sponsored by Federal Ministry of Transport and Digital Infrastructure

Deliverable D5.1: Vehicular ITS Station Specification

Version: V1

July 31, 2019

Contents

1 Scope

This document describes the usage of the C-V2X development platform as vehicular ITS station and how it is integrated into the Audi vehicle.

2 References

[1] ConVeX Project Deliverable D2.2, “Final ConVeX System Architecture Description”

3 Definitions and Abbreviations

3.1 Definitions

Host Vehicle (HV): The vehicle whose driver is alerted or notified by a warning or notification indication by its In-Vehicle Information system (e.g. visually and/or acoustically), as a result of the reception and processing of CAM's and DENM's from surrounding other vehicles.

PC5 transport: Transmission of V2X data from a source UE (e.g., a vehicle) to a destination UE (e.g., another vehicle, road infrastructure, a pedestrian, etc.) via ProSe Direct Communication over the PC5 interface between the UEs (sidelink).

Remote vehicle (RV): The collection of all vehicles in a traffic scenario which surround a considered Host Vehicle. The Remote Vehicles broadcast their positions and vehicle information periodically and also create special DENM warning messages in case of safety-relevant events.

3.2 Abbreviations

CAM	Cooperative Awareness Message as defined in ETSI a message vehicles issue in 1Hz to 10 Hz interval to send their at least position and heading to its surrounding using local communication
CAN	Controller Area Network
C-ITS	Cooperative Intelligent Transport Systems
C-V2X	Cellular Vehicle to Everything
C-V2X DP	C-V2X Development Platform
DENM	Decentralized Environmental Notification Message as defined in ETSI a message vehicles or infrastructure components send case any relevant warning shall be issued to nearby vehicles
DHCP	Dynamic Host Configuration Protocol
DP	Development Platform (here: same as C-V2X DP)
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HMI	Human Machine Interface
HW	Hardware
ICS	ITS Central Station
ITS	Intelligent Transport System
IP	Internet Protocol
IVI	In Vehicle Information as defined in SAE and CEN/ISO information on current (dynamic) sign display to be sent from infrastructure to vehicles
JSON	JavaScript Object Notation
LTE	Long Term Evolution
MMI	Multi Media Interface
MQTT	Message Queuing Telemetry Transport
PC5	ProSe Communication reference point 5
ProSe	Proximity-based Services
RV	Remote Vehicle
RSU	Road Side Unit
SAE	Society of Automotive Engineers
SW	Software
TCP	Transmission Control Protocol
UE	User Equipment
V2X	Vehicle to Everything
V-ITS-S	Vehicular Intelligent Transport System Station

4 Vehicular ITS Station

4.1 Overall Architecture

The main component is the C-V2X Development Platform (C-V2X DP), which runs the ITS stack and ITS application (for details refer to section 5.1 of [1]). It is connected to the car in two ways. For base information like velocity, x- and y-acceleration, braking interventions a CAN interface is used. For interpretation of the CAN messages a database is also provided.

The connection between the box and an additional computer (CarPC) is achieved by using the Wi-Fi interfaces of each device. The CarPC is needed to view icons and text information on the display in the vehicle. In order to make both devices communicating with each other the CarPC needs to access the DP onto the configured IP address. By using a DHCP client the CarPC and the DP receive an allocated address for the communication.

Figure 4.1-1 shows a schematic of the vehicle integration (Note: here Ethernet connectivity is shown, which would also be an option of the connection between CarPC and C-V2X DP).

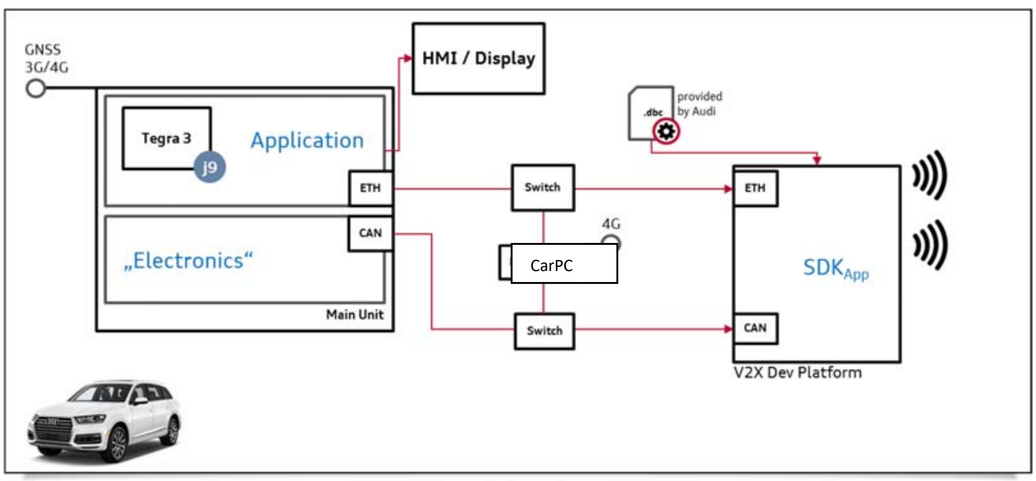


Figure 4.1-1: schematic of the vehicle integration

4.1.1 CAN bus

As mentioned above a major part of vehicle internal information is transmitted via CAN frames. It contains various bits of information and is recognized by an identifier. Figure 4.1-2 gives an overview about the structure of a CAN frame.

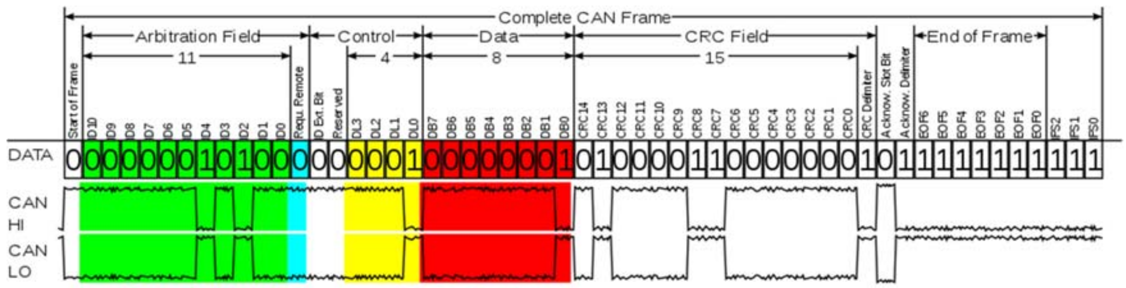


Figure 4.1-2: CAN frame [Source: Wikimedia Commons]

The payload in a CAN frame contains the actual information, which is proprietary to automakers. The specific information of the coding of the frames and identifiers can be defined in a database. The .dbc format file specified by Vector Informatik GmbH is an example, which is intended to be used in this project. The transport layer is specified in ISO 15765-2.

Over the CAN, basic information such as speed, heading, brake operations, accelerations etc. is accessible for the communication platform.
For connecting the C-V2X DP to a vehicle system a Peak CAN adapter is in use which is displayed in the following picture next to a CAN micro gateway.

Figure 4.1-3: PEAK CAN USB and CAN micro adapter

Since the used V2X SW on the DP expects a specific set of CAN messages an adapter gateway is needed to translate the individual CAN messages from the vehicle bus into the expected format. Therefore, the CAN specification represented by the dbc-File is used to interpret the data in a correct way, using it as an input for the output format.

The expected format consists of the defined 6XX messages by CAMP for V2X use cases.

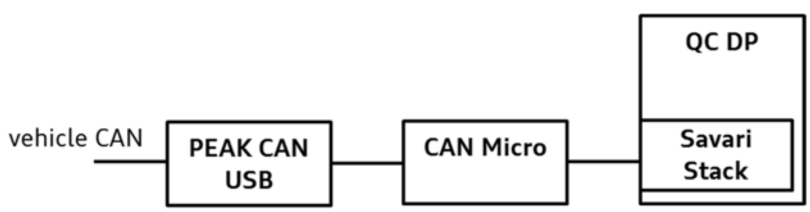
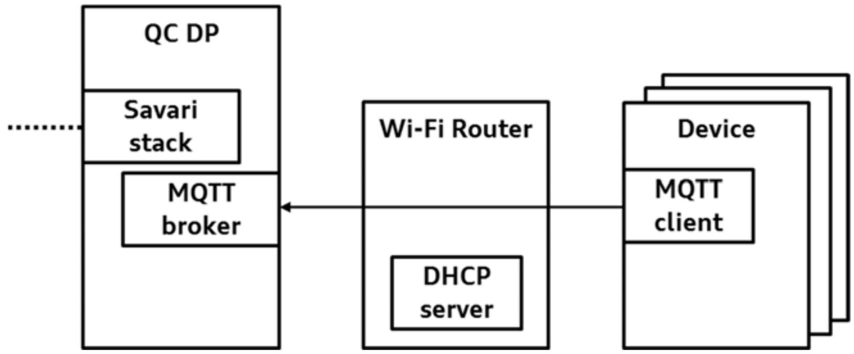


Figure 4.1-4: Schematic of bus translation

4.1.2 MQTT communication and JSON schema

In order to maintain a communication to other devices within the same IP Network an MQTT broker is running on the C-V2X DP. Therefore some other device could connect via accessing the IP address of the DP (address is configured DHCP based) and the allocated port (1883) via TCP to establish an MQTT connection by using a compatible client. This way, a publish/subscribe-system is established and any additional device can be connected to the broker using the connection data as the following scheme displays:



4.1-5: Schematic of MQTT connection to a device like a smartphone, PC or an Audi HMI system

By subscribing to the topic "app2IviTopic" the client periodically receives a JSON message which includes the position of the HV and the locations of available RVs. In case a V2X Use case is triggered the JSON message gets extended to also include that information. The following message shows an example of an IVI message:

```
{
    "body" : {
        "warn_msg" : {
            "audio_asset" : "",
            "clear_duration" : 0,
            "display_assets" : [
                "speed-limit"
            ],
            "enable" : true,
            "parameters" : {
                "speed_limit" : 223.69,
                "weight_limit" : 7.5
            },
            "priority" : 1,
            "severity" : "Imminent",
            "stats" : {
                "hv_info" : {
                    "heading" : 0,
                    "latitude" : 49.48130632,
                    "longitude" : 11.11543422,
                    "range" : 0,
                    "speed" : 0
                },
                "peer_info" : {
```

```
                "hazard_info" : {
                    "latitude" : 0,
                    "longitude" : 0,
                    "range" : 0
                }
            }
        },
        "timestamp" : 1551959504897,
        "vibrate" : false,
        "warning_message" : "speed limit: 223kmph, weight limit: 7tonnes",
        "warning_title" : "IVI"
    }
```


Figure 4.1-6: Sample of a JSON message for an IVI event

4.1.3 Vehicle HMI

The vehicles contain capabilities to present information to the driver acoustically and visually ex-factory. Additionally, there are also interfaces for input available. The driver can select menu entries and interact with the CarPC like this due to his input actions.

Displays

MMI Display:
The MMI display is located to the right of the steering wheel. It has a resolution of 1024x480 pixels. Usually this display shows multimedia applications or is used for visualization of route guidance or other navigation functions. Technically, the full resolution of the display can be used in the project to visualize information. Figure 4.1-7 displays the installation location and navigation screen for routing functions

Figure 4.1-7: MMI display - the top shows the installation location, the bottom a typical navigation screen

Dashboard:
The instrument cluster is located behind the steering wheel. Fixed there are displayed the vehicle's velocity and the engine's rotational speed. The total resolution is 1440 x 540 pixels. In the project available for displaying information is the area in between the fixed tubes, as Figure 4.1-8 shows.

Figure 4.1-8: Audi instrument cluster – the hatched area describes the available space for displaying information in the project.

The displays can be connected like common monitors. With a video port connected to the CarPC an icon or a video can be displayed on the hardware. In order to display a certain message or icon, a notification from a specific use case has to be mapped to that icon image or a display scheme since the DP does not render nor contains visualization components.

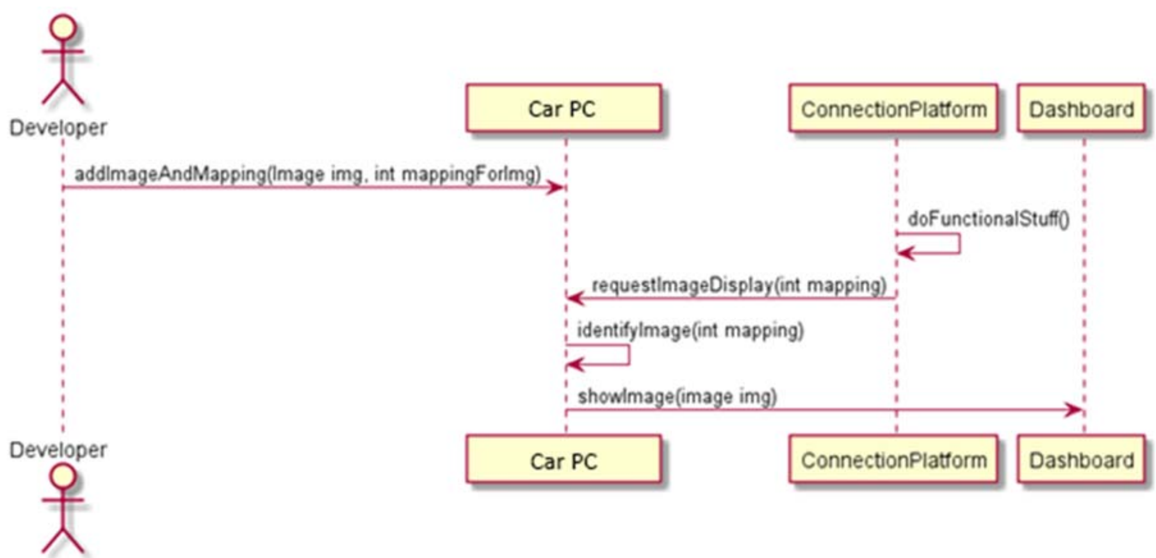


Figure 4.1-9: Sequence of displaying an image

Once the software in the CarPC has the references for image and sounds for the use cases, the HMI would react accordingly to the incoming messages received per MQTT from the topic it is subscribed to.

4.1.4 Antenna systems

During the tests and development phases there are two types of antenna systems in use.

The first set consists of two 5.9 GHz V2X antennas and a single GPS antenna.

Figure 4.1-10: reference antenna system

The second type is an antenna system provided and manufactured by Kathrein. The antenna design resembles the typical Audi roof antenna but was augmented with two V2X dipols. The system does also feature a GNSS and a LTE antenna as Figure 4.1-11 shows. The GNSS/GPS antenna is both connected to the infotainment system and the DP by using a splitter device.

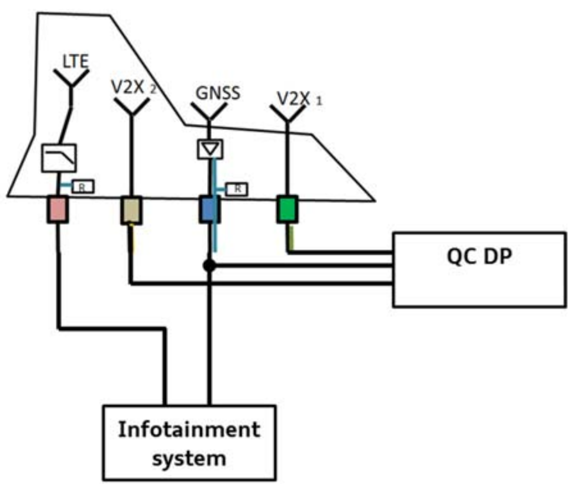


Figure 4.1-11: Kathrein antenna system with two V2X dipols

During the tests and demonstrations there were no spotted issues regarding using different GPS antennas. However, using different 5.9 GHz antenna resulted in different behaviors regarding range and package error rate, with significant advantages for the reference antenna setup. For a commercial deployment, more investigations and optimizations have to be carried out.