allowing to modify the behaviour following commands to these manipulated signals other than those required by the ADAS ECU from the vehicle itself.

To achieve this, it is necessary to transmit all bidirectional communication in real time on this external CAN gateway without adding delays that could cause the vehicle to detect faults and generate Data Trouble Codes (DTC) that affect its operation. In addition, messages of interest are filtered, and their data is modified in order to manipulate the signals according to the coding that the vehicle is expecting.

To integrate these systems into the vehicle, an instrumentation rack has been installed in the trunk (see Figure 2). This rack is powered directly by the vehicle's electrical system through DC/DC devices and is capable of supplying power to all components without requiring any external power source. The complete schema of the device and their connections can be seen in Figure 1.

In terms of onboard instrumentation, the CAVKit vehicle has been equipped with a Speedgoat real-time target machine (Mobile M3 version), which acts as the central processing unit of the system. This unit has the onboard decision-making and control algorithms deployed in it.

For perception system, a variety of sensors have been integrated to enable accurate monitoring of the vehicle's surroundings as well as to support localization tasks, as can be seen in Figure 3. The sensor set includes two smart sensors which output directly the objects list and lines detected, which are a Mobileye Aftermarket 6 and a Continental ARS548 radar. Additionally, other sensors that output raw data are instrumented at the top of the vehicle using aluminium Bosch profiles to develop and deploy computer vision algorithms for further testing. These include 3 BASLER a2A1920-51gcPRO cameras pointing to the front (one centred and two in the corners), and an Ouster OS2-128 360º lidar.

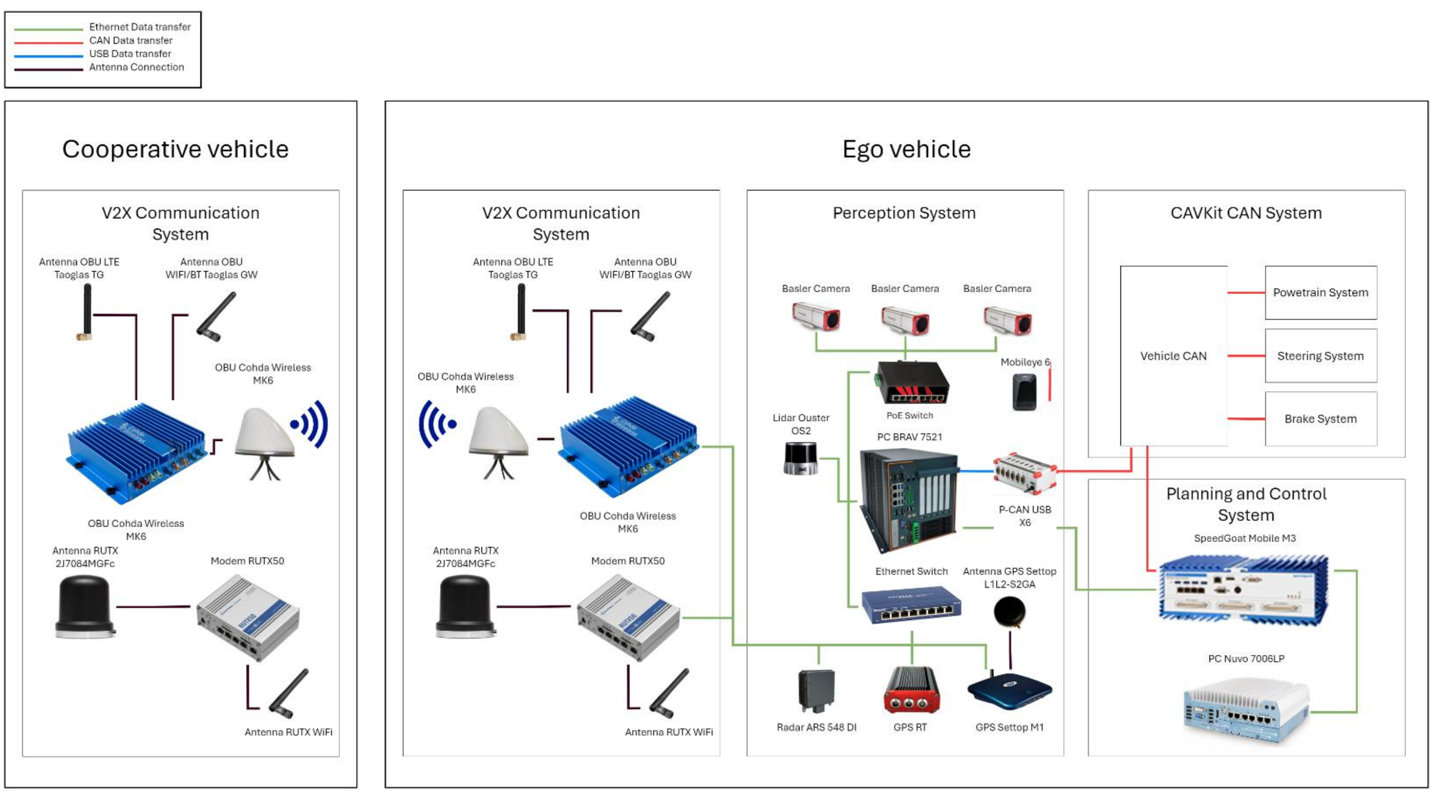

Figure 1 Architecture and communication interfaces for the devices instrumented in the CAVKit prototype vehicle (right) and the other vehicles used for testing (left)

For positioning and motion estimation, the setup includes an Inertial Measurement Unit (IMU) and a Global Navigation Satellite System (GNSS) receiver (Settop M1). Additionally, for reporting purposes, a differential GPS with Real Time Kinematic (RTK) module from OxTS (RT3000) is used to extract ground truth data, which is fundamental for evaluating the performance of the functionalities during the analysis of the test results.

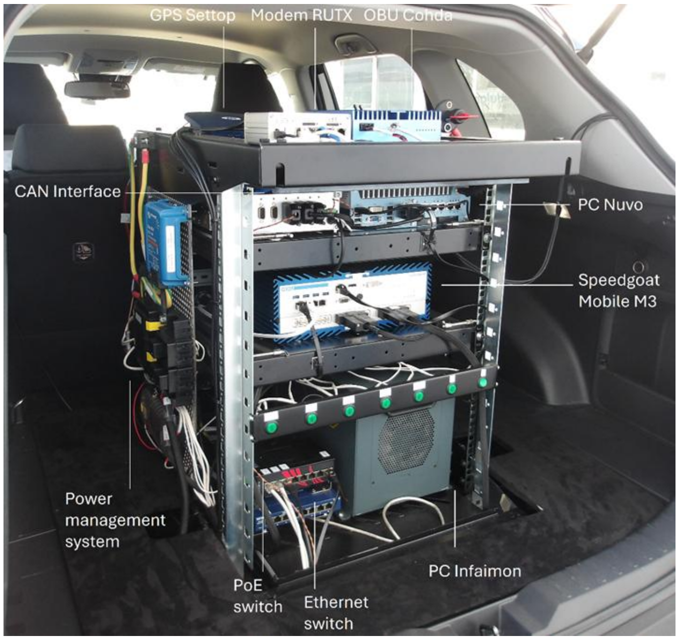

Figure 2 CAVKit rack with all the devices placed in the trunk

All the perception, sensor fusion and localization algorithms are deployed in a Linux PC running on a ROS2 environment for a rapid deployment of such algorithms and it is connected via an Ethernet interface to the Speedgoat real-time target machine to send the resultant information which is used for the planning and control part of the HWP.

Finally, for the V2V connectivity stack, the prototype has been equipped with a Cohda On-Board Unit (OBU), which allows the communication with other vehicle using the state-of-the-art ETSI messages and a Modem RUTX with a 5G SIM card which provides high-speed Wi-Fi connectivity to both the vehicle’s occupants and onboard systems.

Figure 3 Exterior image of the CAVKit vehicle with all the sensors available for the perception and localization systems

Software Architecture

As for the general concept developed to validate the prototype, the software architecture is composed by modules as represented in Figure 4. Speedgoat device hosts autogenerated C code from Simulink models that include the logic and control of the HWP functionality, as well as the external CAN gateway to interact with in-vehicle's communication buses and therefore control the vehicle movement. In parallel, and in a modular approach, there are the perception and connectivity modules, which serve as inputs for the functionality, enabling a precise response from the connected HWP system during the operation. Bidirectional communication messages are transmitted via the User Datagram Protocol (UDP), ensuring coordinated operation of the onboard devices without significant delays. These interactions facilitate V2V communications, which in turn enhance the sensor fusion algorithms.

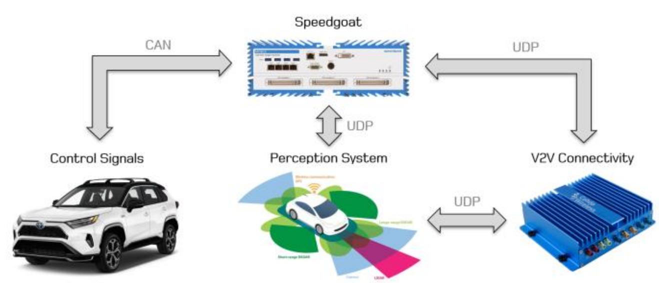

Figure 4 Diagram of the modular systems and the communication protocols linking them

CAN Gateway (Simulink models)

To enable autonomous control over the CAVKit vehicle without interfering with its original electronic systems or generating detectable failures, a strategic intervention has been carried out at the level of the vehicle's CAN communication bus. Specifically, a critical cable within the communication path has been physically interrupted, allowing the software architecture of the prototype to effectively "close the loop" virtually, as represented in Figure 5. This intervention enables the system to act as an external gateway, a transparent node within the vehicle's E/E architecture that maintains the original behaviour of the production system while adding the ability to monitor and override specific control signals. By virtually reconnecting the communication path through our software, the vehicle's internal control units continue to function as expected, without detecting the external intervention or any deviation from their standard operation.

This setup allows the developed system to both read and inject messages on the internal vehicle network. As a result, key state variables such as vehicle speed, longitudinal and lateral acceleration, steering angle, yaw rate, and ACC active flag status can be accurately monitored in real time. In addition, the system is capable of injecting control commands, specifically acceleration requests and steering angle requests, which are interpreted by the vehicle as if they were generated by the ADAS ECU from the manufacturer, enabling the commanded autonomous control over longitudinal and lateral dynamics. To ensure consistency and avoid diagnostic errors, all communication integrity mechanisms, including Cyclic Redundancy Check (CRC) calculations and other embedded verification methods used in the vehicle CAN communication, are carefully replicated and adapted according to the manipulated messages. This approach provides a robust method for enabling automated control in a production vehicle environment while preserving its original safety mechanisms and validated systems.

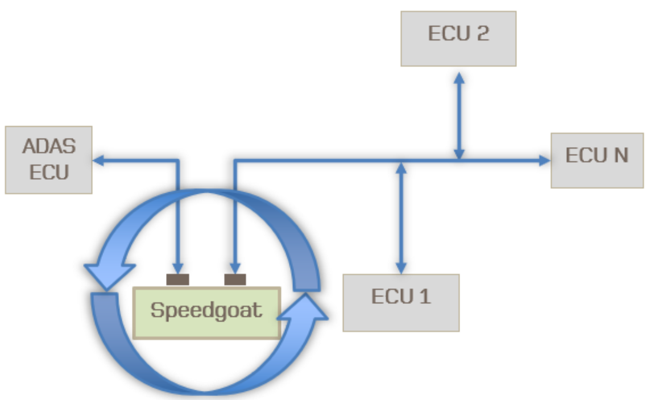

Figure 5 Representation of the concept of the external gateway placed in the vehicle CAN bus

HWP logic and control (Simulink models)

The connected Highway Pilot (HWP) logic and control algorithms have been implemented using a modular approach, where each function is encapsulated in a dedicated Software Component (SWC). The development of such algorithms has been carried out in MATLAB/Simulink, taking advantage of model-based design for faster iteration and easier debugging. After an initial basic validation in simulation, the models are converted into C code via Simulink Coder and deployed on a Speedgoat real-time target machine. Running the control logic on dedicated real-time hardware allows the system to meet strict timing requirements (hard real-time), a very important aspect for safety-critical automotive applications. In this setup, the control loop is executed at 1 kHz, enabling a highly responsive interface with the vehicle's actuators via its communications, such as steering and powertrain control signals.

The HWP model is organised into four main functional blocks: information, judgement, state machine and control. The information layer collects data from different sources, including the vehicle CAN bus, the perception stack, and the connectivity module, and applies the necessary filtering, conversions and processing. With these processed inputs, the judgement module identifies relevant targets, determines the lane currently being travelled, and defines a suitable trajectory. The state machine manages operational modes, taking care of transitions between manual and automated control, as well as managing fallback states in case of detected faults. The control block then translates the planned manoeuvre into steering and longitudinal acceleration commands. In parallel, an Autonomous Emergency Braking (AEB) continuously monitors the environment independently. This safety function is only triggered when high-risk situations appear and can override the main controller to apply maximum braking if a collision is imminent.

The integration between the HWP logic and the rest of subsystems is handled by dedicated input/output adaptation layers. These modules decode and synchronise signals between the HWP and other vehicle systems, including the perception stack, the V2X communication module, and the in-vehicle CAN network. Typical tasks in these layers include scaling and unit conversion, aligning time stamps, and coding or decoding messages (such as CAN or UDP messages) so that all systems can exchange data consistently and without loss of fidelity.

The decision-making of the HWP relies on a fused perception of the surrounding environment, combining information from on-board sensors and V2V communications. This enhanced perception enables the HWP to perform the driving task safely beyond the conventional lane centring and adaptive speed control. For example, if a CAM from another vehicle reports a slower car ahead in the same lane, the system can anticipate and initiate a gentle deceleration before the obstacle becomes visible to the sensors. The planner generates a reference trajectory with a target speed profile. The HWP operation is coordinated through a behaviour architecture that blends normal driving logic with fault-response states and handover management when driver intervention is required.

The motion execution layer follows the planned path and speed adapting the control signals in real time. Regarding the lateral control, it is implemented using a Linear Quadratic Regulator (LQR) [4], which minimises path-tracking error, while with regards to the longitudinal control, it is handled by a PID controller tuned to follow the desired speed profile [5]. These controllers operate at a high update rate and make continuous use of feedback from the steering and velocity sensors to correct and adapt to deviations. The implemented architecture enables the vehicle to respond smoothly and predictably to both the changes in traffic surroundings and to the driver's requests.

Perception and localization (ROS nodes)

As mentioned earlier in this paper, the perception and localization algorithms have been built upon a local ROS2 network in charge of the data acquisition and processing. Each sensor has its unique processing pipeline that generates detections in a unified format, which is later merged using a sensor fusion algorithm based on an Extended Kalman Filter (EKF) [6].

We can differentiate between two types of sensors. On the one hand, smart sensors only require adapting the proprietary driver to forward information into ROS2 using a common data format, being the case of the Mobileye smart camera or the ARS548 radar. On the other hand, classic sensors forward raw data, which is later processed by state-of-the-art detection algorithms to provide obstacle information. In the current configuration of the CAVKit this architecture is used with camera and lidar measurements.

Each sensor estimates the position, velocity, heading and bounding box of dynamic and static obstacles in the environment. Then detections and their estimated covariances are forwarded to the EKF, allowing to optimally combine individual estimates with a motion model to generate an accurate detection of the obstacle.

The modular nature of the system allows enabling or disabling each sensor pipeline seamlessly, allowing to adapt the platform to the specific conditions of the case study being conducted. This architecture allows extracting both road markings and surrounding vehicle information.

To integrate V2V messages in the perception system, we simply convert the data received in the CAM message to our unified data format, including the covariance matrix, and consider it as an additional sensor that can be integrated seamlessly in the sensor fusion pipeline [7].

In this study case, we relied solely on the data provided by a RT3000 module for localization, which provided a centimetre level accuracy, as we have no signal artifacts in IDIADA proving grounds that could degrade the performance of the system. However, it's well known that these systems are susceptible to performance losses in certain conditions, such as around high buildings or when traversing a tunnel. For this reason, a sensor fusion strategy is also available to localise the vehicle by collecting sensor data from a GNSS receiver, an IMU and wheel odometry to be processed by an EKF.

Finally, both localization and perception modules have been combined with a map-based approach. Using a priori knowledge of the area traversed, the information of the perception module can be further refined, not only accounting for the detected road markings but to pre-emptively adapt the behaviour of the vehicle to future changes, such as a speed limit update.

Connectivity module

All V2X communication is handled on a separate device. The OBU connects via Ethernet to both the Speedgoat target computer and the Linux PC. Messages are transmitted via UDP and published as ROS topics for fusing them to the sensors. This connectivity enables us to test the potential improvements in traffic flow and safety on highways.

Cooperative awareness messages (CAM) are transmitted at frequencies of around 10 Hz and contain the current position, heading and speed of the vehicle. In our prototype, the CAM is automatically populated by the OBU using data from the GPS and speedometer and then sent via C-V2X. The connectivity module also listens for and receives CAMs from other vehicles to maintain situational awareness of nearby vehicles. CAMs improve range and perception: a car that is beyond or outside the sensor's line of sight can be detected through its CAM.

While not central to this work, the connectivity layer supports cooperative manoeuvres including sending alerts or suggestions (e.g. flashing signals or dashboard warnings) if it fails and it is extensible to vehicle-to-infrastructure (V2I) for future expansion.

Scenario-Based Safety Testing

The two scenarios executed on proving grounds and presented on this work focus on the validation of the prototype vehicle through the implementation of the connected HWP system, highlighting key considerations for its integration and V2V benefits.

The scenarios presented are two variants of a similar situation where two vehicles (one of which is the ego vehicle) are communicating their speed, position, and other relevant information through CAM messages and an additional non-connected vehicle is located between the ego vehicle and the connected vehicle, obstructing the ego's line of sight to the connected vehicle.

In the first scenario (represented in Figure 6), the three vehicles initially travel at the same speed, with the distance between the non-connected vehicle (hereinafter referred to as the central vehicle) and the connected vehicle (hereinafter referred to as the front vehicle) greater than the distance between the ego vehicle and the central vehicle. The scenario consists of the front vehicle braking and the central vehicle, which was not previously aware of the deceleration, reacting late and braking more abruptly as it approaches it. By receiving CAM updates from the front vehicle, the ego vehicle can anticipate the braking and adjust its speed smoothly to avoid abrupt or even emergency braking, such as that of the central vehicle.

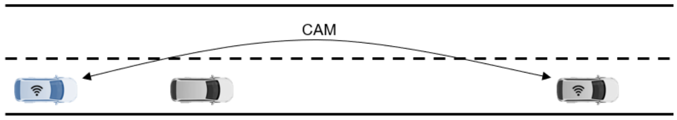

Figure 6 Diagram of the first variant of the scenario under test

In the second scenario (represented in Figure 7), the front vehicle is travelling at a lower speed than the central vehicle and the ego vehicle (which are both travelling at the same speed). In this case, the central vehicle performs a cut-out manoeuvre to avoid the slower front vehicle instead of braking. Despite lacking a direct line of sight to the slower front vehicle, due to the presence of the central vehicle in between, the ego vehicle is continuously tracking its position and motion state through the exchange of CAM messages. As in the first variant, the real-time connectivity message exchange allows the ego vehicle to anticipate. As a result, although the central vehicle performs the cut-out manoeuvre in response to the slower speed of the front vehicle ahead, the ego vehicle is able to adapt the speed to the front vehicle's one while maintaining a comfortable behaviour.

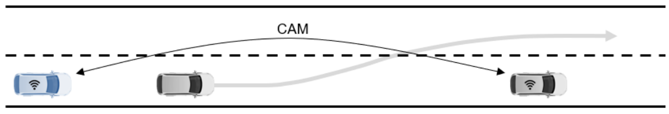

Figure 7 Diagram of the second variant of the scenario under test

The goal of these scenarios is to evaluate how this communication enables smooth and safe longitudinal control by allowing the ego vehicle to proactively respond to changes in traffic ahead and maintain a safe following distance. In the absence of such communication, the system would need to brake as a reaction to the perceived central vehicle braking and, thus, at a later point in time, increasing the risk of collision and reducing comfort due to abrupt decelerations.

These experimental scenarios where synthetised according to the Safety Assurance Framework (SAF) criteria. The underlying idea of this methodology is that a functionality can by validated by designing and assessing a set of critical highway scenarios that cover the system's ODD. In this way, a collection of demanding scenarios is selected to represent the ODD, allowing to build a safety argument for the ADS performance. For the validation of the HWP algorithm we applied the SAF to define a matrix of scenarios covering the speed range for both variants. The scenarios have been evaluated on closed proving grounds, with both autonomous and human-driven actors executing their role as predefined in the scenario matrix. The recorded vehicle data has been used to assess the safety of the HWP pilot algorithm with the V2V enhancement. Additionally, this data is currently used to further refine the algorithm, analysing its nominal behaviour and detecting triggering conditions that resulted in the need of human intervention.

The tests were performed at initial speeds of 80, 100, and 120 km/h and with final speeds 60 km/h or lower (i.e. 20, 40 and 60 km/h), which are typical highway speeds where this kind of system can make a difference. At these speeds, the ability to receive information from vehicles further ahead (beyond the line of sight) helps the ego vehicle react more smoothly and anticipate, improving the overall safety of the function. That's where connectivity becomes especially relevant and shows its potential. The initial headway of the ego vehicle to the central vehicle is 1.6 s in all cases, as it is a typical value for a state-of-the-art ACC system [8]. It should be noted that this is an early-stage proof of concept, aimed not at validating the entire functionality across its full ODD, but rather at testing the integration of the prototype and getting a first sense of the benefits these technologies can bring. The outcome of this initial phase will support future refinements and broader validation work.

It is worth mentioning that the SAF does not only cover the behaviour of the vehicles involved in the manoeuvre, but it encourages to also consider environmental conditions such as weather, light, humidity, and wetness of the pavement. This study focused on track testing, and it was decided to leave the environmental conditions out of the scope of the project.

Experimental Results (Proof-of-Concept)

The collected data has been used to extract safety and comfort metrics. On the one hand, we verified that no collisions or near-misses occurred, ensuring all safety constraints were satisfied during the tests. On the other hand, the acceleration of the vehicle is monitored to work on a comfort zone, ensuring that the vehicle occupants perceived smooth manoeuvres.

Scenario variant 1

For the first scenario variant, as mentioned, tests were conducted at initial speeds of 80, 100 and 120 km/h. The evaluation focused on the time evolution of the ego vehicle's speed and clearance (available free distance) of each vehicle involved in the test with respect to the vehicle directly ahead, as well as on key safety and comfort indicators such as the maximum deceleration experienced (minimum acceleration value) by the ego vehicle and the minimum Time-To-Collision (TTC) with respect to the central vehicle, which represents the potential risk of collision. Table 1 shows that, for all tested speeds, the minimum values of these indicators remained within very comfortable [9] and safe ranges [10], with ego decelerations around 2 m/s² and TTC values consistently above 4 seconds. These results confirm that the system was able to respond smoothly and maintain safe distances throughout the test runs.

Table 1. KPIs from scenario variant 1.

Initial Speed	Min acceleration	Min TTC
80 km/h	-1.94 m/s^2	4.2 s
100 km/h	-1.95 m/s^2	6.5 s
120 km/h	-2.03 m/s^2	5.4 s

As shown in the Table 1, for scenario variant 1, the lowest TTC value is at the test speed of 80 km/h. By analysing this specific case further, it reveals a consistently smooth system behaviour. Figure 8 shows the time evolution of the speeds of the three vehicles involved in the scenario. The front vehicle begins to brake at the fifth second of the plot. 7 seconds after (t=12.1s), the ego vehicle also starts to decelerate in anticipation of the situation, even before the central vehicle initiates braking (t=16.4s). This impacts also in the deceleration required to avoid a collision, as it can be seen in the different slopes of the speed profiles: while the central vehicle exhibits a sharp deceleration, reaching values close to -4 m/s², the ego vehicle's deceleration remains significantly milder, since it is not exceeding -2 m/s².

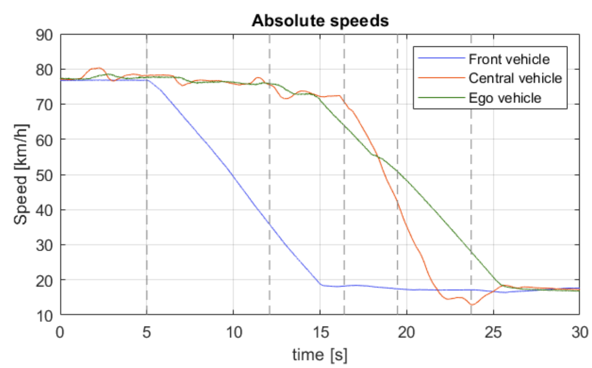

Figure 8 Temporal evolution of the speed of the three vehicles involved in test variant 1 at 80 km/h

Additionally, the clearance (or free distance) from the ego vehicle to the central one remains above 15 m during all the test execution as it can be seen in Figure 9. In fact, the distance between the front and central vehicles is considerably shorter (7 m), meaning that a possible rear-end collision between the ego vehicle and the central vehicle could have resulted in a multiple crash, involving the front vehicle as well. The minimum TTC value is obtained at t=23.4 s when the situation is completely under control and the ego vehicle is at the latest step of the deceleration (already at 30 km/h). This highlights the system's ability to anticipate and react in a controlled manner, contributing to both safety and passenger comfort in this scenario.

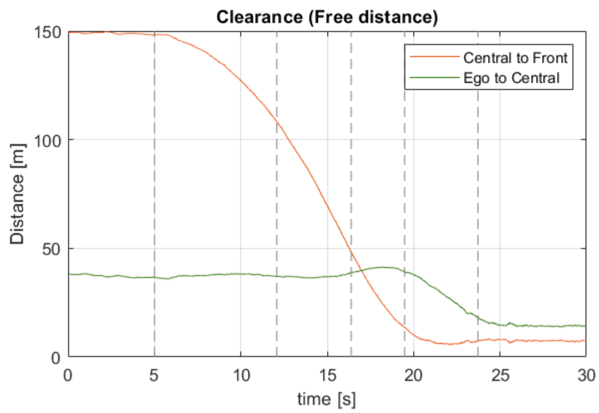

Figure 9 Temporal evolution of the longitudinal free distance between the vehicles involved the test variant 1 at 80 km/h

Scenario variant 2

For the second scenario variant, as mentioned, tests were conducted at initial speeds of 80, 100 and 120 km/h. The evaluation followed the same approach as in the first variant and focused on the time evolution of the ego vehicle's speed and clearance (available free distance) with respect to the rest of the vehicles involved in the test, as well as on key safety and comfort indicators such as the maximum deceleration experienced (minimum acceleration value) by the ego vehicle and the minimum TTC with respect to the front vehicle, which in this case represents the potential risk of collision, since the central vehicle cuts-out and leaves the ego vehicle's path. Table 2 shows that, for all tested speeds, the minimum values of these indicators remained within very comfortable and safe ranges, with ego decelerations around 2 m/s² and TTC values consistently above 8 seconds. These results confirm that the system was able to respond smoothly and maintain safe distances throughout the test runs.

Table 2. KPIs from scenario variant 2.

Initial Speed	Min acceleration	Min TTC
80 km/h	-1.9 m/s^2	8.16 s
100 km/h	-2.15 m/s^2	9.64 s
120 km/h	-1.69 m/s^2	12.64 s

The case at the highest initial speed (120 km/h) is particularly interesting, as it shows the highest recorded minimum TTC value, 12.64 s. This case has been analysed in more detail, since the system's early reaction transforms what could have been a potentially hazardous situation into a safe one, thanks to the use of CAM messaging. Figure 10 illustrates the time evolution of the speeds of all three vehicles involved in the scenario. It can be seen that the ego vehicle starts braking at t=6.5 s. When the central vehicle performs the cut-out manoeuvre at t=15 s, the ego vehicle is already travelling approximately 30 km/h slower. In fact, the lowest TTC value occurs just before this cut-out manoeuvre begins, at t=14.1 s. This means that by the time the front vehicle becomes visible to the ego vehicle, the TTC is already increasing, indicating that the risk of collision has already been mitigated.

However, it is worth noting that the ego vehicle's speed profile during the initial braking phase is not entirely smooth. The deceleration is slightly inconsistent, alternating between actual braking and brief periods of brakes release. This irregular behaviour is, in fact, the result of the system anticipating the need to slow down too early. Although the situation remains safe, this points to room for improvement in the decision-making and control algorithm to achieve a more comfortable and natural driving response.

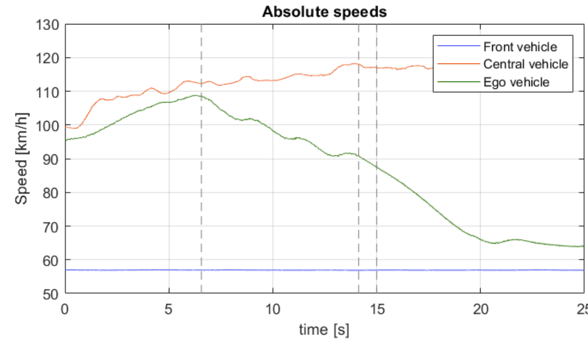

Figure 10 Temporal evolution of the speed of the three vehicles involved in test variant 2 at 120 km/h

Regarding the distance maintained by the ego vehicle with respect to the other two vehicles, Figure 11 shows how these distances evolve over time. From around t = 6.5 s, when the ego vehicle begins to decelerate, the distance to the central vehicle starts increasing. Meanwhile, the distance to the front vehicle is decreasing, but with a progressively flatter slope. At t = 15 s, when the central vehicle begins its lane change and leaves the lane travelled by the ego and front vehicles, the distance between the ego vehicle and the front one is still above 100 m. This distance then gradually stabilises above 50 m by the end of the test, in a smooth and controlled manner. These results support the statement that the system behaves safely and effectively, thanks to considering the information provided by CAM messages to improve the ego vehicle's awareness of its surroundings and reduce potential risks.

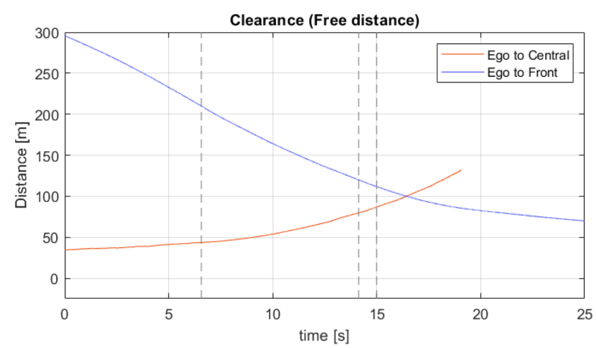

Figure 11 Temporal evolution of the longitudinal free distance between the vehicles involved the test variant 2 at 120 km/h

The results obtained across all test cases show consistently safe and controlled behaviour, demonstrating the overall system performance of the connected HWP functionality. These findings highlight the strong potential of incorporating CAM messaging to enhance safety in complex scenarios. At the same time, they confirm that the developed prototype is fully operational and serves as a valuable platform for testing and validating emerging technologies. By enabling early proof-of-concept evaluations, this approach helps to reduce development costs and effort, supporting active progress and innovation in the field of connected and autonomous mobility.

Conclusions and Future Work

This work presented the development and validation of a connected Highway Pilot (HWP) prototype vehicle, designed to operate autonomously in highway environments while leveraging Vehicle-to-Vehicle (V2V) communication to enhance system performance and safety. The main contributions of this study are the following: the detailed integration of sensing, computing, and communication hardware into a production vehicle to build a fully functional automated driving testbed; the design and deployment of a modular software architecture supporting real-time perception, localization, planning, control, and V2V connectivity; the implementation of Cooperative Awareness Messages (CAM) to enable proactive decision-making in complex scenarios, and thus improving situational awareness beyond onboard sensor's line-of-sight; and the application of the Safety Assurance Framework (SAF) as a structured approach for scenario-based testing in a controlled proving ground environment.

Experimental results confirmed that the connected HWP system operates safely and comfortably in complex highway situations. In both tested scenario variants, the CAM-based connectivity allowed the ego vehicle to anticipate critical events, such as sudden decelerations or cut-out manoeuvres, even when the critical target was obstructed. This capability enabled smoother responses with lower deceleration rates, thus improving passenger comfort and reducing the risk of emergency braking. In contrast, relying solely on onboard sensors would have needed later and more abrupt reactions, impacting negatively to both safety and driving quality.

This initial proof-of-concept demonstrates the potential of enhancing with connectivity the highway pilot functionalities and provides a solid foundation for future developments. As next steps, we aim to expand the functionality, fixing some detected issues, and also enlarge the scenario matrix to gather more complex conditions, such as multi-lane traffic interactions, manoeuvre coordination and Vehicle-to-Infrastructure (V2I) communications. Further work will also include the integration of simulation-based testing to complement physical tests, enabling broader coverage of the Operational Design Domain (ODD). The SAF methodology will be extended to cover all the analysis, including robustness against sensor faults and degraded conditions in order to seek the corner cases and the limits of the system performance.

The open and modular nature of the platform ensures the seamless integration of new components, allowing for rapid prototyping of emerging functionalities such as AI-driven perception or advanced motion planning. The iterative use of the SAF across these stages will support continuous improvement and systematic validation, contributing to the long-term goal of achieving innovation in the field of autonomous driving systems.

References

1. SAE, "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," *J3016_202104,* pp. 24-32, 2021.
2. "SUNRISE," 2025. [Online]. Available: https://ccam-sunrise-project.eu/.
3. ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service," ETSI EN 302 637-2 V1.4.1, Apr. 2019.
4. Yang T, Bai Z, Li Z, Feng N, Chen L. "Intelligent Vehicle Lateral Control Method Based on Feedforward + Predictive LQR Algorithm". *Actuators*. 2021; 10(9):228, doi:10.3390/act10090228.
5. Visioli, A. (2012). "Research Trends for PID Controllers." Acta Polytechnica, 52(5). doi: 10.14311/1656.
6. M. Perez and A. Agudo, "Robust multimodal and multi-object tracking for autonomous driving applications," in *21st International Conference on Advanced Robotics (ICAR)*, 2023, doi: 10.1109/ICAR58858.2023.10406433.
7. M. Perez and A. Agudo, "Sensor-agnostic multimodal fusion for multiple object tracking from camera, radar, lidar and V2X," in *FISITA World Congress*, 2023.
8. S. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., *Handbook of Driver Assistance Systems*, Amsterdam, The Netherlands: Springer International Publishing, 2014, pp. 405-430, doi:10.1007/978-3-319-09840-1.
9. C. Wang, X. Zhao, R. Fu, and Z. Li, "Research on the comfort of vehicle passengers considering the vehicle motion state and passenger physiological characteristics: Improving the passenger comfort of autonomous vehicles," *Int. J. Environ. Res. Public Health*, vol. 17, no. 18, p. 6821, Sep. 2020, doi: 10.3390/ijerph17186821.
10. B. Sultan and M. McDonald, "Assessing the safety benefit of automatic collision avoidance systems (during emergency braking situations)," in *Proc. 18th Int. Tech. Conf. Enhanced Safety of Vehicles*, Nagoya, Japan, May 2003, DOT HS 809 543.

Acknowledgments

The research presented in this work has been made possible by the SUNRISE project. This project is funded by the European Union's Horizon Europe Research & Innovation Actions under grant agreement No. 101069573. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Abbreviations

ACC	Adaptive Cruise Control
ADAS	Advanced Driving Assistance System
ADS	Automated Driving System
AEB	Automated Emergency Braking
CAM	Cooperative Awareness Message
CAN	Controller Area Network
CAV	Connected and Automated Vehicle
CCAM	Cooperative, Connected and

	Automated Mobility
CRC	Cyclic Redundancy Check
DTC	Data Trouble Code
ECU	Electronic Control Unit
EKF	Extended Kalman Filter
ETSI	European Telecommunications Standards Institute
E/E	Electrical/Electronic
GNSS	Global Navigation Satellite System
HWP	Highway Pilot
IMU	Inertial Measurement Unit
LKA	Lane Keep Assist
LQR	Linear Quadratic Regulator
OBU	On-Board Unit
ODD	Operational Design Domain
OEM	Original Equipment Manufacturer
SAE	Society of Automotive Engineers
SAF	Safety Assurance Framework
SWC	Software Component
TTC	Time-To-Collision
UDP	User Datagram Protocol
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything