



# DEVELOPMENT AND TESTING OF THE INDUCTIVE WIRELESS CHARGER

D5.4 – INDUCAR System Development and Laboratory Test

Date: **29/11/2022**

Author(s): **Cas van Raan (IPT), Sergio Perez (IPT)**



This project has received funding from  
the European Union's Horizon 2020  
research and innovation programme  
under grant agreement No [875187]



## Deliverable details

Project number	Project acronym	Project title
875187	USER-CHI	Innovative solutions for USER centric Charging Infrastructure

Title	WP	Version
D5.4 – INDUCAR system development and Laboratory Test	WP5	1.2

Contractual delivery date	Actual delivery date	Delivery type*
M27 (April 2022)	M34 (November 2022)	R

\*Delivery type: **R**: Document, report; **DEM**: Demonstrator, pilot, prototype; **DEC**: Websites, patent fillings, videos, etc; **OTHER**; **ETHICS**: Ethics requirement; **ORDP**: Open Research Data Pilot.

Author(s)	Organisation
Sergio Perez	IPT-Technology
Cas van Raan	IPT-Technology

## Document history

Version	Date	Person	Action	Status*	Dissemination level**
V0.1	1-9-2022	Cas van Raan (IPT) Sergio Perez (IPT)	New Document	Draft	CO
V0.2	19-10-2022	Cas van Raan (IPT) Sergio Perez (IPT)	Technical review	Draft	CO
V0.3	26-10-2022	Cas van Raan (IPT) Sergio Perez (IPT)	Technical review	Draft	CO
V0.4	9-11-2022	Cas van Raan (IPT) Sergio Perez (IPT)	Technical review	Draft	CO
V0.5	25-11-2022	Cas van Raan (IPT) Sergio Perez (IPT) Victor Hoyneck (IPT)	Final management internal review	Draft	CO
V1.0	28-11-2022	Ángel Moya (ETRA)	Peer review	Draft	CO
V1.1	29-11-2022	Sergio Pérez (IPT)	Pre-final version	Draft	CO
V1.2	29-11-2022	Ángel Moya (ETRA)	Final version	Submitt	PU

\*Status: Draft, Final, Approved, Submitted (to European Commission).

\*\*Dissemination Level: **PU**: Public; **CO**: Confidential, only for members of the consortium (including the Commission Services); **EU-RES** Classified Information - restraint UE; **EU-CON**: Classified Information - confidential UE; **EU-SEC**: Classified Information - secret UE

## Abstract

The INDUCAR Product belongs with the INSOC Product, in the USER-CHI project, to the main work package dedicated to the development of new technologies. The INDUCAR – Inductive Charging for e-Cars – consists of an inductive charging system to recharge the e-Cars battery with a solution bringing an important level of automation. The wireless system will implement Machine-to-Machine communication allowing the system to connect to standard user-friendly applications. These applications might take in charge the functionalities typically expected for the whole process, like the identification of the e-Car, the payment and charging feature the user needs to go through for the charging process.

The aim of the deliverable D5.4 – INDUCAR System Development and Laboratory Test is to describe the development process and the test results of the laboratory tests leading to a full operational inductive system retrofitted to an existing e-Car. More specifically, in the frame of the

USER-CHI project, it has been determined that the INDUCAR will be implemented in the Barcelona Demonstrator in two existing Renault Zoe that belong to the fleet of the AMB – Àrea Metropolitana de Barcelona. The Renault Zoe is retrofitted with the inductive charging solution taking into account in its design the mechanical and electrical integration. The INDUCAR design defines the functional integration for the whole charge process and provides the system internal functionalities inherent to it.

## Keywords

Wireless charging system, inductive charger, automatic charging, autonomous driving, inductive system for cars, requirements for inductive technology, inductive design.

## Copyright statement

The work described in this document has been conducted within the USER-CHI project. This document reflects only the USER-CHI Consortium view and the European Union is not responsible for any use that may be made of the information it contains.

This document and its content are the property of the USER-CHI Consortium. All rights relevant to this document are determined by the applicable laws. Access to this document does not grant any right or license on the document or its contents. This document or its contents are not to be used or treated in any manner inconsistent with the rights or interests of the USER-CHI Consortium or the Partners detriment and are not to be disclosed externally without prior written consent from the USER-CHI Partners.

Each USER-CHI Partner may use this document in conformity with the USER-CHI Consortium Grant Agreement provisions.

## Executive summary

The USER-CHI deliverable D5.4 “INDUCAR System Development and Laboratory Test” summarises the results of the tasks T5.5. “Development of the inductive automated wireless charging system”, and T5.6 “Testing and refinement of the inductive automated wireless charging system”. The task T5.5 aims to bring forward the practical development of the automated wireless system according to the outcome resulting of the Task T5.4 “Design of services and specifications of the inductive automated wireless charging system” and the corresponding deliverable D5.3 “INDUCAR system design”. The task T5.6 aims to evaluate and conclude about the performances of the equipment developed for INDUCAR during the task T5.5. that shall be demonstrated in Barcelona.

This document describes first the development efforts through the description of the different milestones needed to be achieved to fulfil all the requirements stated in the first deliverable D5.3. The second main section describes the works done during the testing phase.

The USER-CHI task 5.5 and task 5.6, and the present report D5.4 are the result of the development of the INDUCAR product, that shall lead into the demonstration phase foreseen during the project.

This deliverable “D5.4: INDUCAR System Development and Laboratory Test” verifies joint with “D5.2: INSOC system hardware development and lab-testing” (foreseen to be submitted by M40) the milestone MS4.

## Table of Contents

<b>1.</b>	<b>Introduction .....</b>	<b>8</b>
1.1	Purpose of the document .....	8
1.2	Scope of the document .....	8
1.3	Structure of the document .....	8
1.4	Important definitions .....	9
1.5	System overview .....	10
<b>2.</b>	<b>Development .....</b>	<b>12</b>
2.1	Milestone 1: CCS communication .....	13
2.2	Milestone 2: Searching state .....	15
2.3	Milestone 3: Found state .....	17
2.4	Milestone 4: Z mover state .....	19
2.5	Milestone 5: Normal operation state .....	19
2.6	Milestone 7: CCS driver deployed.....	20
2.7	Milestone 8: CCS driver integrated .....	20
2.8	Milestone 6 & 9: integration.....	22
<b>3.</b>	<b>Testing .....</b>	<b>28</b>
3.1	Software in the loop.....	28
3.2	Hardware in the loop .....	29
3.3	Integration testing.....	33
<b>4.</b>	<b>Conclusion.....</b>	<b>35</b>
<b>5.</b>	<b>References and acronyms.....</b>	<b>37</b>
5.1	References.....	37
5.2	Acronyms.....	40

## List of figures

---

Figure 1: system architecture .....	10
Figure 2: The CCS driver application and AUTOSAR Adapter .....	14
Figure 3: test setup CCS communication .....	15
Figure 4: overview of the build environment.....	16
Figure 6: compatibility matrix.....	18
Figure 7: schematic overview of pre-charge .....	19
Figure 8: overview of integration of the CCS driver application .....	21
Figure 9: The test setup without wireless charging.....	23
Figure 10: Deployment architecture.....	24
Figure 11: test setup with wireless charging outside of the vehicle .....	26
Figure 12: location of the junction box .....	27
Figure 13: additional screen interface .....	28
Figure 14: example output of Ceedling run .....	29
Figure 15: On-desk test setup .....	30
Figure 16: CANoe simulation for CCS supply equipment board .....	31
Figure 17: Vehicle simulation .....	32
Figure 18: Lab test setup.....	33
Figure 19: vehicle during integration testing.....	34
Figure 20: Final installation of the charger at the AMB parking offices .....	36

# 1. Introduction

---

## 1.1 Purpose of the document

The purpose of this deliverable is to document the work carried out in the Task T5.5 concerning the development, and the work carried out in the Task T5.6 concerning the lab testing of the inductive automated wireless charging system. This document contains the development and testing procedure of the product INDUCAR which refers to the Inductive Charging for e-Cars development inside USER-CHI project. Its design shall fit better the “users’ charging preferences and maximise their convenience” as described in the USER-CHI project targets.

## 1.2 Scope of the document

The D5.4 provides the fundamental process used to develop and test the prototypes and final wireless charging system defined as the product INDUCAR inside the USER-CHI project. The document focus is centered on the way of working as well as the results of the project.

## 1.3 Structure of the document

This document is constituted of two main sections:

1. Development (chapter 2) which will go into detail of the development process. That chapter is divided into the phases which were executed during the project. The chapter itself will be divided into the milestone’s which were defined internally for this project.
2. Testing (chapter 3) which will go into detail on the way testing has been executed in this project. That chapter is divided into the different layers of testing from a bottom-up perspective. It will start with software in the loop testing and work up all the way to integration testing on the vehicle.



## 1.4 Important definitions

Term	Definition
Foreign Object	in the context of this document, is defined as any object that is not an attached part of the vehicle or the WPT system
High Voltage	in the context of this document, is defined as a voltage greater than 60V and less than 600V
ORU	Onboard Receiving Unit; the unit built into the vehicle to receive the power wirelessly
PAD	The unit on a parking space, connected to the grid to provide the electromagnetic fields to charge wirelessly.
AUTOSAR	AUTOSAR (AUTomotive Open System ARchitecture); a platform to standardize the interface within automotive ECU's.

## 1.5 System overview

The system which is integrated on the Renault Zoe's is based on a system tailored to another vehicle. During this project the developed system was tailored to the new vehicles and extended to communicate over CCS to avoid the need for full integration with the vehicle. Figure 1 shows the system level architecture from an integration point of view.

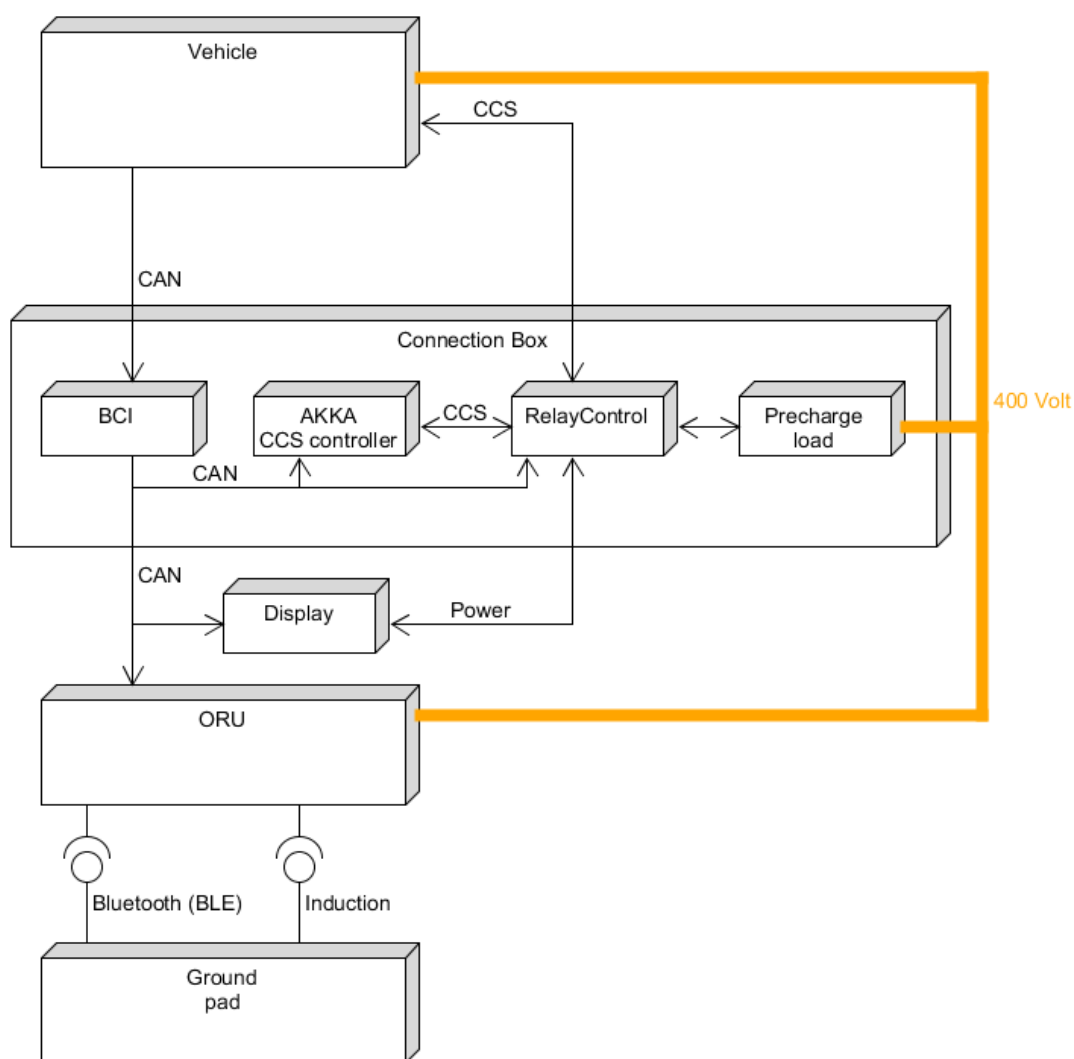


Figure 1: system architecture

Two interfaces to the vehicle will be used:

1. CCS port to communicate with the battery management system
2. CAN bus connection through a gateway which will forward information from the vehicle CAN bus to the new CAN bus. This will be used for signals like the vehicle speed and steering angle.

On the wireless charging CAN bus there are five devices:

1. Vehicle pad (ORU), which will handle:
  - The conversion of the inductive energy from the wayside pad
  - The communication over Bluetooth energy
  - Providing the vehicle with high voltage power
2. The screen which will show user feedback on the status of the charger and the location of the wayside pad in relation to the vehicle.
3. The CCS communication board which will communicate with the vehicle to start charging and how much it should charge.
4. The BCI translates the Renault Zoe CAN data to a standardised format to be used by the ORU.
5. The RelayControl has the following functions:
  - Switches the CCS communication pins
  - Powers the display
  - Sets the load for the precharge functionality

The wireless charging solution will communicate to the ground plate using Bluetooth low energy.

## 2. Development

The development of the solution was divided into nine milestones:

- **Milestone 1: CCS communication:** Have CCS communication with a Renault Zoe through the charging port.
- **Milestone 2: Searching state:** Have the ORUP go into searching state, by only delivering it the CAN signals present in the Renault Zoe.
- **Milestone 3: Found state:** Have the ORUP go into found state, by only delivering it the CAN signals present in the Renault Zoe and powering the Pad.
- **Milestone 4: Z mover state:** Have the ORUP go into Z mover state, by only delivering it the CAN signals present in the Renault Zoe, powering the Pad and putting the ORUP over the pad in the test setup.
- **Milestone 5: Normal operation state:** Have power transfer from the Pad to the ORUP, using the signals from the Renault Zoe, outside of a vehicle.
- **Milestone 6: Integration:** Have power transfer from the Pad to the ORUP, using the signals from the Renault Zoe, in a Zoe.
- **Milestone 7: CCS driver deployed:** Have the CCS driver deployed on the target and working with an external power supply.
- **Milestone 8: CCS driver integrated:** Have the CCS driver deployed on the target and working with wireless energy transfer.
- **Milestone 9: Final integration:** Have two (2) Zoe's ready for delivery with the system completely integrated.

Milestone 2 through 5 are based on the internal states of the ORU. The rest of the milestones is for integration or building the CCS driver. The paragraphs 2.1 until 2.8 will go into detail of every milestone. In these paragraphs, Milestone 6 and milestone 9 are combined.



## 2.1 Milestone 1: CCS communication

The first milestone has the goal to verify if the chosen hardware will be able to interface with the vehicle.

### 2.1.1 Execution

All the development for this milestone has been done using a regular windows computer. Removing the embedded target will help to shorten the loop between development and testing. To avoid implementing the same logic twice, the application was built in such a way that it would be easily portable to the embedded target.

The CCS Controller Board driver is depicted in Figure 2 as a single state in the worker thread on the right. It is designed to easily slot into an AUTOSAR environment, using the AUTOSAR adapter in Figure 2.

The AUTOSAR adapter serves to simplify the function names provided by the AUTOSAR environment into something briefer. In the case of the desktop test application, it is co-opted into redirecting the interface into the CAN adapter written for that environment.

The CCS Driver main function is as follows:

1. In case of any error, disconnect CCS and set requested power to 0.
2. If pads are aligned and the user requested charging:
  - a. Reflect any request back to the CCS controller,
  - b. Translate signals for requesting voltage and current for the existing code and send them.
3. If the user requested charging to stop:
  - a. Send the stop signal,
  - b. Follow vehicle state.

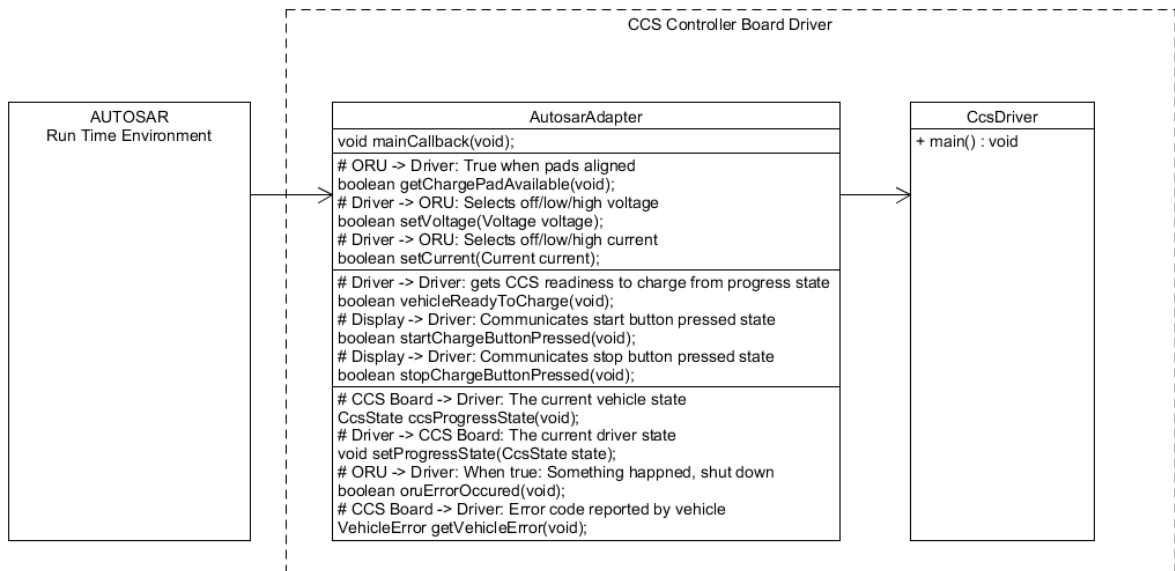


Figure 2: The CCS driver application and AUTOSAR Adapter

During the onsite test, the CCS driver has been tested using a Windows target with a simulated AUTOSAR run time environment. In the test setup is depicted in Figure 3. The CCS driver runs on Laptop 1, will replace the onboard receive unit. The high voltage charger is simulating the wireless energy transfer.

Laptop 2 is connected for debugging and generating logging files, it is removed for the end product.

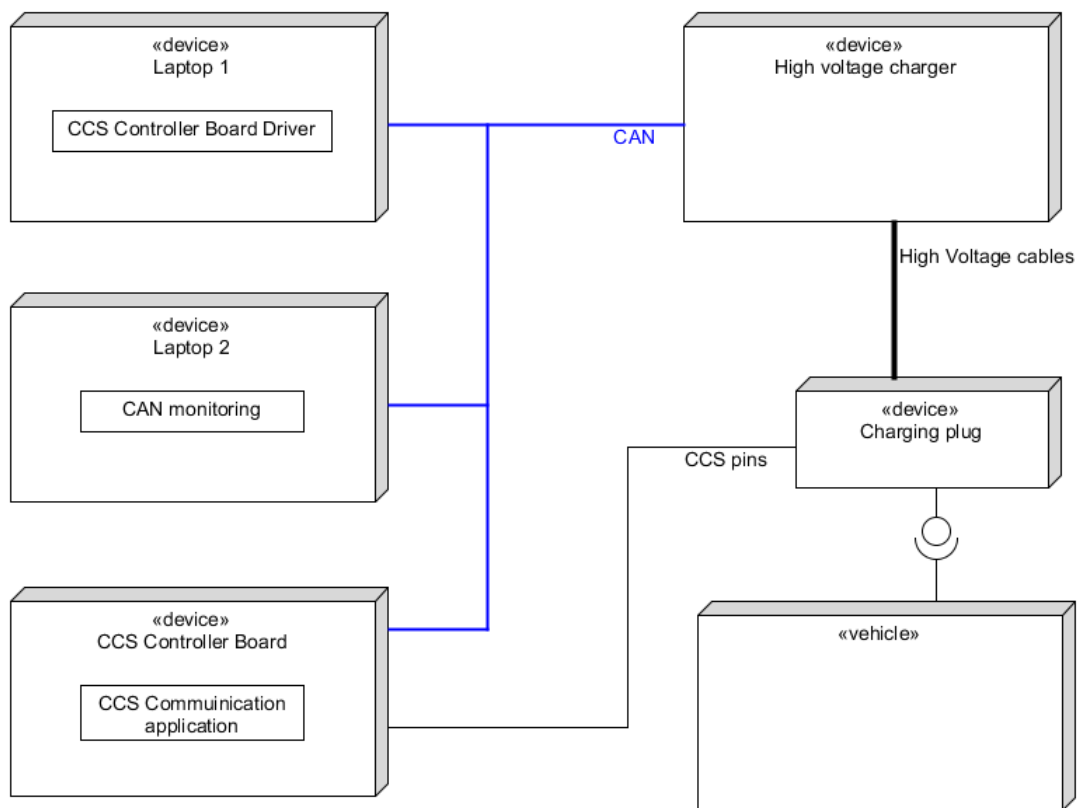


Figure 3: test setup CCS communication

## 2.2 Milestone 2: Searching state

Milestone 2 has the following goals:

- Define the interface between the new CAN bus and the existing CAN bus.
- Adjust the CAN specification for the ORU in such a way that it will go into searching state when the signals of a Renault Zoe are provided.
- Define the workflow to adjust the CAN specification.
- Get the code generation environment up and running.

## 2.2.1 Execution

### 2.2.1.1 Defining the workflow and get tools up and running

Figure 4 describes all the relations between the different tools which are used to build the application to be flashed on the ORU.

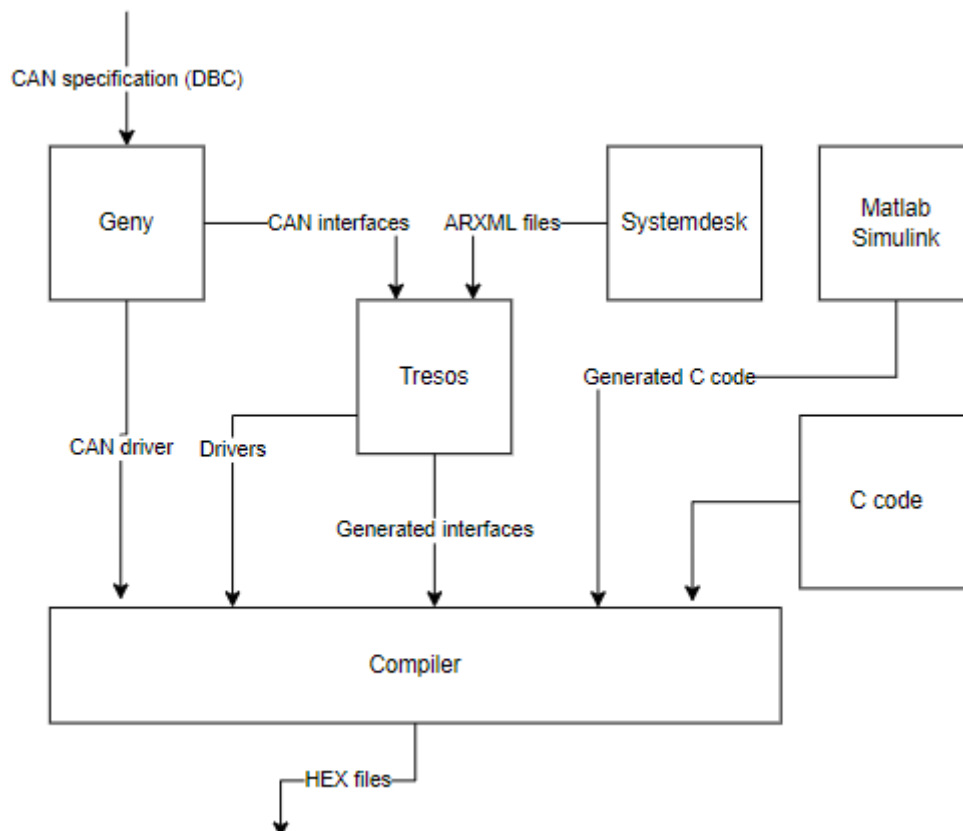


Figure 4: overview of the build environment

Geny is responsible for generating the CAN interfaces and drivers. Systemdesk is used to generate ARXML files which describe the interfaces of all the applications running on the target. Tresos generates code (interfaces and drivers) with this input.

The code generated by Tresos and Geny is combined with generated code from Matlab Simulink and manually written C code in the compiler to generate one binary file which is flashed into the target.

A bottom-up way of working was used to define this workflow. First the workings of Geny were discovered and documented. After that the same happened with Systemdesk and Tresos. Since there was already quite some knowledge on Matlab Simulink, the effort was only related to documentation. Lastly all the build scripts were checked to see if they would work in the USER-CHI environment.

After all this, the HEX files (binary files) could be generated and flashed onto the target.



### 2.2.1.2 Defining the CAN specification

There was no support from Renault in this project. To define the CAN specification towards the vehicle, an effort was made to reverse engineer the CAN protocol of the Renault Zoe. Renault uses a proprietary protocol. Therefore, the choice was made to use an off the shelf gateway from one of the partner companies. This gateway has the CAN definition implemented into it and can forward data to a standardized protocol to the other CAN bus.

## 2.3 Milestone 3: Found state

Milestone 3 has the following goal:

- Have a stable Bluetooth connection between the ORU and PAD. Communicating everything necessary to start a charging session.

### 2.3.1 Execution

The 3<sup>rd</sup> milestone has mainly been an execution of recommissioning the system already in place. The first step needed for this was to determine which software versions were compatible with each other. The following components have software versions which need to be compatible:

1. Foreign object detection
2. Bluetooth firmware of the PAD
3. Main controller of the PAD
4. Bluetooth firmware of the ORU
5. Main controller of the ORU

Figure 5 shows how these programs are used in the hardware. Both the ORU and PAD have software related to their main controller board and for the Bluetooth chip. The PAD has the firmware for the foreign object detection as well.

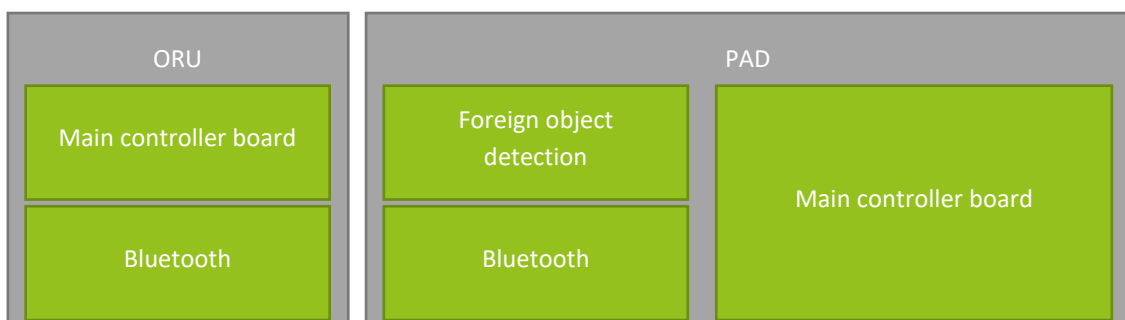


Figure 5: overview of different microcontroller in the ORU and PAD

Figure 6 shows the compatibility matrix made for this product. It gives information on the compatibility between software versions and hardware. The hardware versions are on the left of

the table, the software versions are on the top and halfway the table. If software is compatible with the hardware it will show by having a green checkmark. For example, the X26 version of the ORU is only compatible with the X370. This software is only compatible with the X26 hardware of the PAD, so no mixing in hardware version can happen here, which is possible with X410 software for example. This can work with X27 until X31 hardware of the ORU and X28 until X30 hardware of the PAD.

Since the X35 PAD is still a prototype, X34 was used for this project. This means that the X450 software needs to be used. X32 was chosen for the ORU, since that is the most recent version and is compatible with the X450 software.

For both the PAD and the ORU the X440 needs to be used for the Bluetooth firmware.

The firmware of the Foreign Object Detection is updated automatically when the PAD software is updated.

	SW Release	X370	X371	X380	X410	X415	X420 <sup>1)</sup>	X430 <sup>2)</sup>	X440	X450
	ORU	x		x	x	x	x	x	x	x
	Pad	x	x	x	x	x	x	x	x	x
ORU	SW BTL	X360		X360	X410	X416 (X420)	X420	X420	X426	
	SW BLE			X356	X356	X415	X420	X420	X440	
Muster	ORU HW Nummer PL65									
	X26	✓	✓	✓						
C0	X27-X29				✓	✓	✓	✓	✓	✓
C1	X30				✓	✓	✓	✓	✓	✓
	X31				✓	✓	✓	✓	✓	✓
C2	X32								✓	✓
PAD	SW BTL	X360		X360	X410	X416	X421	X421	X426	
	SW BLE			X356	X357	X415	X420	X420	X440	
Muster	PAD HW Nummer PL65									
	X26	✓		✓						
	X27		✓							
C0	X28-29				✓	✓	✓	✓		
C1	X30				✓	✓	✓	✓		
C2	X31								✓	✓
	X32								✓	✓
	X33								✓	✓
	X34									✓
	X35									✓

Figure 6: compatibility matrix

When all the correct software versions are loaded onto all the microcontrollers in the system and the CAN bus of the vehicle is simulated, the system goes into found state. It is then possible to

have full communication between the PAD and ORU. Information on how to flash the microcontrollers can be found in the software guidebook.

## 2.4 Milestone 4: Z mover state

Milestone 4 has the following goals:

- Have the ORUP go into Z mover state, by only delivering it the CAN signals present in the Renault Zoe;
- powering the Pad and putting the ORUP over the pad in the test setup.

### 2.4.1 Execution

To reach the Z-mover state, the system needs to be connected to the high voltage battery. This is not possible in the Zoe, since the connection is made through CCS and the vehicle only enables the battery contactor when there is a certain voltage on the wires. The reason for this is to avoid high in-rush currents when the contactor is enabled. Since the wireless charging system is only able to generate a current and not a voltage, a resistor and a relay have been added. Figure 7 shows the schematic overview of this. During the pre-charge phase of the vehicle the Z mover state of the vehicle is activated, as well as the pre-charge relay. Because the resistance is known, ohms law can be used to calculate how much current needs to be delivered to get to the correct voltage.

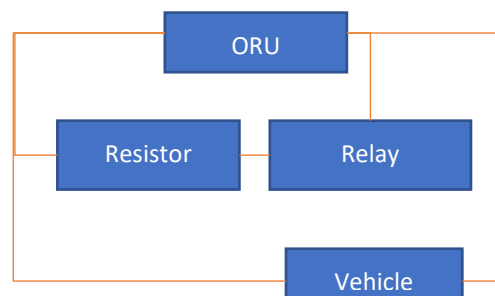


Figure 7: schematic overview of pre-charge

## 2.5 Milestone 5: Normal operation state

Milestone 5 has the following goal:

- Have power transfer from the Pad to the ORUP, using the signals from the Renault Zoe, outside of a vehicle.

### 2.5.1 Execution

This milestone was estimated to be one of the larger ones, since to get to the “Normal operation state” all safety requirements need to be met. When progressing from the “Z mover state” it was

quickly found that the “Z mover state” already checks all the safety requirements and the progress to “Normal operation state” is done automatically.

Since this is the state where actual power transfer can happen, the following was verified using the lab setup:

- Stability of the product to run longer periods of time
- Correctness of assembly (end of line tests)
- Calibration of the metal object detection
- Full functionality of the product

## 2.6 Milestone 7: CCS driver deployed

Milestone 7 has the following goals:

- Add the application developed during milestone 1 to the system.
- Verify that the application is still working as seen during milestone 1.

### 2.6.1 Execution

At the start of this milestone an automated test script was developed (see chapter 3.2.1 for more information on automated test scripts). This script has been used to verify the application at the end of this milestone.

First step in the actual development was to use the workflow defined in milestone 2 to add new messages to the CAN communication. After this the application was added using Systemdesk.

Testing was done in two steps:

1. The first step was used to verify if the ORU was running the application in the same way as the Windows computer was in milestone 1. These tests were done using the automated tests developed at the beginning of this milestone.
2. After the automated tests the application was tested with the test setup used in milestone 1. The tests specified that the application still has the same behavior as when it was running on the Windows environment.

## 2.7 Milestone 8: CCS driver integrated

Milestone 8 has the following goal:

- Have the CCS driver working in the software architecture, so that wireless energy transfer is possible based on request on the CCS connection.

### 2.7.1 Execution

Figure 8 shows the overview of how the CCS driver is integrated in the software architecture. The CCS driver is highlighted in blue, the other applications running in the ORU are highlighted in green. The CCS driver has the following interfaces:

- CAN interface to directly get signals from:
  - Display to:
    - Determine when to start charging
    - Provide information about the current status
  - CCS control board to:
    - Gather information on power requests from the vehicle
    - Communicate with the vehicle about the available power
- Safety communication manager to get information on error which could impact the safety of the system
- Vehicle adapter to gather vehicle specific information. In this case the vehicle speed.
- Communication manager to gather the current state (for example, found state, charging state, etc.) of the ORU.

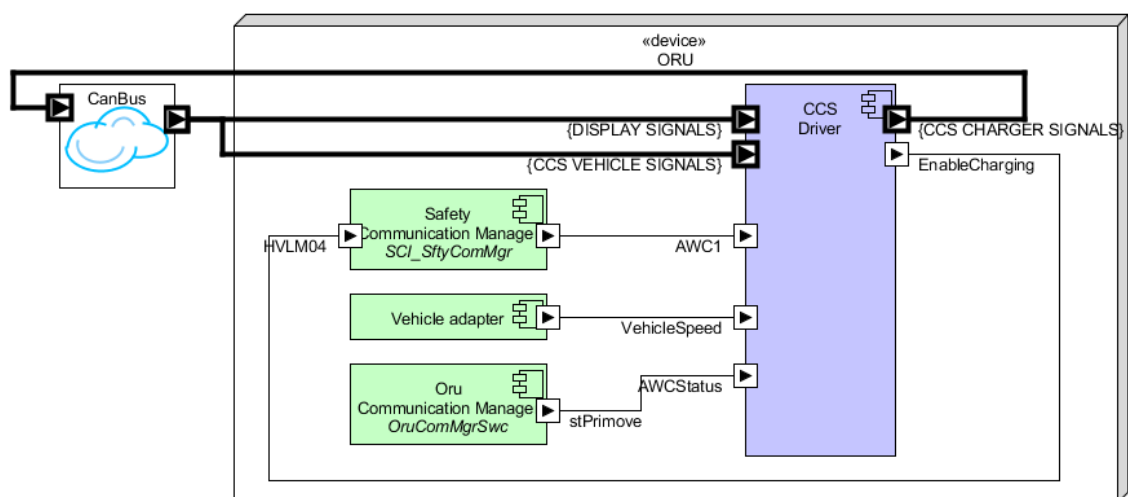


Figure 8: overview of integration of the CCS driver application

Since this part of the software is directly connected to the control of the high voltage, it was tested using the lab. See chapter 3.2.2 for more information on the lab testing.

## 2.8 Milestone 6 & 9: integration

Milestones 6 & 9 have the following goal:

- Have two (2) Zoe's ready for delivery with the system completely integrated.

### 2.8.1 Execution

Integration has been done in the smallest steps possible to embrace the fail-fast method of working, which is a way of working in software development to tackle the high-risk integration issues in an early phase of the project. In this project it meant that there were 3 moments of integration:

1. Without wireless charging
2. With wireless charging outside of the vehicle
3. With wireless charging inside of the vehicle

These moments will be discussed in the following paragraphs.

#### 2.8.1.1 Without wireless charging

The first things to verify was if it was possible to establish a connection through the CCS connection of the vehicle and to start a charging session. The CCS driver now running in the ORU was developed in such a way that it would be possible to run it using a regular Windows machine. During the onsite test, a first attempt was made to see if the assumptions when writing the first iteration of the driver were enough to start charging. In the test setup is depicted in

Figure 9, almost everything is a stand in for the future component.

1. The CCS Controller Board driver runs on Laptop 1, to become onboard receive unit's ECU,
2. The high voltage charger, to be replaced with the charge coils,
3. The charging plug will become a relay,
4. The vehicle is a not the target vehicle, since that was not available at the time,
5. The CCS Controller Board is the only component that will be built into the vehicle as is.

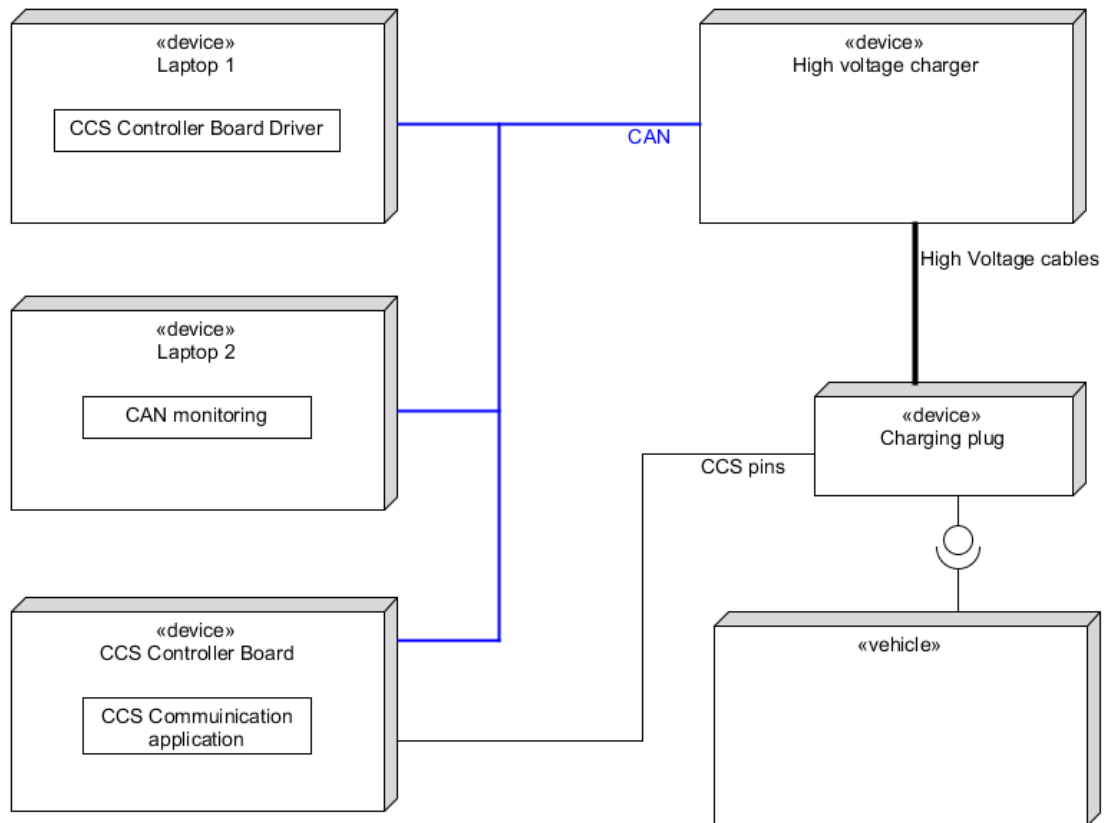


Figure 9: The test setup without wireless charging

For contrast, the final deployment setup is depicted in Figure 10. In this picture the following is described:

1. The laptop is replaced by the ECU becoming part of the onboard receive unit.
2. The charge coil replaces the high voltage charger, also part of the receive unit.
3. The charge turns into a relay, allowing for switching between the regular charge port and the wireless charging system.
4. Laptop 2 is removed.
5. The CCS controller board remains a separate component.
6. The whole is placed inside the vehicle.

Note in this picture the emergence of the Bluetooth Low Energy Driver. This is used to communicate to the Way Side Pad, to tell it to start and stop charging, replacing the CAN link to the high voltage charger in the test setup.

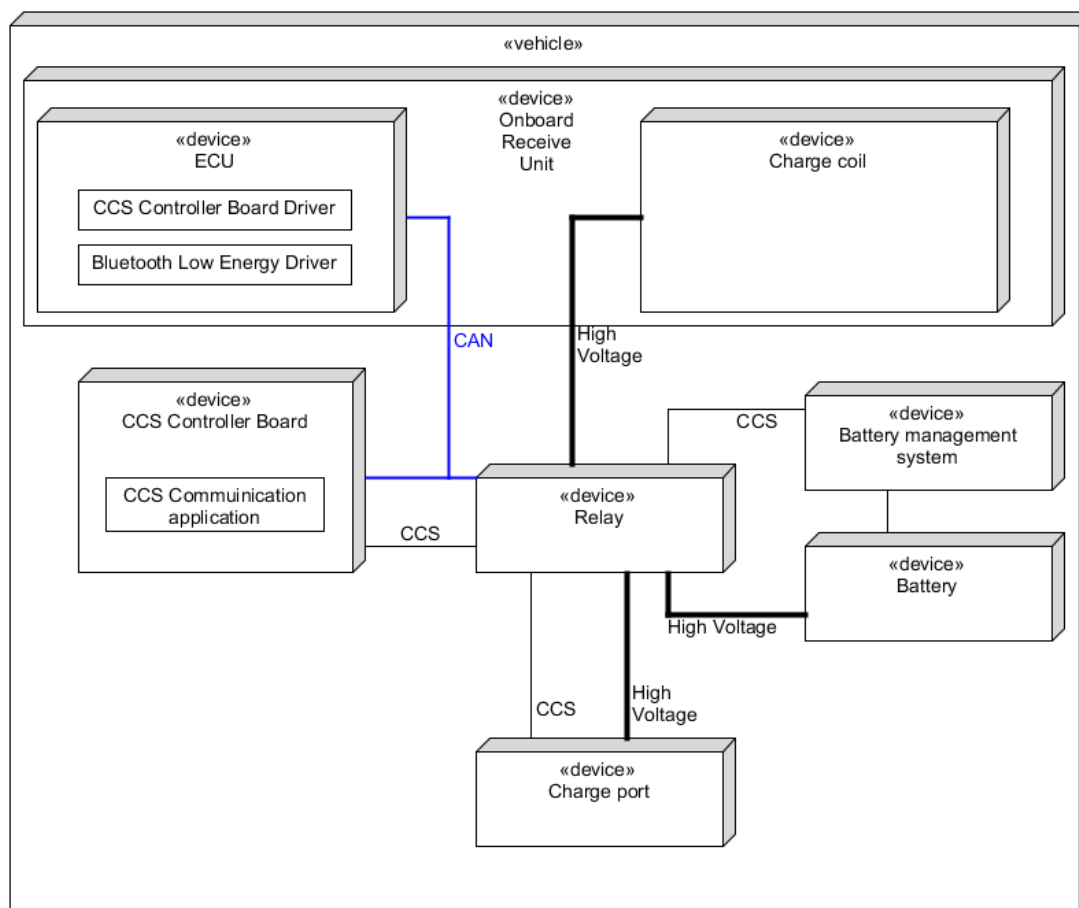


Figure 10: Deployment architecture

#### 2.8.1.1.1 First test cycle

The following was learned from the first test cycle:

- It is necessary to communicate the maximum abilities of the system to the vehicle so that the vehicle can adjust its requests accordingly
- Delivering less power than requested from the vehicle is possible
- To stop charging without physically pressing buttons in the vehicle, the emergency stop routine has to be used
- The basics of the CCS driver were working as expected



#### 2.8.1.1.2 Second test cycle

A CCS enabled Renault Zoe was borrowed from the local dealership for the second test cycle. This vehicle was used for the same tests as the first cycle. The following was learned from the second test cycle:

- The Renault Zoe is a lot more sensitive to hickups in communication
- The time for precharge in the Renault Zoe is only 10 seconds
- If establishing the connection between the Renault Zoe and the charging system fails, the vehicle will go to a fault state which can only be recovered at the dealership.
- The flap in front of the CCS socket needs to be open (or the vehicle has to think it is open) in order to charge
- The CCS driver is working as expected

### 2.8.1.2 With wireless charging outside of the vehicle

For the second step the vehicle was involved, but still without installing the system into the vehicle. This way the system can be tuned to the vehicle, without making any adjustments which need to be patched later on. The system looks like Figure 1: system architecture, with the exception that the CAN bus was simulated and the high voltage wiring and CCS communication was done over a DC charging cable. Figure 11 shows the test setup as it was used for these tests.



Figure 11: test setup with wireless charging outside of the vehicle

The following was learned from these tests:

- The concept will work when it is built into the vehicle after small modifications
- Pre-charge speed is fast enough
- Pre-charge is accurate enough
- Vehicle is accepting the power delivery characteristics of the charging solution
- Timing of the mechanism which retries to establish connection needs to be changed

### 2.8.1.3 With wireless charging inside of the vehicle

The last step of integration was putting all the components into the vehicle itself. The following has been adjusted to the vehicle:

- The ORU has been mounted underneath the vehicle.
- On the left side of the vehicle a connection has been made to the wiring harness which goes into the front bumper. This is so that the system is able to detect opening or closing of the flap covering the charge port and to disable opening and closing the flap. This is done so that all parts which can have high voltage present are shielded from the user.
- On the top of the charge port a connection has been made in between the CCS communication pins. This is done so that the CCS communication pins can be routed to the wireless charging system, while still maintaining the possibility to charge over CCS.
- The junction box is located on the right side of the front bumper. Figure 12 shows the junction box mounted and opened with the front bumper removed.

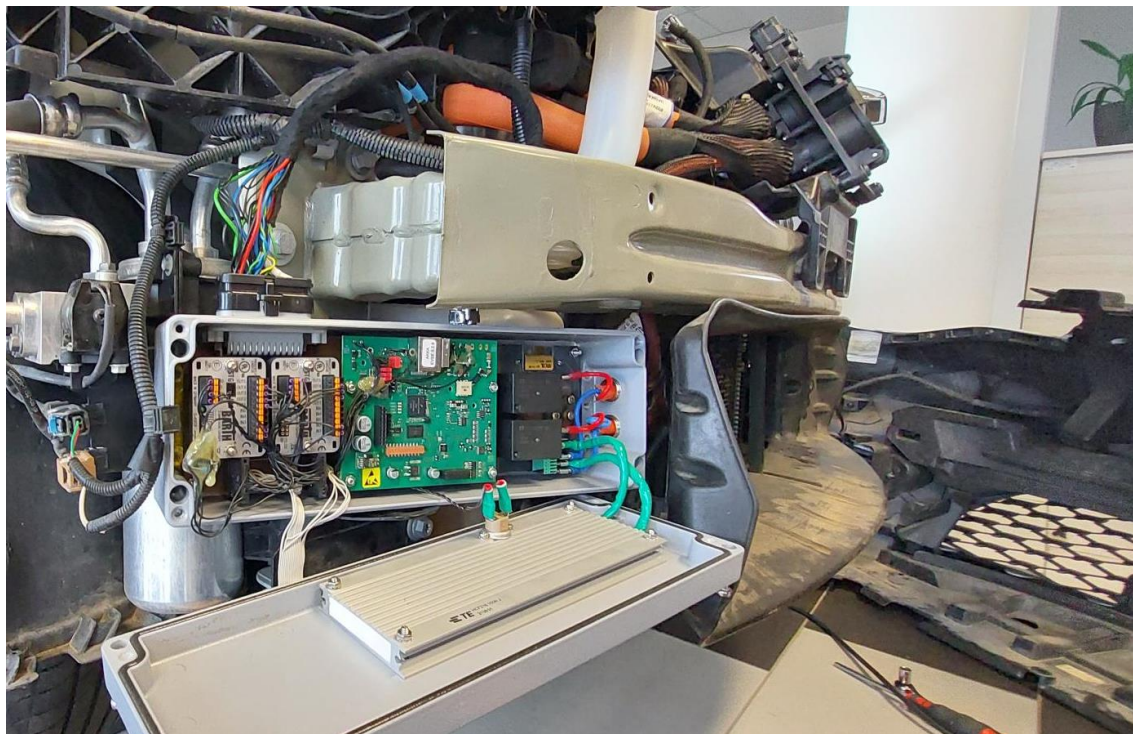


Figure 12: location of the junction box

- The gateway connection the CAN bus of the wireless charging system to the CAN bus of the vehicle is located behind the instrument cluster
- The screen (Figure 13: additional screen interface).



Figure 13: additional screen interface

## 3. Testing

Testing has been done in three levels:

- Software in the loop tests
- Hardware in the loop tests
- Integration testing using the vehicle

These three levels will be explained in the next paragraphs.

### 3.1 Software in the loop

During this project, software in the loop tests were executed using Ceedling. This is a framework to fully automate software only tests. The main advantage of automating these tests is that the tests are easily executed repeatedly. Since all tests can run easily, tests will run more often, and bugs, regressions and integration issues will be found quickly. The main disadvantage is that the integration with hardware cannot be tested.

Tests are written according to the Arrange-Act-Assert pattern. Each test function is grouped by these functional sections, separated by blank lines:

- Arrange all necessary preconditions and inputs.
- Act on the object or method under test.
- Assert that the expected results have occurred.

An example of output generated by Ceedling can be found in Figure 14: example output of Ceedling run.



Test 'test_AutosarAdapter.c'
Generating runner for test_AutosarAdapter.c...
Compiling test_AutosarAdapter_runner.c...
Linking test_AutosarAdapter.out...
Running test_AutosarAdapter.out...
Test 'test_CcsDriver.c'
Generating runner for test_CcsDriver.c...
Compiling test_CcsDriver_runner.c...
Linking test_CcsDriver.out...
Running test_CcsDriver.out...
GCOV: IGNORED TEST SUMMARY
[test_AutosarAdapter.c]
Test: test_TresosWrapper_NeedToImplement
At line (19): "Need to Implement AUTOSAR Adapter"
GCOV: OVERALL TEST SUMMARY
TESTED: 9
PASSED: 8
FAILED: 0
IGNORED: 1
GCOV: CODE COVERAGE SUMMARY
AutosarAdapter.c Lines executed:0.00% of 140
AutosarAdapter.c Branches executed:0.00% of 18
AutosarAdapter.c Taken at least once:0.00% of 18
AutosarAdapter.c Calls executed:0.00% of 50
CcsDriver.c Lines executed:73.97% of 146
CcsDriver.c Branches executed:96.55% of 58
CcsDriver.c Taken at least once:81.03% of 58
CcsDriver.c Calls executed:75.56% of 45

Figure 14: example output of Ceedling run

## 3.2 Hardware in the loop

Hardware in the loop tests have been executed on two levels:

1. On desk
2. In the lab

The main reason for this is that testing on desk cannot include high voltage. It is added however because testing in the lab requires more preparation time while it should be possible to do testing quick and often. Both setups will be discussed in the following paragraphs.

### 3.2.1 On-desk

Figure 15: On-desk test setup shows the on-desk test setup. With this test setup it is only possible to test the low-voltage part of the system. It includes the following parts:

- The low-voltage part of the ORU
- All electronics which are installed in the vehicle
- A board which allows the user to communicate with the system through CCS. This board can be controlled over CAN using the CANoe simulation in Figure 16: CANoe simulation for CCS supply equipment board. The vehicle can be simulated with the CANoe simulation shown in Figure 17: Vehicle simulation.

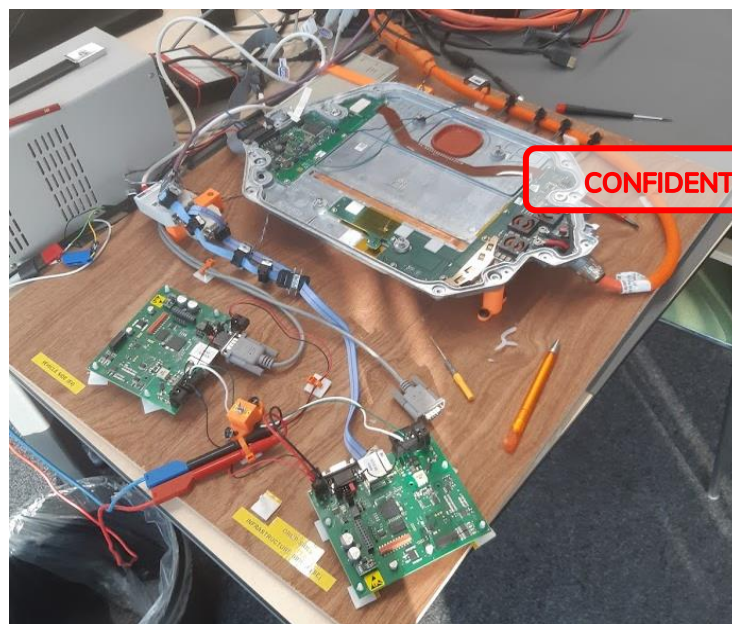
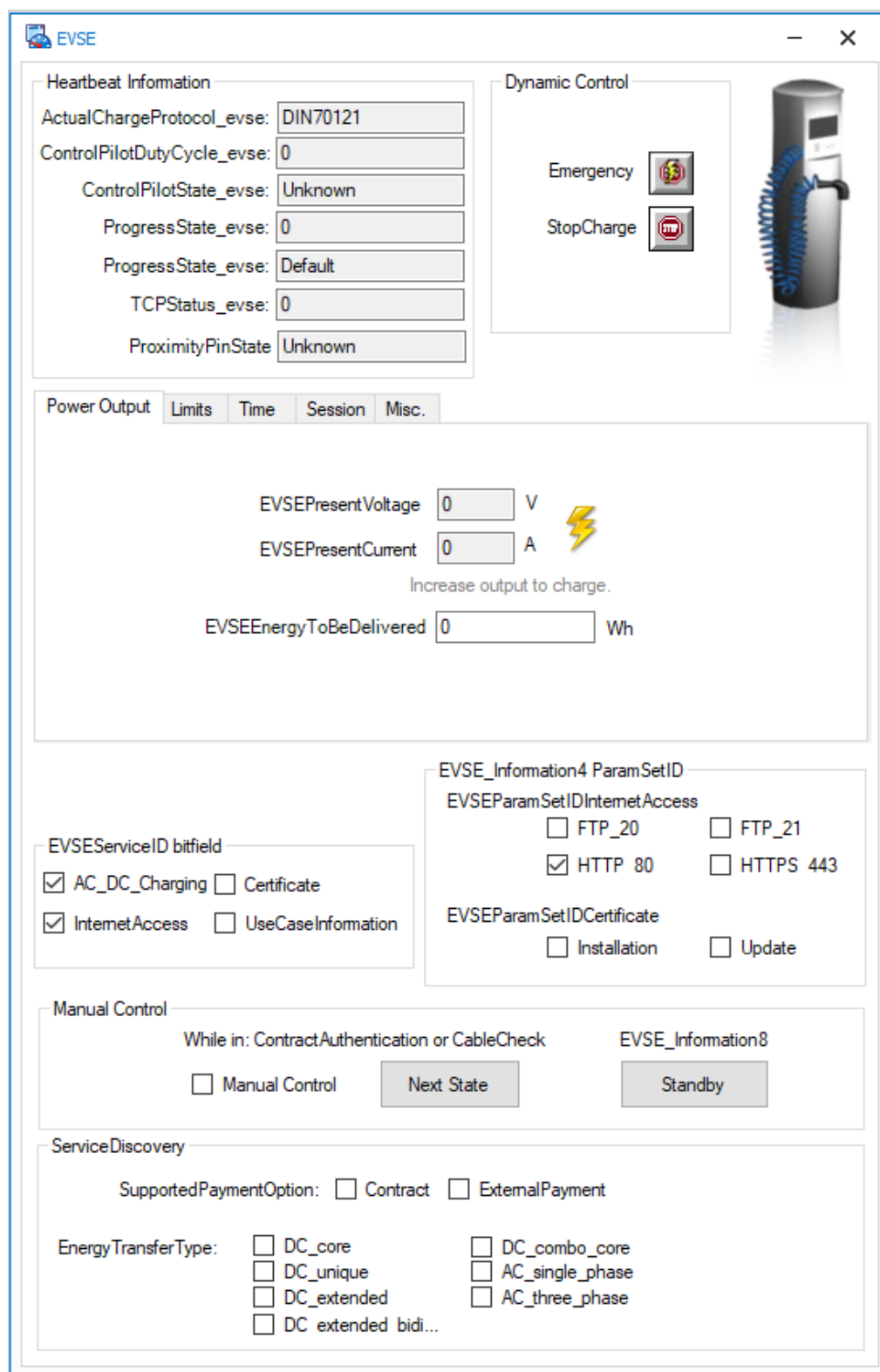


Figure 15: On-desk test setup



**Heartbeat Information**

ActualChargeProtocol\_evse: DIN70121

ControlPilotDutyCycle\_evse: 0

ControlPilotState\_evse: Unknown


ProgressState\_evse: 0


ProgressState\_evse: Default

TCPStatus\_evse: 0

ProximityPinState: Unknown

**Dynamic Control**

Emergency 

StopCharge 

**Power Output** Limits Time Session Misc.

EVSEPresentVoltage 0 V

EVSEPresentCurrent 0 A

Increase output to charge.

EVSEEnergyToBeDelivered 0 Wh

**EVSEServiceID bitfield**

☒ AC\_DC\_Charging ☐ Certificate

☒ InternetAccess ☐ UseCaseInformation

**EVSE\_Information4 ParamSetID**

EVSEParamSetIDInternetAccess

☐ FTP\_20 ☐ FTP\_21

☒ HTTP 80 ☐ HTTPS 443

EVSEParamSetIDCertificate

☐ Installation ☐ Update

**Manual Control**

While in: ContractAuthentication or CableCheck

☐ Manual Control

**ServiceDiscovery**

SupportedPaymentOption: ☐ Contract ☐ ExternalPayment

EnergyTransferType:

☐ DC\_core ☐ DC\_combo\_core

☐ DC\_unique ☐ AC\_single\_phase

☐ DC\_extended ☐ AC\_three\_phase

☐ DC\_extended bidi...

Figure 16: CANoe simulation for CCS supply equipment board

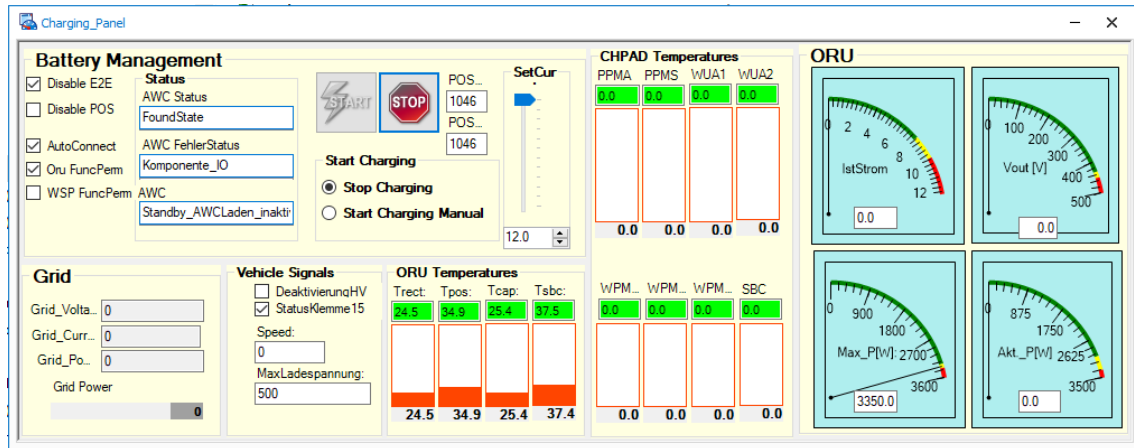


Figure 17: Vehicle simulation

### 3.2.2 In the lab

If testing with high voltage is needed, it is done in the lab. The lab setup is shown in Figure 18: Lab test setup. The following components are in the lab setup:

- Battery simulator, which consists of a load and a power supply, which can be manually set to a certain battery state.
- Vehicle simulator, divided into three components:
  - CAN simulation, which is the same as used in the on-desk testing
  - CCS simulation, which is the same as used in the on-desk testing
  - Computer controlling simulations
- Wireless charging system, including all components which will be built into a car.



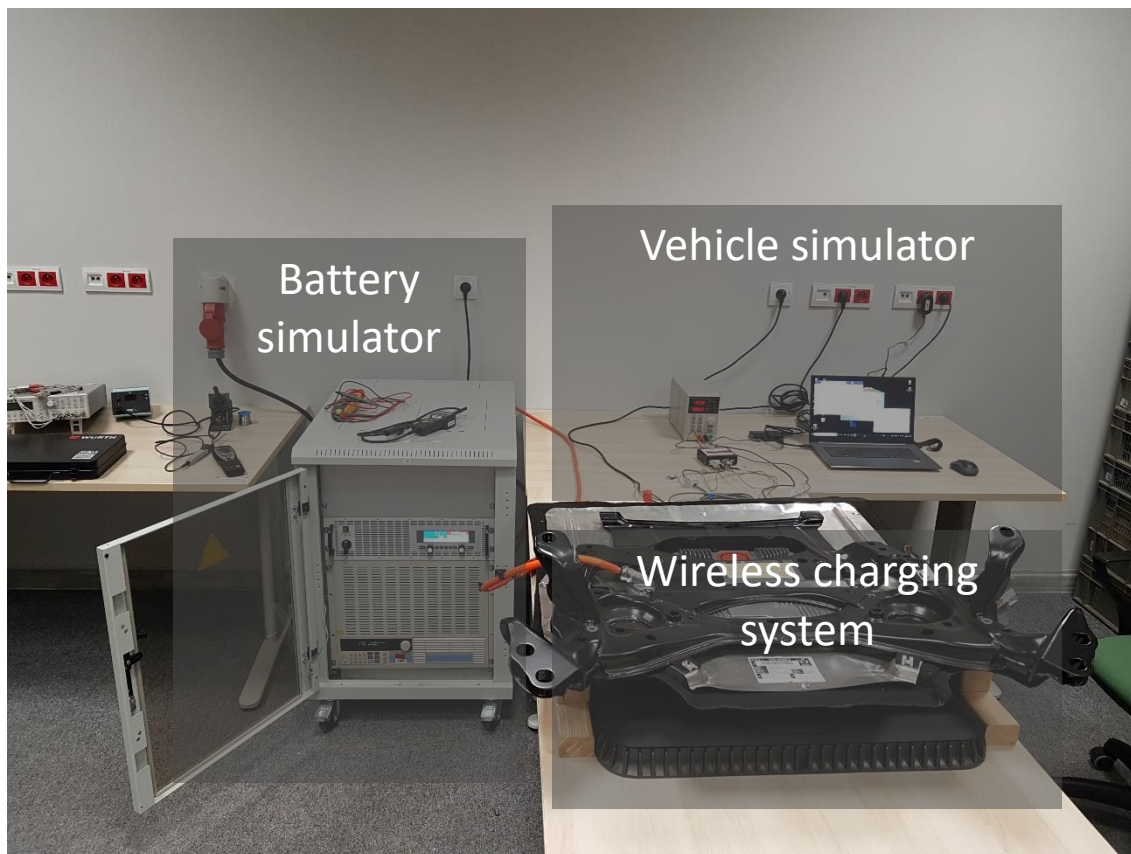


Figure 18: Lab test setup

Due to safety reasons lab tests are executed manually. This should be automated in a future project after safety analysis to save testing time.

### 3.3 Integration testing

A space has been reserved for the last integration testing with the vehicle. In Figure 19: vehicle during integration testing, the vehicle is shown in this location during integration testing. The following things are special in this setup for the integration tests:

- The PAD is placed on a slippery surface so that it can be moved to test specific positioning use-cases.
- Part of the bumper has been removed. This is done to be able to access the debug connectors going to both the connection box and the ORU.
- Special wiring has been used for the pad to provide the test engineer with the internal CAN bus for debugging and flashing of the PAD.



Figure 19: vehicle during integration testing

All tests specified in the test plan have been done using this setup.

## 4. Conclusion

After integration and installation at the parking of the AMB offices (shown in Figure 20: Final installation of the charger at the AMB parking offices), and after reviewing the feedback from AMB, the following conclusions have been stated:

- Integrating a wireless charging system through CCS as an aftermarket option is possible, although it still needs to be tailored to the specific vehicle. This means that an universal solution is possible with minor adaptations.
- Being able to charge a vehicle wirelessly without handling cables adds to the 'luxurious feel' of an electric vehicle.
- Since the vehicles in this project are built as proof of concept, it is expected that issues may occur during the adaption phase. This means two things:
  1. The test phase shall partly be implemented during the current months and the consortium partners shall stay open to communicate any new problem that could be reported from the side of the AMB to bring the system to a next level of maturity.
  2. To bring the system to an universal solution to be implemented through the CCS to any vehicle having this feature, further testing should be implemented preferably on the following levels:
    - Automated unit testing
    - Automated integration testing
    - Integration testing with an actual vehicle, preferably one which is tailored to the needs of the engineering department
- The user interface does fulfil the principal main objectives. The vehicle positioning and the ability to start and stop the charging process are accessible. For some of the extra information flow and interactions with the user, the system leans on the standard display of the vehicle, for example the information on the current state and what the system is doing, but and expansions of the functionalities of the user interface are still possible. Currently the following information states could be recommended:
  - The current state of the system:
    - Charging
    - Not charging, but connected to the vehicle
    - Not connected to the vehicle, but Bluetooth connection available

- The system needs some time to establish connection to the vehicle. Right now, this is not shown in the user interface. The current step in establishing connection should be shown in the future.
- Measurements on efficiency are only executed in lab tests. Measurements on efficiency when the system is installed in the vehicle is still an open point for the future implementation, when the product shall be brought to series production activities.

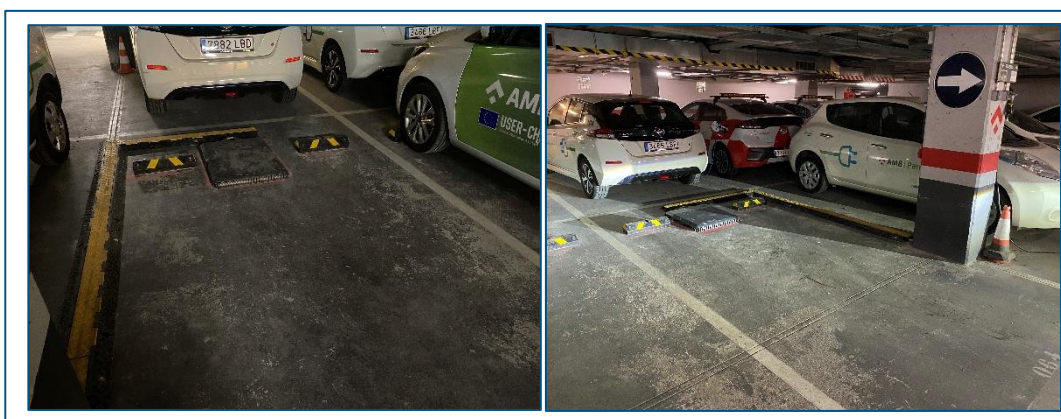


Figure 20: Final installation of the charger at the AMB parking offices

# 5. References and acronyms

## 5.1 References

- [1] CETP.00.00-E-412 – Corporate Engineering Test Procedure Title: Electrical and Electronic Component Environmental CETP: 00.00-E-412 Compatibility Test.
- [2] Directive 2004/104/EC – Commission Directive 2004/104/EC of 14 October 2004 European Automotive EMC directive.
- [3] FMVSS 305 – National Highway Traffic Safety Administration (NHTSA) - Federal Motor Vehicle Safety Standard (FMVSS) No. 305 “Electric-powered vehicles”.
- [4] ICNIRP 2010 – ICNIRP Guidelines for Limiting Exposure to Time-Varying Electric and Magnetic Fields (1hz – 100 kHz) International Commission on Non-Ionizing Radiation Protection.
- [5] IEC 60068-2-52 – International Standard Environmental testing – Part 2-52: Tests – Test Kb: Salt mist, cyclic (sodium chloride solution).
- [6] IEC 60364-7-722:2018 – Low-voltage electrical installations - Part 7-722: Requirements for special installations or locations - Supplies for electric vehicles.
- [7] IEC 60529:1989+AMD1:1999+AMD2:2013 CSV – Consolidated version - Degrees of protection provided by enclosures (IP Code).
- [8] IEC 61000-3-11:2017 RLV – Redline version – Electromagnetic compatibility (EMC) – Part 3-11: Limits – Limitation of voltage changes, voltage fluctuations and flicker in public low-voltage supply systems - Equipment with rated current  $\leq 75$  A and subject to conditional connection.
- [9] IEC 61000-3-12:2011 - Electromagnetic compatibility (EMC) - Part 3-12: Limits - Limits for harmonic currents produced by equipment connected to public low-voltage systems with input current  $>16$  A and  $\leq 75$  A per phase.
- [10] IEC 61000-3-2:2018+AMD1:2020 CSV – Consolidated version – Electromagnetic compatibility (EMC) – Part 3-2: Limits – Limits for harmonic current emissions (equipment input current  $\leq 16$  A per phase).
- [11] IEC 61000-3-3:2013+AMD1:2017 CSV – Consolidated version – Electromagnetic compatibility (EMC) – Part 3-3: Limits – Limitation of voltage changes, voltage fluctuations and flicker in public low-voltage supply systems, for equipment with rated current  $\leq 16$  A per phase and not subject to conditional connection.
- [12] IEC 61000-4-2:2008 – Electromagnetic compatibility (EMC) – Part 4-2: Testing and measurement techniques – Electrostatic discharge immunity test.
- [13] IEC 61000-4-3:2020 – Electromagnetic compatibility (EMC) – Part 4-3: Testing and measurement techniques – Radiated, radio-frequency, electromagnetic field immunity test.



- [14] IEC 61000-4-4:2012 RLV Redline version – Electromagnetic compatibility (EMC) – Part 4-4: Testing and measurement techniques – Electrical fast transient/burst immunity test.
- [15] IEC 61000-4-5:2014+AMD1:2017 CSV Consolidated version – Electromagnetic compatibility (EMC) – Part 4-5: Testing and measurement techniques – Surge immunity test.
- [16] IEC 61000-4-6:2013 – Electromagnetic compatibility (EMC) – Part 4-6: Testing and measurement techniques – Immunity to conducted disturbances, induced by radio – frequency fields.
- [17] IEC 61439-5:2014 – Low-voltage switchgear and control gear assemblies – Part 5: Assemblies for power distribution in public networks.
- [18] IEC 61439-7:2018 – Low-voltage switchgear and control gear assemblies – Part 7: Assemblies for specific applications such as marinas, camping sites, market squares, electric vehicle charging stations.
- [19] IEC 61508:2010 CMV Commented version – Functional safety of electrical/ electronic/ programmable electronic safety-related systems – Parts 1 to 7 together with a Commented version.
- [20] IEC 61851-1:2017 – Electric vehicle conductive charging system – Part 1: General requirements.
- [21] IEC 61851-21-1:2017 – Electric vehicle conductive charging system – Part 21-1 Electric vehicle on-board charger EMC requirements for conductive connection to AC/DC supply.
- [22] IEC 61851-21-2:2018 – Electric vehicle conductive charging system – Part 21-2: Electric vehicle requirements for conductive connection to an AC/DC supply – EMC requirements for off board electric vehicle charging systems.
- [23] IEC 61851-23:2014 – Electric vehicle conductive charging system – Part 23: DC electric vehicle charging station.
- [24] IEC 61980-1:2020 – Electric vehicle wireless power transfer (WPT) systems.
- [25] IEC 61980-1:2020 – Electric vehicle wireless power transfer (WPT) systems – Part 1: General requirements.
- [26] IPC-J-STD-001H – Requirements for Soldered Electrical and Electronic Assemblies.
- [27] IPC-2221 – Generic Standard on Printed Board Design.
- [28] IPC-A-610 – Acceptability of Electronics Assemblies.
- [29] ISO 15118 – Road vehicles - Vehicle to grid communication interface
- [30] ISO 16750 – Road vehicles – Environmental conditions and electrical testing for electrical and electronic equipment
- [31] ISO 20653:2013 – Road vehicles – Degrees of protection (IP code) – Protection of electrical equipment against foreign objects, water and access.
- [32] ISO 26262 – Road vehicles – Functional safety.

- [33] ISO 6469 – Electrically propelled road vehicles. Safety specifications.
- [34] ISO/IEC 17409:2020 – Electrically propelled road vehicles – Conductive power transfer – Safety requirements.
- [35] ISO/IEC 19363:2020 – Electrically propelled road vehicles – Magnetic field wireless power transfer – Safety and interoperability requirements.
- [36] SAE J2894 – Power Quality Requirements for Plug-In Electric Vehicle Chargers
- [37] UNECE R.100 – Uniform provisions concerning the approval of vehicles with regard to specific requirements for the electric power train Addendum 99: Regulation No. 100.

## 5.2 Acronyms

Acronym	Meaning
AMB	Àrea Metropolitana de Barcelona
BIT	Built In Test
CBIT	Continuous Built In Test
D	Deliverable
DE	Deutschland / Germany
DTC	Diagnostic Trouble Code
EC	European Commission
ETRA	ETRA I+D (project partner)
EVCC	Electric Vehicle Communications Controller
EMC	Electro Magnetic Compatibility
FOD	Foreign Object Detection
FMVSS	Federal Motor Vehicle Safety Standard
GA	Grant Agreement
HV	High Voltage
IBIT	Initialisation Built In Test
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
LV	Low Voltage
O	Objective
MOD	Metal Object Detection
P	Product
PBIT	Power down Built In Test
RCD	Residual-current device
SECC	Supply Equipment Communications Controller
SOW	Statement of Work
USER-CHI	Project Title: innovative solution for USER centric CHarging Infrastructure
VIN	Vehicle Identification Number
WP	Work Package
WPT	Wireless Power Transfer