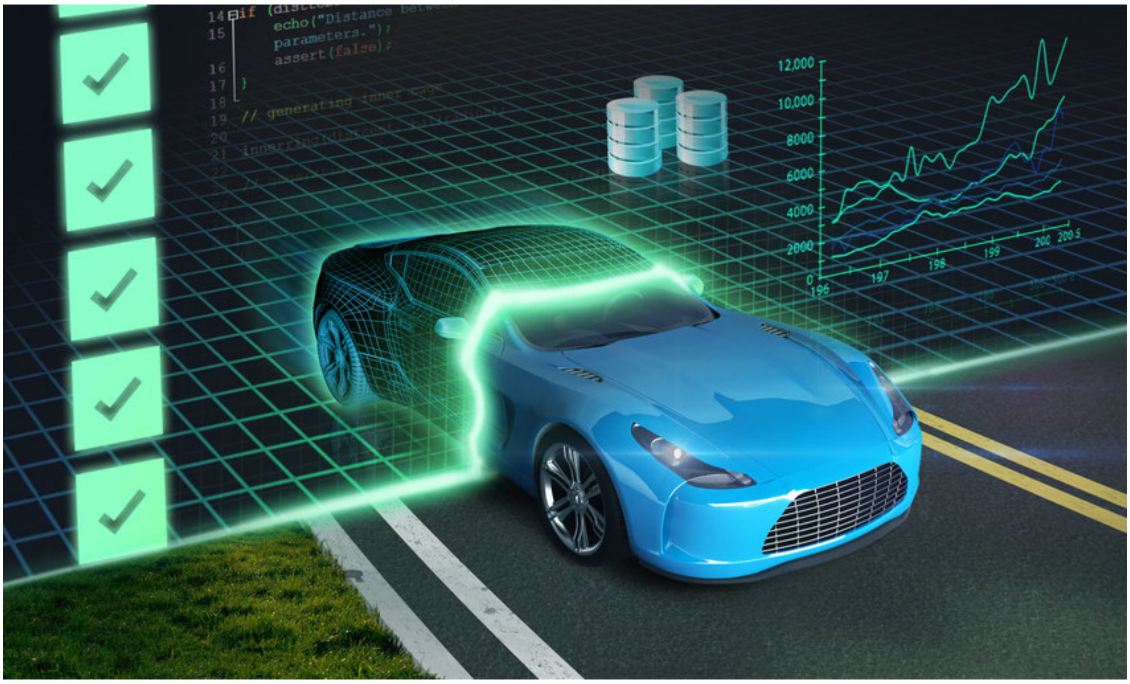
CHALMERS
UNIVERSITY OF TECHNOLOGY



Software-in-the-Loop Techniques for Thermal Control in Electric Vehicles

Master's thesis in System, Control, and Mechatronics

GHASSAN ABDELKHALEQ

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022
www.chalmers.se

Master's thesis 2022

Software-in-the-Loop Techniques for Thermal Control in Electric Vehicles

[1cm] Ghassan Abdelkhaleq

Department of Electrical Engineering
Division of Automation
Chalmers University of Technology
Gothenburg, Sweden 2022

Software-in-the-Loop Techniques for Thermal Control in Electric Vehicles
Ghassan Abdelkhaleq

Supervisor: Yashasvi Nandivada, Thermal Managment System @Volvo Cars
Examiner: Full Prof. Martin Fabian, Department

Master's Thesis 2022
Department of Electrical Engineering
Division of Automation
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Software in the loop testing using Silver.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2022

Software-in-the-Loop Techniques for Thermal Control in Electrical Vehicles
GHASSAN ABDELKHALEQ
Department of Electrical Engineering
Chalmers University of Technology

Abstract

Model-in-the-Loop and Software-in-the-Loop tests are two methods used to virtually test the set up of the model prior to the actual installation on the vehicle. Those advantageous methods can help in saving time and cost since testing can start already in parallel with the development of the system. This in return gives more time to find faults and debug in an early stage before implementing on the vehicle. Currently, at Thermal System Development, a technique has been developed to perform Model-in-the-Loop testing. Software-in-the-Loop testing will be implemented in Silver to evaluate how the parameters talk to each other between the model and production software. Thus, Software-in-the-Loop testing will be the main goal of this work.

The goal of this thesis is to set a virtual environment for models in Simulink in the software testing platform called Silver. Those models are developed in the the thermal management system which involves various pumps and valves to direct the flow of coolant towards different vehicle components. For instance, control the fan or pump speed used to cool down the battery temperature. In addition, cooling or heating the 3 main circuits in the thermal management system, battery circuits, electrical circuits and climate circuits. Therefore, the SiL will be a handy tool to spot the faults in the developed control module before we go for in-vehicle testing.

Keywords: Software in the Loop, Thermal Management System, Drive Cycle, Electric Vehicle

Acknowledgements

In the beginning, I would like to thank my supervisor Full Prof. Martin Fabian for his constructive guidance and approaches throughout the master thesis work. Moreover, I would like to express my gratitude to my industrial supervisor Yashasvi Nandivada for giving me his ultimate support and always finding time for consultations. Beside, always being open to discuss about different matters on the personal and technical level. In addition, I would take the chance to thank my industrial team manager Roger Alexendersson for his assistant and support throughout the thesis.

Ghassan Abdelkhaleq, Gothenburg, 09 2022

Nomenclature

Below is the nomenclature of indices, sets, parameters, and variables that have been used throughout this thesis.

Abbreviations

EU	European Union
$Mtoe$	Mega tonnes of Oil Equivalent
ETS	Emissions Trading System
BEV	Battery Electric Vehicles
MiL	Model in The Loop
SiL	Software in The Loop
HiL	Hardware in The Loop
MPC	Model Predictive Control
$HLCM$	High to Low Voltage Converter Module
OBC	On-board Charger
BMS	Battery Management System
PFC	Power Factor Correction
$BESS$	Battery Energy Storage Systems
SSC	System Supervisory Control
AC	Alternative Current
DC	Direct Current
VCU	Vehicle control unit
$HVCH$	High Voltage Coolant Heater
$HVAC$	Heating, Ventilating, and Air Conditioning
SPM	Production Software
$NEDC$	New European Driving Cycle
$WLTP$	Worldwide Harmonised Light Vehicle Test Procedure

Contents

List of Figures

1
Introduction

1.1 Background

Energy consumption evaluation is an important step toward complete vehicle testing. This step is important as the percentage of CO2 is increasing globally, while the EU is taking preventive measures towards reducing CO2 consumption. EU has committed itself to reduce 20 % energy consumption reduction by the year 2020, and this includes all kinds of sectors such as automotive, power water desalination, and much more. As shown in the figure below, Sweden was one of the leading countries for this change by a 75% reduction between the years 2017 and 2019. Those plans are aimed to continue developing to blaze the road toward an eco-friendly plant. By 2030 the 28 countries have a goal of getting a 32.5% reduction in energy consumption. This amount is equivalent to the final energy consumption of no more than 956 Mtoe. [1]

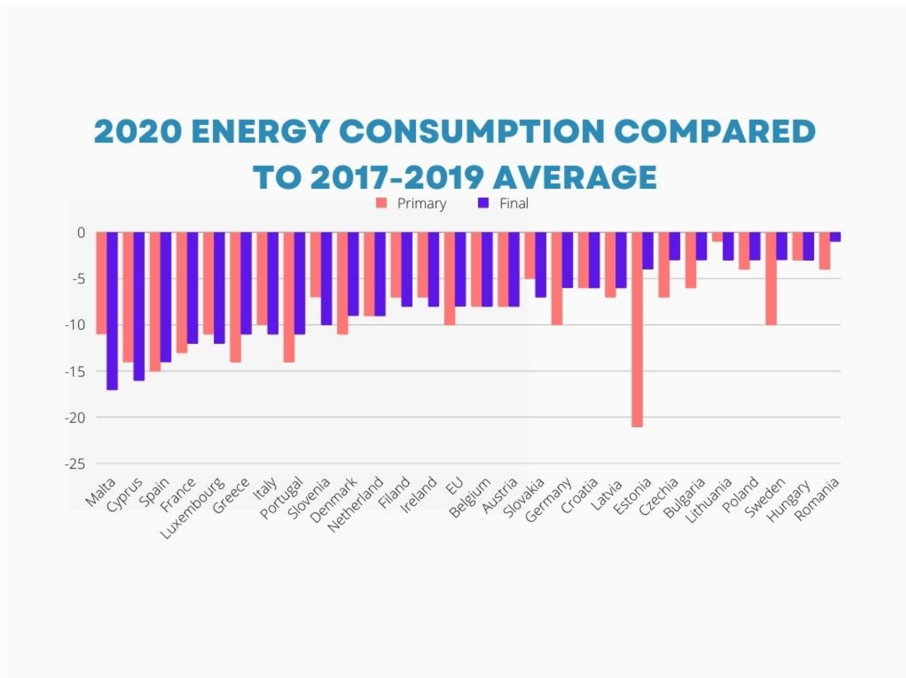


Figure 1.1: Energy consumption in EU [1]

Throughout the years, the transportation sector had the biggest portion of the EU's CO2 emissions with almost 30%, where 72% of them is from road transportation.[5]

As a result, the EU announced that by 2050 this percentage would decrease to 12% with the effort of the Emissions Trading System "ETS."[6] The ETS is the first organization in the world that obligates companies to have a valid permit to emit one tonne of CO2, where companies have to purchase them through auctions. It was proven that the ETS about 40% of the EU's emissions and gives compensations to develop innovation in the sector.[7]

To narrow it down, cars, in general, are the major and main polluters, which comprise around 60% of the total CO2 emissions by transportation around Europe. [5]. But is BEV better than ICE? The answer is yes, and here is why. Many factors make BEV a better choice for the consumer. Firstly, efficiency-wise. The fuel efficiency for ICE is approximated to be around 40 %-60% compared to almost 90% in the BEV.[8] Therefore, the amount of energy used to cross the same distance is less in the BEV than in ICE. Secondly, the running cost is relatively less compared to ICE. for instance, BEV has fewer dynamic components and less metal-to-metal friction. For example, no need for engine oil, gasket, and spark plug change. Lastly, it's more environmentally friendly. Even though BEV production comes with environmental challenges but still it's way cleaner than BEV. A study made by Reuters shows that a Tesla Model 3 needs to drive 13500 km until it reaches the breakdown point. In other words, the tesla model needs to drive the mentioned distance to reach the environmental effect that a brand new Toyota Corolla has.[9]

1.2 Scope

In this thesis, we are interested in implementing SiL techniques for the Thermal Management System, which involves various pumps and valves to direct coolant flow toward different vehicle components. The objective here is to control the cooling and heating of different parts, for instance, the electrical battery and parts of the climate circuit of the system. The thermal management function regulates the temperature in those components inside the vehicle. In this specific case, it is about faults in the developed control models. SiL setup will help identify the faults in the developed code/controls SW before we go for in-vehicle testing. Therefore, Model-in-the-Loop and Software-in-the-Loop tests perform the virtual tests before the system is tested on the car. This can help us save time and cost, where we can already start testing in parallel while developing the system. This, in return, gives more time to find errors and debug them in an early stage before implementing them on the vehicle. Currently, a technique has been developed in thermal system development to perform Model-in-Loop testing.

1.3 Overview

The thermal management systems contain two main parts that work closely and are complementary to each other. The thermal hardware, as well as thermal software, combined, leads to a robust and reliable thermal management system. The Thermal Hardware is modeled in Software tools and linked with the Software Models for creating Model-in-the-Loop and Software-in-the-Loop platforms. This enables the users to design better future systems through early testing, less hardware use, less manpower, and faster validation. However, as part of the thesis, we are not re-inventing the wheel of either developing the hardware or the Software models. MiL will be implemented through Simulink; this includes the harness of the battery circuit, electric circuit, and climate circuit. The SiL will be tested in Silver, allowings easier and faster communication between Simulink and the production software. Harness and production software will be in a Dynamic-link library "dll" extension to recognize all inputs and outputs.

1.4 Limitations

The main purpose of the thesis is to test and validate the results from the SiL environment in Silver. Even though testing will be done on the harness and the production software, there will probably not be sufficient time to do the SiL on the production software model. This means this thesis scope can be extended to validate the model blocks for the controller built on Simulink.

1.5 Timeline

The work done for this thesis will be presented in 5 chapters, excluding references, a list of abbreviations, and an abstract. Introduction chapter 1 illustrates a brief description of the work conducted and some obstacles that may be present. Theory, chapter 2 is divided into two sections literature review and a technical review. It contains literature on software-in-the-loop techniques. In addition, elaboration of some specific parts in the battery circuits, electrical circuits, and climate circuits. Methodology chapter 3 details the methods used and needed to solve the current problem, including creating the SiL environment. Chapter 4 presents the data found and some reasons to support them. In conclusion, chapter 5 shows a conclusion drawn from the whole work that has been done.

2

Theory

2.1 Software-in-the-Loop

Software-in-the-loop (SIL) approaches companies and industries from different perspectives and is developing. It is used mainly for testing and validation purposes of the script. Therefore, building a simulation environment allows users to identify bugs and errors and cost-effectively enhance the script features. The SiL testing is ranked in the initial phases of the software quality development model, while other testing approaches like HiL are made in later phases. OEMs are increasingly interested in this testing since it allows them to investigate real-time functions through the software-based built environment. This is due to the entire virtual format that can be built, enabling faster testing than real-time testing. Moreover, it gives the engineers the ability to simulate thousands of miles of driving in a variety of real-time scenarios.[10]

Nowadays, different software programs require a lot of specific requirements this is where the idea of the SiL came from. Since it is hard to try and validate each requirement to ensure it does what it is supposed to. Therefore, the advantages of SiL, such as cost efficiency, time-saving, and practically easier to test thousands of kilometers on the environment than on the track in different scenarios, were a huge help. Big companies in the automotive sector perform daily application testing. Starting from writing codes to the actual validation. This can include integration, deployment, and testing.

There is plenty of advantage of SiL, especially in the automotive industry. That is why it is widely spread. When using SiL, all the simulations can be done on a normal computer with basic standards, making simulations that take days only takes a couple of minutes. Moreover, there is no need for high-specification testing equipment like in HiL testing. Therefore, the costs of building a SiL environment were significantly reduced compared to other testing types. Consequently, decreasing testing bottlenecks and speeding up the development process. Since the simulation can be performed completely on the software, the testing time can be adjusted even faster than in real-time.

SiL is considered a flexible and repeatable way of testing, giving testers a wide range of options and paths. Testers can perform many simulation scenarios and fix one parameter while others in the same simulation stay constant. Moreover, make a constructive feedback loop with the other software developers. In addition, SiL gives a feature for developers, which is to decouple the software and hardware development, where the developers are connected and are working on the same system together. Therefore, creating features and functions during the process without waiting for the entire product to be ready. It allows the tester to perform multiple tests simultaneously rather than sequentially, improving the time and efficiency. Furthermore, after gaining data and results from SiL testing, those results can be used again in HiL tests to monitor physical hardware performance and cross-correlations.

The data and results collected from the tests can be easily shared with the development team. The quality of the SiL highly depends on the software used for modeling, testing, and scripting to simulate a specific environment in a specific condition.

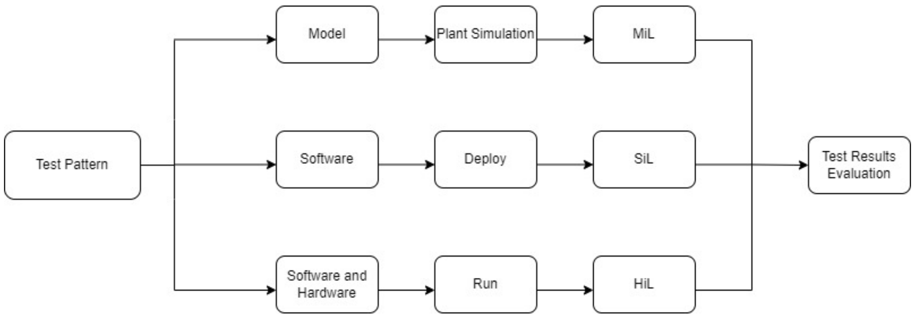


Figure 2.1: Testing approaches

2.2 System Layout

When discussing the BEV coolant system, there are usually primary circuits in discussion. Battery circuit, electric drive circuit, and climate circuit check figure 2.2. Those circuits are connected with a mix of solenoid valves and valves to mix the coolant temperature.
Generally, in most of the BEV the electric drive circuit consist of main component such as the inverters, OBC, converter and more. Components in this circuit are the second highest to withstand temperature, compared with the other 2 circuits. Second, the climate circuit, mainly used to cool down the different component such as radiator. This circuit is the highest to withstand temperature. The last circuit, the battery circuit, containing the battery module, it is the least circuit to withstand heat.

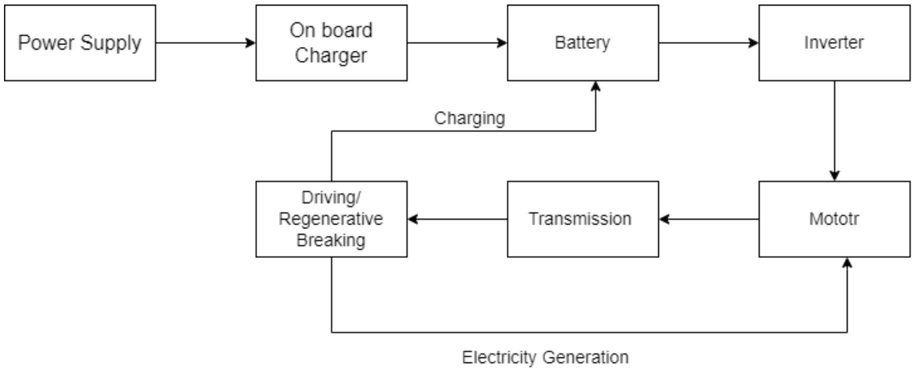


Figure 2.2: EV layout [2]

For instance, those are examples of some component in the BEV circuits in the drivetrain, energy storage, and conversion in BEV.

- On-board charger: It serves as a supplier for the Li-ion batteries, to achieve the required voltage through an AC-DC converter with PFC.[2] It is used to control the voltage and current parameters and therefore maintain stable values[11]. This part will be discussed later in the electric circuit chapter.
- Battery management system: A robust BMS is needed for the BESS, this is to observe and sustain optimal and reliable operation parameters of both the battery pack and SSC to make the whole system monitored.[12] batteries chemistry is dynamic and in most cases, they operate outside the equilibrium cycles and BMS helps in maximizing the efficiency and safety of overcharging and over-discharging. [2]
- DC/DC converter: Mainly to convert DC voltage to DC voltage. This is because different component in the circuit has different operating voltage which needs a component transfer to the required voltage. [13]
- Main inverter: Responsible for converting the high voltage DC/AC in the electric circuit and then from that to the electric motor. [2]

2.3 Battery Circuit

One of the main components in BEV is the battery circuit. During the past years, BEV's development has increased dramatically due to the new regulations enacted to reduce emissions and make car manufacturers aware of climate change. The range of today's vehicles can vary depending on the atmospheric condition, driver style, road condition, battery SOC, and much more. [14]

A battery circuit is used to cool down or heat the car's temperature, deliver power to the electric motor, and store the vehicle's energy. The battery circuit is considered a low-temperature circuit compared to the other circuits. The temperature of the coolant out can approximately reach up to 45 deg during regular operating conditions.

2.3.1 High voltage battery

It maintains the battery cells' safe and reliable operating temperature. Moreover, it protects the battery from aging too fast because overcharging or over-discharging the cells might lead to aging. Those two phenomena occur when the battery is charged or discharged over or below the operating voltage limit specified by the electrode chemistry coupling.[15] Therefore, the battery management system (BMS) came to avoid any internal short caused by overcharging or over-discharging. This allows the cells to operate with their normal operating voltage limit and monitor the cells to spot malfunctioning cells within a battery pack before any fatal error.[16] Moreover, the BMS can make sure that the cells are charged and discharged equally to guarantee and avoid any faulty problems, knowing that the cell's capacity varies from one to another by charging and discharging properties [17]
Those are some of the components that the high voltage battery consists of: Cooling shut-off valve, heat exchanger included Expansion tank level sensor Temperature sensor, External temperature sensor, Coolant fluid pump, Bypass valve, and Cooling fan.

2.3.2 Converting high voltage to low voltage

High voltage transformers convert voltages from one level or phase configuration to another; in BEV, usually from higher to lower voltage. In addition, this component is considered sensitive, and this is due to its small size, which gives it a very small thermal mass. This is why it is important to monitor this component because it tends to heat up very fast, affecting the vehicle's functionality. This part is equipped with an electrical isolation, power distribution, control, and instrumentation applications. Those converters depend on laws of magnetic induction between coils to convert high voltage to lower levels.

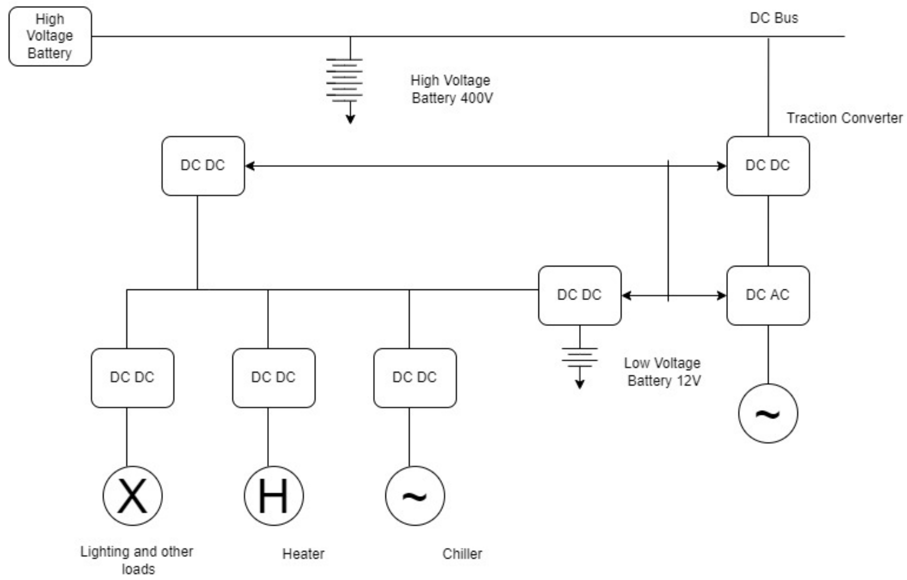


Figure 2.3: Converters to transform the high voltage to low voltage [3]

2.3.3 Chiller

The main purpose of the chiller is to cool the coolant in the cooling circuits and avoid heating the critical component. Moreover, a reliable and well-functioning chiller allows operating the component in the circuit in optimum performance. This is done by a vapor compression refrigeration cycle. This cycle is used to excess heat from a low-temperature area to the low temperature of the cycle by adding work to the system and exhaust heat at a high temperature. Four steps need to be done to get a robust refrigeration cycle. [4] [18]

1. vapor-liquid mixture, the evaporator is transferring heat from low temperature to the evaporator, which is at an even lower temperature. Then the heat is transferred, and refrigerant is evaporated at this point. Having a saturated vapor at point 2.
2. The saturated vapor is getting work from the compressor to raise the pressure from low to higher. Assuming an adiabatic process to increase the pressure of a gas, we have also to increase the temperature because we are putting work into the gas
3. The high temperature and high-pressure vapor are then condensed to a liquid at constant pressure. Remove heat from here to condense to a liquid at high pressure.
4. This high-pressure liquid then expands through a throttle, an expansion valve. Assume it is adiabatic because it is rapid, and no work is being done. Nevertheless, lowering the pressure of a saturated liquid. The liquid evaporates because it is adiabatic. The energy that evaporates comes from cooling everything down. This is how we get the low temperature needed to complete our refrigeration. [19] [20]

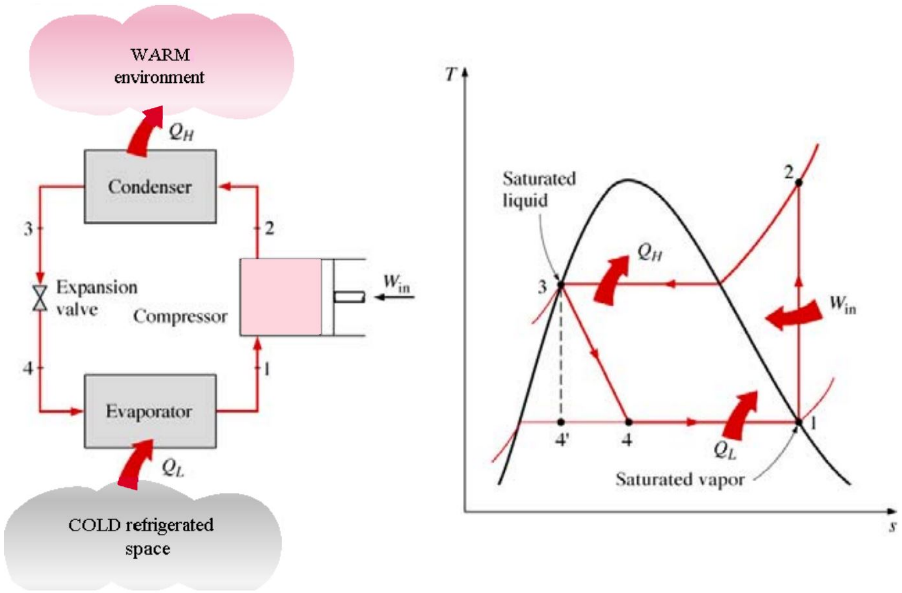


Figure 2.4: Left: cycle of vapor-compression refrigeration; right: refrigeration scheme [4]

2.4 Electric Drive Circuit

The second circuit that comes in the BEV layout is the electric drive circuit, which is responsible for transferring power from the battery pack to the drive motor. The mechanical power is transferred to electrical power by sending a signal from the accelerator pedal to the controller, which alters the vehicle speed by varying the frequency of the AC power from the inverter to the motor, connecting and turning the wheels through a gear. When the driver uses the brakes and the car's speed goes down, the motor act accordingly. Then an alternator produces power which is sent back to the battery powering an electric vehicle. [21] In this circuit, the temperature is medium and can reach up to 70C at the outlet of each component

2.4.1 VCU

VCU's mission is to be the control center. The purpose behind using it is to analyze the data obtained by transducers and send commands to executors or actuators to do their function.[22]It ensures that the components are talking to each other and coordinating for better performance. For example, if the driver turns on the vehicle and presses the accelerator to peddle, this will be the electric signal sent to the VCU. The VCU will decode this signal and tell the motor to run at a specific RPM. The RPM of the motor will be determined from the percentage of the throttle given.

2.4.2 Front and Rear Inventor

The inverter's function is to change the DC on the battery into an AC, and then this alternating current is used by an electric motor. In addition, the inverter on an electric car also has a function to change the AC when regenerative braking to DC and then use it to recharge the battery. Some electric cars use different inverter models, such as the bi-directional inverter category.

2.4.3 OBC

A board charger allows the driver to charge their battery anywhere there is AC power, not just a charging station. It works by taking a wall output and an AC output, running it through the PFC, which gives you a DC output then you run that into the LLC board, which converts it to a voltage to charge the car. OBC can run in multiple modes, one of which is voltage control mode and another is current control mode. In voltage control mode, you set the voltage to whatever output you want to charge your battery. In control mode, you set your current to whatever level you want to power the OBC. [23] [24]

2.5 Climate Circuit

The last circuit to discuss is the climate circuit. This circuit contains components such as water condenser WCOND, HVCH, and HVAC; all three will be discussed later on, and how they are connected. Here the temperature of the coolant is the highest among all circuits and can reach up to 90C at the outlet.

2.5.1 Water condenser

It is used to absorb the excess heat from the chiller and condense it, allowing it to send this heat to the cabin. The condenser is a heat pump alternative. The principle in heating mode is to absorb heat from the colder area, such as the ambient air at 0C, then use this heat to heat the cabin. In heating mode, the coolant exit the compressor in high-pressure, high-temperature vapor and passes to the reversing valve. which allow the coolant to go to the cabin, it is used to remove heat from the cabin.

The refrigerant boils at very low values. Therefore, when it boils, it is accompanied by a thermal potential. For example, taking a known example, after boiling water, the thermal potential of its steam is carried away. The refrigerant boils at a very low temperature, which will carry away the thermal energy as it boils. So the refrigerant picks up the thermal energy from the outside air and leaves the outdoor heat exchanger as a low-pressure, low-temperature, slightly superheated vapor, then heads back to the reversing valve. The reversing valve forward and change the process to the compressor to repeat the cycle. [4] [18] [25] So the advantage of using a heat pump is to utilize the heat available in the ambient air to heat the cabin, reflecting less energy from the battery.

2.5.2 Heat Exchanger

The heat exchanger is a system used to transfer heat between two or more fluids, liquid or gas. The heat is transferred by conduction through the exchange of fluids separated by a conductive medium to avoid the risk of short-circuiting the airstreams. Cross flow heat exchanger interchanges thermal potential from a 2-sided airstream in an air handling unit. The type of heat exchanger used the most is the cross-flow heat exchanger, where fluids flow perpendicularly. This component is made from narrow aluminum metal sheets. The thermal potential is transferred between sheets. Those kinds of heat exchangers have a good thermal efficiency that varies between 40 % and 60% and are made of square cross-sections. Some problems that may come up with this component are humid ambient that might lead to freezing and forming ice. [26]

2.6 Silver Library

Some steps need to be taken in advance to export the harness model from Simulink as a Silver module. First, the model in Simulink must be defined as Simulink Import and Simulink Outport at a high level exactly like in the figure 2.5.

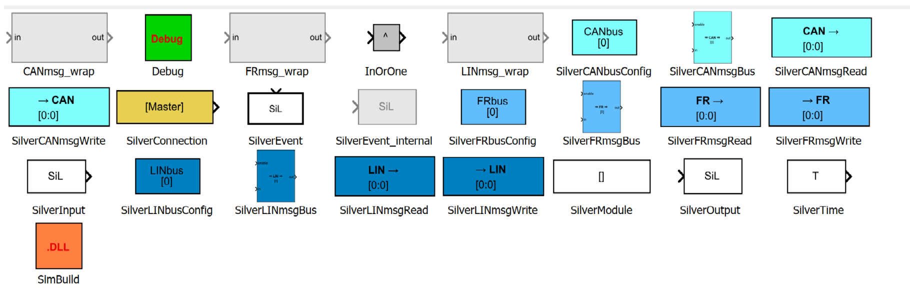


Figure 2.5: Silver Library

Second, the model needs to have SilverInput and SilverOutput blocks. This can be imported from the Silver library installed manually from the file settings that come with Silver installation; check figure 2.6. This does not necessarily need to be at the top level, rather than any level of the harness.

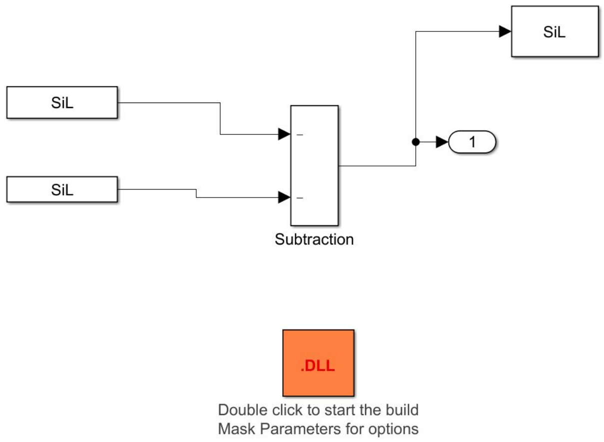


Figure 2.6: Input and Output blocks

Moreover, some windows and inputs will be counted as part of the environment interface to be inspected and controlled in the software. For instance, regarding the parameters of each module, if the InlineParams parameter is off, then the adjustable parameters in the blocks model will be exported. On the other hand, if the InlineParams parameter is on, then the adjustable parameters configured as global shall be exported.

To start the importing process to Silver from Simulink, there must be a Simulink coder to handle the script transferring to Silver. One of the compatible compilers used in this process is the basic C++ compiler, Visual Studio 13 community edition. However, it should work fine with noncommunity editions as well. Like any Simulink Coder export, the Silver export requires selecting a 'fixed-step' solver and an appropriate step size. Note that the step size has to be smaller than Silver's macro-step size; otherwise, Silver will refuse to run the generated DLL.

Finally, to check if the Simbuild block is working properly, build a dummy project and try executing in an individual path. All the instructions should be shown in the Matlab command window. If everything goes as planned, a message should appear: "Building Dll was successful."

2.7 WLTP

Worldwide Harmonised Light Vehicle Test Procedure is a test conducted by automobile manufacturers to calculate fuel consumption, CO2 emissions, and adjutant emissions. This test was established in 2017, replacing the old test version of NEDC.[27] The new test was implemented after the increase in technological evaluation, which led to the measurements being outdated and lost authenticity; itthe EU law and governments set it.[28] The difference between the two tests is that the NEDC vehicle consumption values are calculated on a theoretical basis. On the other hand, the WLTP takes measurements based on real-life driving data, which is categorized into four different speed intervals with various accelerations and braking stages.[29]

The WLTP will give customers a closer and more realistic look at their vehicle condition and performance than the NEDC. Nevertheless, this does not mean all the possible variations will be included in the test. To illustrate, this test is done individually, meaning each driver has their driving style. Some might accelerate faster, take a sharper corner, or use the brake more frequently. Moreover, considering the traffic and weather conditions vary from country to country, making it is harder to have a specific standard for those parameters. This will result from having different emissions measured from the lab compared to real-life scenarios. However, as there is no single real-world emission value, only values obtained by standardized laboratory tests allow us to directly compare the emissions and fuel consumption of different car models from different car manufacturers.[30]

3
Methods

3.1 Blocks Extraction

This was the building block and the first input of the article. The purpose This step was to extract the plant models from the complete vehicle platform environment and integrate them with the harness in Simulink. The complete vehicle platform is a cross-functional platform that Volvo Cars use to virtually develop all the well-functioning Simulink plant model, including the controller and thermal side. It Gives the engineers the ability to simulate, test, analyze, and verify the vehicles on a high-level evaluation of the system's behavior and integration. It is a complete vehicle simulation setup. It includes vehicle components such as high voltage batteries and cooling system control. So everything in the complete vehicle platform is like a backend for the GUI. To have accurate SiL testing, it was necessary to extract all the circuits, including the battery, electric drive, and climate circuit.

The harness has signals acting as a dummy; therefore, a functional plant model with the required signal was needed to start the simulation process. For instance, the reason to use a dummy pressure drop is that we need to have the pressure drop in the component to guess the mass flow. This model was imported from the complete vehicle platform environment with the attached mat file containing many parameters and variables. To start the extraction, input, and outputs needed to be specified to check the functionality of the models. Such signals in the battery circuit are: new temperature out, pressure drop out, and mass flows out.

Before extraction, each Simulink plant model plant is connected to file dependencies that make it function as requested. Those files contain the setting of the plant model, such as the valve opening position or the ambient temperature. Those files are selected and imported to the same path as the plant model. Inputs are either written in the variables mat file or added as signals in the plant. The pars file, which includes all the needed parameters and constant to simulate the plant, also is modified to add or remove the required or excess variables to reduce the running time of the simulation. The final step is to integrate all the files, signals, mat files, and parameters into the suitable space, then incorporate them with the Simulink model.

3.2 DLL Transformation

DLL files are shared libraries on Silver to identify plant inputs, outputs, and valuable information such as function and data. Applications or other shared libraries can use those supporting files during the run time. The reason behind using all files is due to many factors. For example, the application file "The plant model" is more compact and takes less space and CPU, Where the mat file and data can be stored in a separate file. Moreover, the DLL file is disconnected from the running application, which means that the DLL file will run only when the application loads, therefore saving RAM power. Lastly, the Dll file allows the ability to use and edit the code optimally. For instance, the parameter file we have is used in many plant models; if an error occurs in this file, it can be easily adjusted and then directly compiled into the other models. Instead of needing to edit raw code in all models and recompile all of those models, which saves a lot of time.

To have a complete SiL setup, each plant model must have Dll and a2l files. Similarly, Dll and a2l file for the production software to control the output and inputs of the plant. Below the process of transforming each of them is illustrated:

3.2.1 Plant Model

After the complete integration with the harness and extraction of the model, the plant is ready to be transformed into a DLL file to become readable by Silver. For this step, it was necessary to specify what is considered input of the plant and output of the plant in both the production software and the harness. Using the Silver library mentioned earlier, signals such as fan speed and pump speed have been specified as SilverOutput block in Simulink, which at the same time input for the SPM. SilverInput block took signals such as actuators; this is at the same time as the output of the SPM. After that, the SimBuild block is imported into the upper level of the harness, which can transfer the whole plant to dll format. During the transformation, many files will be stored in the same Matlab path where the work is being done. For instance, the a2l file is responsible for reading the parameters and values.

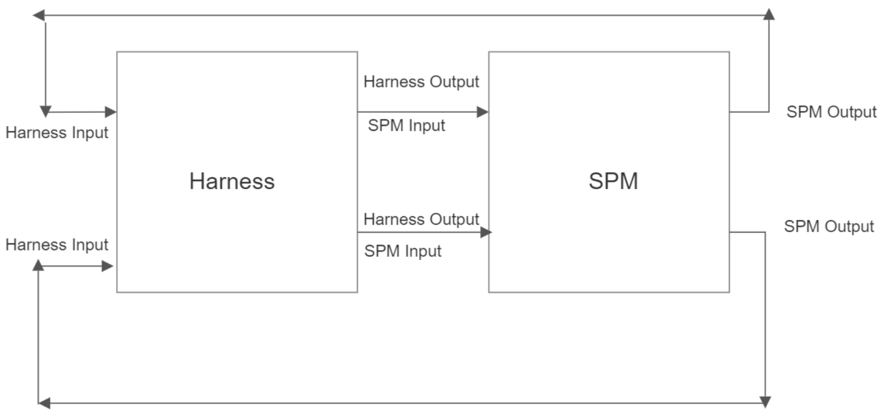


Figure 3.1: Process illustration

3.2.2 Production Software

The production software is the thermal control software that controls the thermal management system. It is responsible for delivering efficient cooling or heating to the thermal component, like controlling the settings of the coolant pump flow rate of the high voltage battery circuit. The SPM always tries to find the best estimation for those parameters to enhance the performance of the thermal system. Likewise, this software is written in Dll format and installed on the vehicle's hardware. Therefore, this code also needs to be generated in Dll to Silver to recognize it and make it communicate with the harness Dll.
A different approach was used to generate the SPM Dll using an in-house tool called TestWeaver GUI, which allows easier continuous integration and validation of the results. First, the SPM contains all different electrical sensors and actuators. Some act as receivers and transmitters from and toward the SPM. Moreover, in this GUI, there is all the thermal management software model; what is essential here is the thermal control models. Since the whole harness is included in the dll, all the updated models must be transferred to Dll format. After adding them to a specific path and installing the necessary reader, such as visual studio, the GUI starts the process of transforming. Next, after finishing in the path, a py file is found containing the code and a subinterface having the Inputs and Outputs of every model. In each of the models, there is a Simulink file containing the actuators and signal that it controls; also, there is a config file that contains all the information needed about the signal with JSON extension, such as the name of the signal, configuration, description, the maximum, and minimum allowed range for it, type of the value like float32.

3.3 Integration with Silver

Now all files, settings, and configurations are ready to be compiled on Silver. The Dll file gives the ability for the plant and SPM to communicate between each others. This is needed for the parameter interaction to give accurate testing results. Both Dll needs to be set and uploaded in the same SiL file containing the a2l file. Before starting the simulation, there must be a checklist.

First, choose New Module to select the folder sign to make a new environment, then choose the Dll model. Here, the process will be done twice since there are two models, the plant harness containing the function and the SPM to control those functions. Then the file type needs to be chosen from the Select Module pop-out window. In the variable tab, the list of all the parameters is available from both the SPM and Harness. However, for Silver to recognize it, the only parameters available are those specified as SilverInput or SilverOut in the file. In the console tab, the error that doesn't let the simulation start will be present here; in addition, important information about the environment in general. Also, a module's warning output but not any module's input.
On the left side of the page is the Project tree for the currently loaded module configuration. Provides access to all modules, widgets, and current configuration

variables. Use the following toggle buttons to configure the tree display.Show or hide the modules' widgets of the GUI module show variables as a flat list or as a tree. The tree option uses the tree variable separator characters.

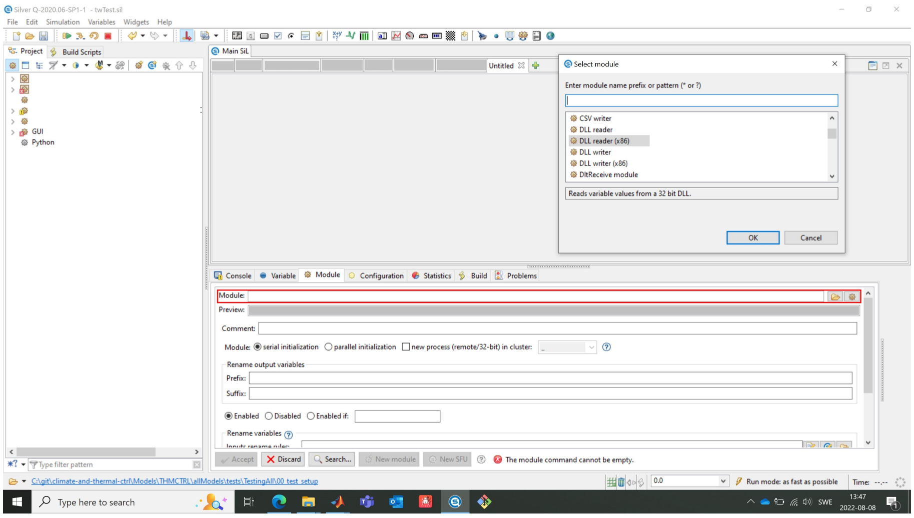


Figure 3.2: Silver layout

The modules and variables in the project tree are decorated as follows: If modules are shown in the project tree, the variables carry additional decorations to indicate the the direction of signal flow. To get started with understanding the Silver environment. Blue circle with right arrow indicates an output of the module. Blue circle with left arrow indicates an input of the module. Blue circle with right and left arrow indicates input and output of the module, i.e., output at time t + macro step width, depends on the input at time t. Note that the output value of the module at one macro step equals the input read by the the module at the next macro step only if no other module overrides that output. Blue circle with letter M indicates that the module modifies the variable. Modified variables can also be inputs and outputs of the module

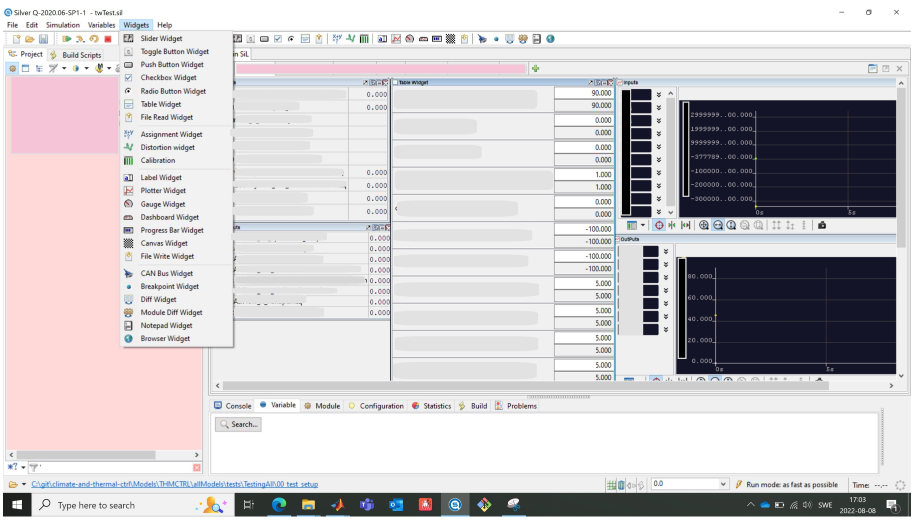


Figure 3.3: Layout Setup

All models and different inputs were added to the workspace to be monitored. For example, vehicle speed, electric motor angular speed, torque, and HV battery current are inputs. The monitoring is done through different widgets, as shown in the figure. There are different ways to show your results and control your input, like a slider widget, plotter, label, and more. The last step before a simulation is to check the Console; if no error is present and written in the workspace, then the simulation can start smoothly from the Simulation task bar. The results will be monitored through the tabs of each model.

3.4 WLTP Cycles

Different WLTP cycles were integrated into the harness to predict different scenarios the thermal management system can face. This includes a large number of readings and data at a specific time. Using a multi-port switch in Simulink

4

Results

The results will be illustrated, shown, and reflected upon in this chapter of the article. The plots and graphs demonstrated are compiled and generated through Silver. In addition, different settings and configurations of the WLTP cycles were tested to see how the vehicle parameters react in various behaviors.

4.1 Harness Input Data

Six main inputs were inserted in each simulation to monitor how adjusting them can affect the thermal system management. Those inputs are front motor speed, rear motor speed, front motor torque, rear motor torque, ambient temperature, and vehicle speed. Depending on the drive cycle selected from the user-specific inputs,, the data-sheet is imported,, including the simulation run time and the values of each input. The figures below show all mentioned inputs when activating the WLTC_35 drive cycle. Then if the user is interested in a different type of environment, what is needed is to rest the simulation and then toggle the slider to the desired drive cycle; all inputs will change accordingly. Those inputs are fed into the harness to activate the needed function to have a complete cooling or heating system.

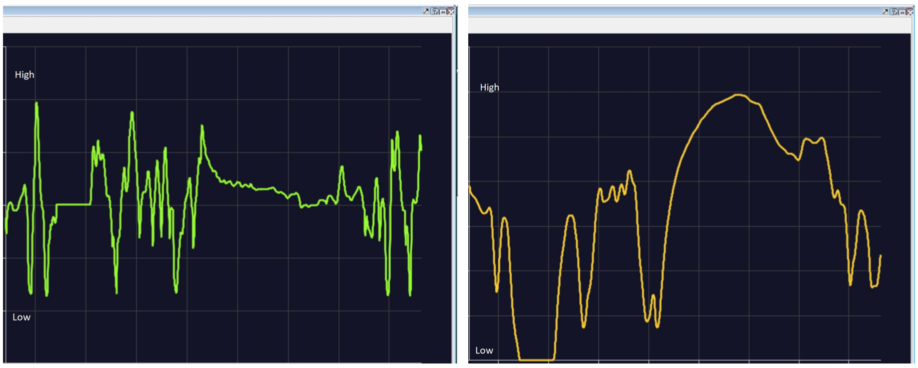


Figure 4.1: Front

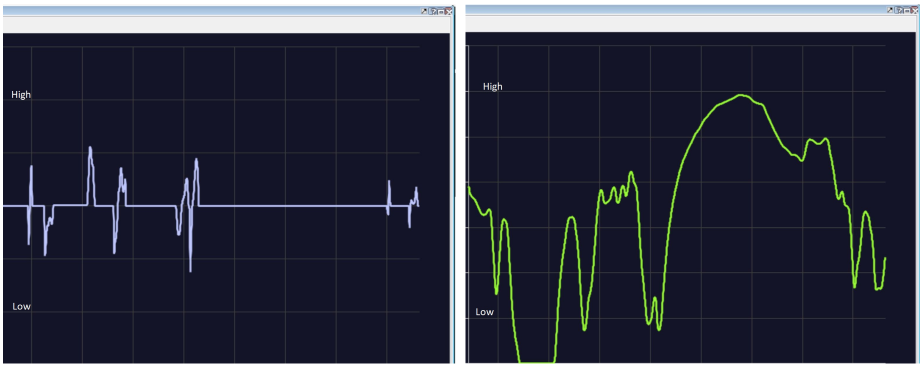


Figure 4.2: Rear

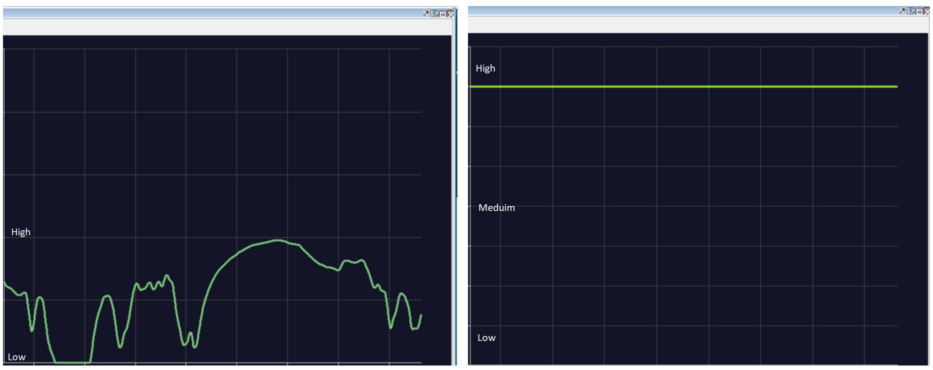


Figure 4.3: Speed and Temperature

4.2 Input

The input represents what the harness model is sending to the SPM. For instance, the coolant flow estimation model sends specific values to the SPM to allow the model to function in an optimum way. One of the inputs in this model, sent to the SPM, is the coolant motor request in and out of the electric drive circuit. The figure 4.4 shows how the layout is set in the Silver environment.

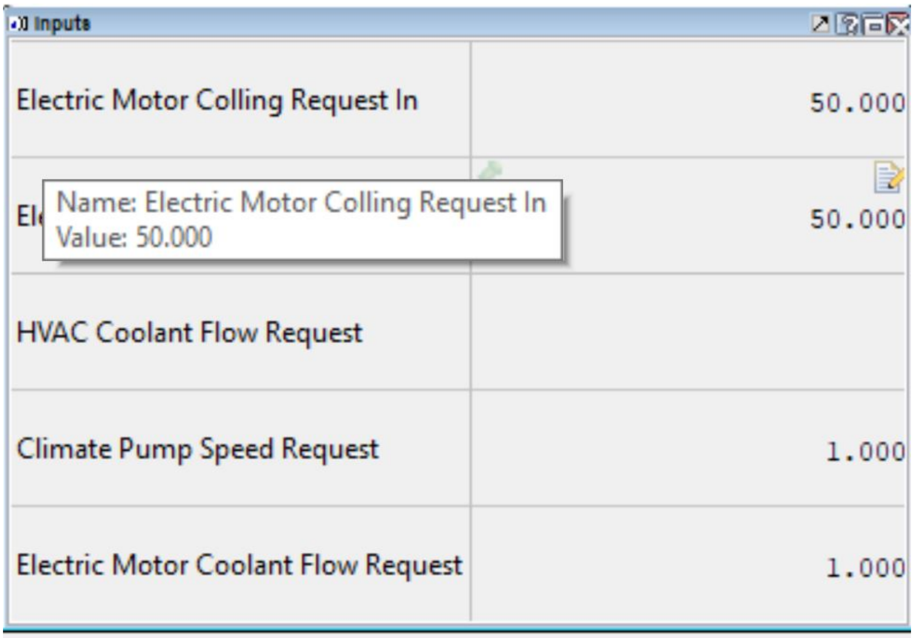


Figure 4.4: Inputs

4.3 Outputs

The output represents what the SPM model is sending to the harness. Taking the thermal system control, the SPM Dll calculates and controls the model to request a specific value to enhance the thermal management system. Like sending the signal to the coolant radiator valve to open or close as one or zero. Then this value will be forwarded to the respective signal in the control model to bypass or omit the coolant through this valve. Moreover, the mode of the battery circuit, electrical drive circuit, and climate circuit are also given through the output window in Silver. It sends an activation signal to the respective circuit to trigger active or passive cooling. Thermal modes respond as requested.

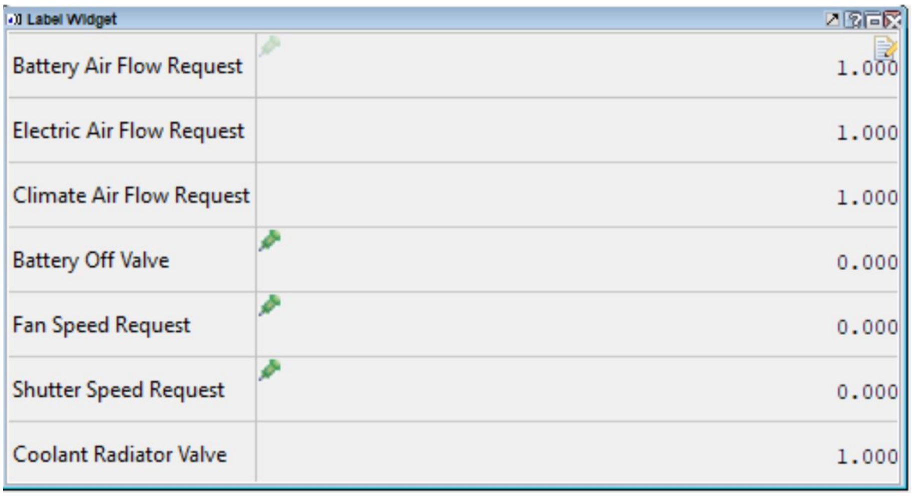


Figure 4.5: Outputs

4.4 Parameters

All parameters from the SPM Dll file can be adjusted and calibrated through Silver. Silver provides this feature to change the targeted parameters and monitor how the outputs react. During the simulation, the plots of the output will be visible on the workspace; This allows the user to assist if the results are close to the real-time testing. If not, the parameter in the model's tab shall be adjusted accordingly.

4.5 Models

Each of the thermal models, such as cooling system control, consists of all the Inputs, Outputs, and parameters needed to complete the functionality of each system. Having this layout in each model with its own window allows for easier access to the data corresponding to that model.

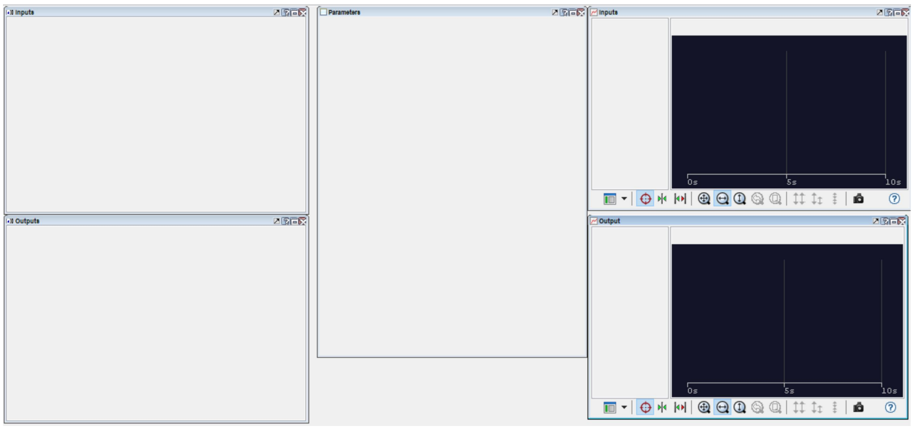


Figure 4.6: Models Layout

5 Conclusion

The demand for the battery electric vehicle is increasing daily, where the market is expanding, and the competition is growing. Therefore, the need for parallel validation and verification testing throughout the development process has gained attention from academia and industry. Plenty of studies have pointed out that early testing is beneficial in identifying early bugs and errors during the process implementation. This master thesis shows that SiL techniques are advantageous and applicable in the automotive industry.

The main goal of this thesis was to build a virtual environment in Silver to connect models from both Simulink and the production software. This allowed the models to interact and communicate with each other, in addition to sharing data coming in and out of the environment.

The study was performed on three different circuits battery, electric drive, and climate circuit. The SiL environment successfully shared data between the harness and SPM. This includes values such as battery air flow and fan speed requests. Moreover, Silver allows communication between models to give the battery circuit different cooling and heating modes.

The accuracy of the environment is still to be investigated. However, this kind of testing will efficiently decrease the time to find bugs and errors in the system, similar to the actual time testing. In addition, it reduces the simulation time and gives a higher efficient optimization procedure during both physical and testing.

Bibliography

[1] Energy-efficiency-statistic. https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Energy_efficiency_statistics#Primary_energy_consumption_and_distance_to_2020_and_2030_targets. (Accessed: 2022-06-17).

[2] Joe Bush. Future-proofing ev battery management systems.;. https://www.electronicspecifier.com/industries/automotive/future-proofing-ev-battery-management-systems. (Accessed: 2022-07-31).

[3] Salvador Litrán, E. Duran, Jorge Semião, and Cristian Díaz Martín. Multiple-output dc–dc converters: Applications and solutions. *Electronics*, 11:1258, 04 2022. doi: 10.3390/electronics11081258. URL https://www.researchgate.net/publication/360003737_Multiple-Output_DC-DC_Converters_Applications_and_Solutions.

[4] Y. A. Cengel and M. A. Boles. *Thermodynamics: An Engineering Approach*. Mcgraw-Hill College, (2008).

[5] Co2 emissions from cars. https://www.europarl.europa.eu/news/en/headlines/society/20190313STO31218/co2-emissions-from-cars-facts-and-figures-infographics. (Accessed: 2022-06-17).

[6] Reducing carbon emissions. https://www.europarl.europa.eu/news/en/headlines/priorities/climate-change/20180305STO99003/reducing-carbon-emissions-eu-targets-and-measures. (Accessed: 2022-06-17).

[7] Ets. https://www.europarl.europa.eu/news/en/headlines/society/20170213STO62208/the-eu-emissions-trading-scheme-ets-and-its-reform-in-brief. (Accessed: 2022-06-17).

[8] Efficiency of bev. https://www.renaultgroup.com/en/news-on-air/news/the-energy-efficiency-of-an-electric-car-motor/#:~:text=For. (Accessed: 2022-06-17).

[9]

[10] MOBILITY INSIDER. What is software-in-the-loop testing? https://www.aptiv.com/en/insights/article/what-is-software-in-the-loop-testing#:~:text=Software. (Accessed: 2022-08-15).

[11] M. Grenier, M. G. Hosseini Aghdam, and T. Thiringer. Design of on-board charger for plug-in hybrid electric vehicle. In *5th IET International Conference on Power Electronics, Machines and Drives (PEMD 2010)*, pages 1–6, 2010. doi: 10.1049/cp.2010.0101.

[12] Matthew T. Lawder, Bharatkumar Suthar, Paul W. C. Northrop, Sumitava De, C. Michael Hoff, Olivia Leitermann, Mariesa L. Crow, Shriram Santhanagopalan, and Venkat R. Subramanian. Battery energy storage system (bess) and battery management system (bms) for grid-scale applications. *Proceedings of the IEEE*, 102(6):1014–1030, 2014. doi: 10.1109/JPROC.2014.2317451.

[13] rohm. What is a dc-dc converter? https://www.rohm.com/electronics-basics/dc-dc-converters/what-is-dc-dc-converter. (Accessed: 2022-08-1).

[14] Emma Arfa Grunditz and Torbjörn Thiringer. Performance analysis of current bevs based on a comprehensive review of specifications. *IEEE Transactions on Transportation Electrification*, 2(3):270–289, 2016.

[15] Arumugam Manthiram. An outlook on lithium ion battery technology. *ACS central science*, 3(10):1063–1069, 2017.

[16] Languang Lu, Xuebing Han, Jianqiu Li, Jianfeng Hua, and Minggao Ouyang. A review on the key issues for lithium-ion battery management in electric vehicles. *Journal of power sources*, 226:272–288, 2013.

[17] Daniel Juarez-Robles, Anjul Arun Vyas, Conner Fear, Judith A Jeevarajan, and Partha P Mukherjee. Overdischarge and aging analytics of li-ion cells. *Journal of the Electrochemical Society*, 167(9):090558, 2020.

[18] J. R. Howell and R. O. Buckius. *Fundamentals of Engineering Thermodynamics.* Mcgraw-Hill College, (2008).

[19] Central air conditioner and refrigeration.;. http://www.central-air-conditioner-andrefrigeration.com/Refrigeration_Cycle.html. (Accessed: 2022-07-30).

[20] Berg chilling systems inc.;. https://berg-group.com/engineeredsolutions/the-science-behind-refrigeration/. (Accessed: 2022-07-30).

[21] Mark Duvall et al. Advanced batteries for electric-drive vehicles. *Electric Power Research Institute (EPRI)*, 2004.

[22] Zhu Yunzheng, Qu Shaofei, Allen C. Huang, and Xu Jianhong. Smart vehicle control unit - an integrated bms and vcu system. *IFAC-PapersOnLine*, 51(31):676–679, 2018. ISSN 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2018.10.156. URL https://www.sciencedirect.com/science/article/pii/S2405896318326223. 5th IFAC Conference on Engine and Powertrain Control, Simulation and Modeling E-COSM 2018.

[23] Kenichi Itoh, Masanori Ishigaki, Naoto Kikuchi, Tomohisa Harada, and Takahide Sugiyama. A single-stage rectifier with interleaved totem-pole pfc and dual active bridge (dab) converter for phev/bev on-board charger. pages 1936–1941, 2020. doi: 10.1109/APEC39645.2020.9124083. URL https://ieeexplore.ieee.org/document/9124083.

[24] Aziz Rachid, Hassan El Fadil, Fouad Giri, and Abdellah Lassioui. Nonlinear output feedback control of v2g single-phase on-board bev charger. *Asian Journal of Control*, 22(5):1848–1859, 2020. doi: https://doi.org/10.1002/asjc.2082. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/asjc.2082.

[25] Paul Evans. The engineering mindset.;. https://theengineeringmindset.com/how-do-refrigerants-work. (Accessed: 2022-08-3).

[26] Grundfos. https://www.grundfos.com/us/learn/research-and-insights/cross-flow-heat-exchanger. (Accessed: 2022-08-3).

[27] WLTP Facts. How does it work. https://www.wltpfacts.eu/what-is-wltp-how-will-it-work/. (Accessed: 2022-08-16).

[28] eur-lex Europa. Summaries of eu legislation. https://eur-lex.europa.eu/EN/legal-content/summary/worldwide-harmonised-light-duty-vehicles-test-procedure-wltp-and-real-driving-html. (Accessed: 2022-08-16).

[29] Nathan Dale. What is wltp and what does it mean? wltp explained. https://www.evanshalshaw.com/blog/what-is-wltp/. (Accessed: 2022-08-16).

[30] WLTP facts. Will wltp end the discrepancy between the laboratory and on-road performance of cars. https://www.wltpfacts.eu/wltp-discrepancy-laboratory-road-performance-cars/. (Accessed: 2022-08-16).

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se