Design and Documentation of the Comfort Control Unit in Systems Modeling Language(SysML)

Kevin Davis, Felix Diekenbrock, Enea Robaj, Naved Afridi Rashiq

Hochschule Hamm-Lippstadt, Lippstadt, Germany

Interactive Systems Engineering 2, BSc Electronic Engineering

{kevin.davis, felix.diekenbrock, enea.robaj, naved-afridi.rashiq}@stud.hshl.de

Abstract

This technical report documents the design of applying the Systems Modeling Language to a preexisting system. The system in particular is the Comfort Control Unit or CCU in an automobile. The CCU's purpose is to enhance the user's experience in terms of heightened convenience, comfort, and safety of the passenger/driver. Using SysML, aspects of the CCU's Requirements, Structure, Behavior and Parametric are represented in terms of diagrams. Diagram types used to represent the previously mentioned aspects include: Requirements(req), Use Case(uc), Block Definition Diagram(bdd), Internal Block Diagram (ibd), Activity Diagram(act), State Machine (stm) and Sequence Diagrams(sq) respectively. Through these diagrams, the CCU's system model is expressed as a model repository that shows how each component and its associated functions integrate in order to satisfy its requirements and meet the overall objective.

Table of Contents

1 Introduction

As society becomes more advanced, so does the inter-connectivity of the systems making ours lives more convenient and efficient. More recently, driven by efficiency and market goals, manufacturing companies look to improve the methods used to design and manufacture products and systems. One particular area of difficulty during the product life cycle, is the conceptual stage, where shareholders, designers and manufacturers decide on the architecture of the system.

This initial stage is where the customer's needs are transformed into functions and use cases within the product. Without a systematic approach to this design process, it will lead to difficulty in tracing whether the overall objective of the product is met. As well, a lack of a High-level view of the project can also cause to inadequate decisions being made, leading to possible individual unit and integration issues. It is these issues confronted during the conceptual phases of the project, that the Systems Modeling Language(SysML) intends to alleviate.

The goal of our project is to model the Comfort Control Unit (CCU) of an automobile using the Systems Modeling Language (SysML). Through SysML, we can effectively communicate the intended functions of the system , how the user interacts with the system, the composition and the intended behavior of the system depending on external stimulus/input.

2 Requirements and Specifications

The Comfort Control Unit in an automobile is used to manage various comfort-related systems. Moreover, it also manages certain safety systems that also contribute to the driver/passenger's comfort. Through these systems, the CCU aims to enhance the user's overall driving experience. Composed of both hardware and software components, the system should ensure proper operation at all times. Key criteria is to include high energy efficiency and smooth real-time performance to deliver a reliable driving experience. The systems components include:

2.1 Hardware Components:

- **Microcontroller Unit (MCU):** Central controller for all relevant Units and components (e.g., ARM Cortex-M, AVR).
- **Actuators:** For Windows, Side Mirror, Seat and Locking control(Door and Trunk).
- **Lights:** Includes Interior, Headlights, Fog lights, Brake lights and Tail lights.
- **Sensors:** Temperature, Light, Rain and Distance Sensors.
- **User Input Interfaces:**: Button, Lever and Dial.
- **Specialized Units:** Air Conditioning, Airflow Control.
- **Miscellaneous:** A/D Converters, Locks and Antenna.

2.2 Software Architecture:

- **Embedded C/C++ Firmware:** Ensures real-time control of the CCU.
- **Diagnostic/Logging Modules:** For fault detection and Maintenance.
- **Communication Protocol:** Controller Area Network Bus(CAN).

With key Hardware Components and Software Architecture identified, attention must be turned to critical design considerations:

2.3 Design Considerations:

- **Automotive Compliance:** The system must meet the ISO 26262 standard criteria for safety.
- **Temperature Range:** The system should operate between -40°C to +85°C in normal operations.
- **Design Constraints:** The system's physical design should be compact as space in a vehicle cabin/body is limited.
- **Fail-Safe Mechanism:** The system must ensure basic functionality even during faults (ie. Power Failure).

3 SysML Diagrams and Analysis

3.1 Requirements

To model and track the requirements of the system. We model the functions of the CCU using the Requirements diagram. The CCU contains two systems that contribute to the user's driving experience, of which are **Safety** and **Comfort**.

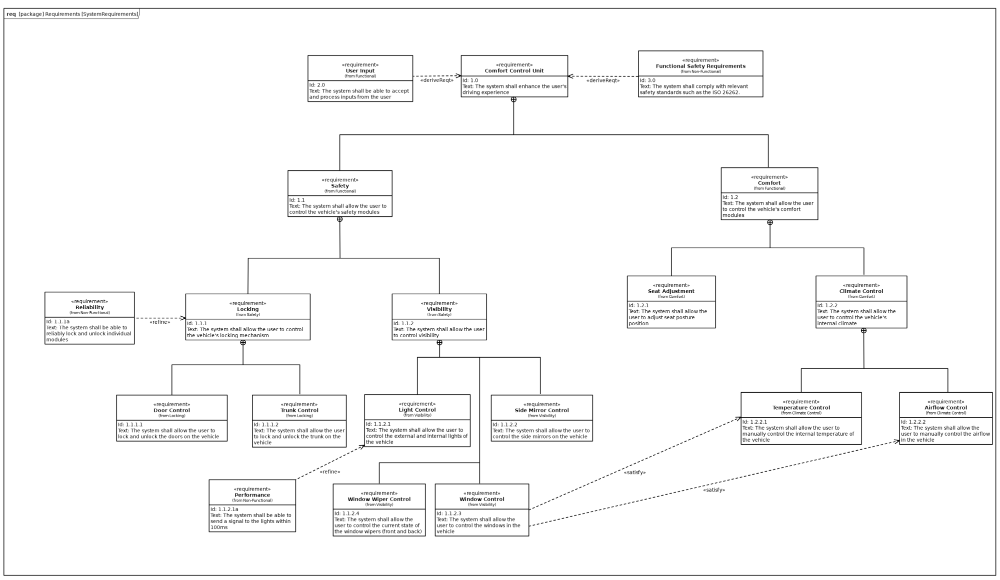


Figure 1: CCU System Requirements Diagram

The **Safety** requirement contains the **Locking Control** and **Visibility Control**. **Locking** contains **Door Control** and **Trunk Control** as these are the parts of the vehicle that the user will want to inhibit access to in order to prevent theft. A non-functional requirement of the Locking requirement is **Reliability**, thereby the system should consistently perform its function. This is a ≪**refine**≫ relationship as it adds detail to a requirement. Moreover, the **Visibility** requirement contains **Light**, **Side Mirror**, **Window** and **Window Wiper Control**. Similar to **Reliability**, the Non-Functional requirement of **Performance** also has a ≪**refine**≫ relationship with **Light Control** by adding detail that the system needs to send a signal to the lights within 100ms. Additionally, the requirement of **Comfort** contains **Seat Adjustment** and **Climate Control**, Seat adjustment was not broken down further as it would not add refinement and would just be a simple actuation-system in comparison to **Climate Control**. Contained in **Climate Control** is **Temperature Control** and **Airflow Control**. **Window Control** has a ≪**satisfy**≫ relationship with the latter as opening a window also contributes to affecting the internal climate within a vehicle. Derived Requirements are **User**

Input and **Functional Safety Requirements**, these are important as the user needs to be able to interact with the system and also the system should comply with relevant safety standards such as the ISO 26262.

3.2 Use Case

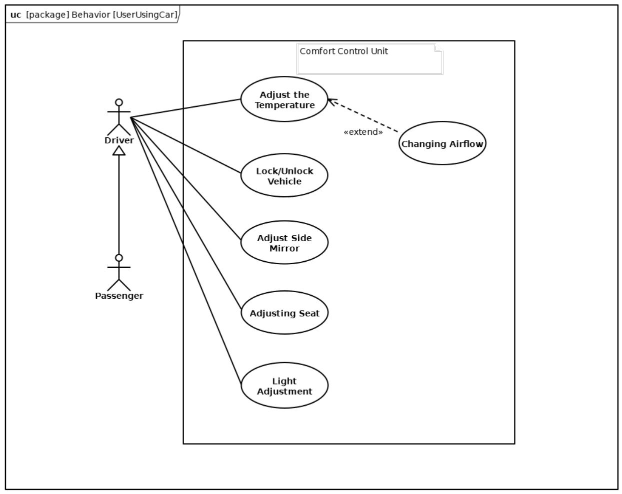


Figure 2: CCU System Use Case Diagram

Figure 7 presents a use case diagram that illustrates the functionality of the Comfort Control Unit of a car, emphasizing the interactions between the system's users, namely the Driver and Passenger, and various comfort-related features. The use case diagram focuses on how the Driver and Passenger interact with this unit to perform different comfort-related actions. The involved Actors include:

- **Driver**: The primary actor who has access to and control over all comfort-related features of the vehicle. The driver initiates and interacts with all the listed use cases.
- **Passenger**: A generalized version of the Driver who inherits access privileges.

The following are the core functionalities represented as use cases in the Comfort Control Unit:

1. **Adjust the Temperature:** This use case enables the driver or the passenger to modify the interior temperature for better comfort. Includes an $\ll$**extend**$\gg$ relationship with **Changing Airflow**, signifying that airflow modification is an optional or conditional extension of temperature adjustment.
2. **Lock/Unlock Vehicle:** This use case enables the driver to operate the vehicle's central locking system through the comfort control interface, providing full control over all doors. In contrast, the passenger has limited access, typically restricted to locking or unlocking the door adjacent to their seat.
3. **Adjust Side Mirror:** This functionality allows the driver to modify the orientation of the side mirrors for optimal visibility.
4. **Adjusting Seat:** This use case facilitates customization of seat positioning, such as recline, height, and lumbar support.

5. **Light Adjustment:** Provides the driver with control over both interior and exterior vehicle lighting systems. Additionally, the passenger has access to adjust interior lighting for their convenience.

3.3 Block Definition Diagram

Using the Block Definition Diagram, the overall structure and composition of the CCU can be represented. As a disclaimer, when the term '*reference*' is mentioned, this is referring to a **Shared Association**. Moreover, when the term '*composed*' is mentioned, this refers to a **Part Association** relationship.

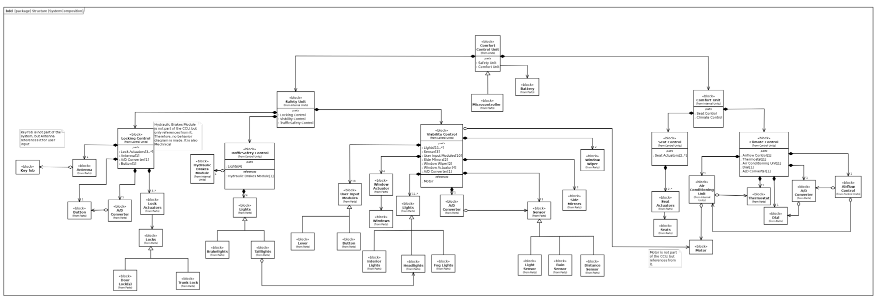


Figure 3: CCU System Composition

The CCU is **generalized** as a Microcontroller block that manages other internal control units. It also has a **Reference Association** to a Battery. This means the CCU is structurally associated with the Battery. In Gaphor, this is represented as "Direct Association". The CCU is *composed* of the **Safety Unit** and the **Comfort Unit**. In each of the mentioned **Units**, there is a A/D Converter that has a compositional relationship within the unit in order to take inputs from the user.

The **Comfort Unit** is *composed* of **Seat Control**, **Thermostat**, **Dial**, **Airflow Control** and **Air Conditioning Unit**. The Airflow Control *references* from the Air Conditioning Unit, as well as inputs from the Dial via the A/D Converter. The **Air Conditioning Unit** *references* from the internal Motor state which will be expressed in a future State Machine Diagram. This Unit is responsible for managing inputs from the user to control the internal Climate.

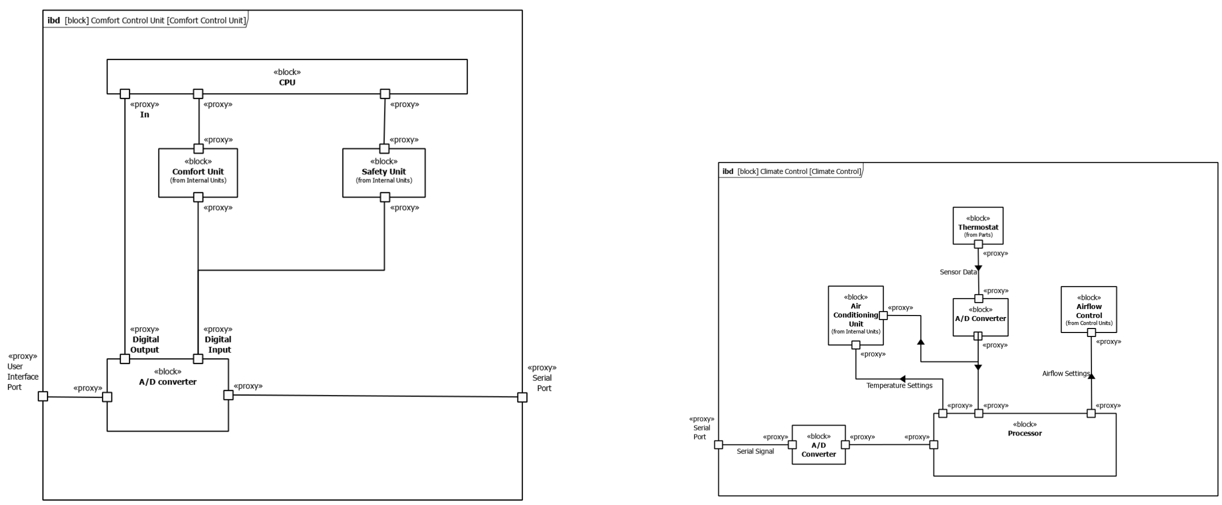
The **Safety Unit** is *composed* of the **Locking Control**, **Traffic Safety Control** and **Visibility Control**. The Locking Control is *composed* of an **Antenna**, **Button** and **Lock Actuators** with minimum 3 Actuators. This is because that there will be 2 Actuators, one for the Driver side and the other for the Passenger side. The last Actuator is for the trunk. The A/D Converter will *reference* from the input of the user via a Button. Additionally, the Antenna will take inputs from the Key fob via an RFID signal which is why there is a *reference* connection.

The **TrafficSafety Control** is *composed* of a **generalization** of lights which includes: (Brakelights and Taillights). Notably also has a *reference* connection to the **Hydraulic Brakes Module**. This Unit is responsible for controlling the traffic safety lights in normal operations while the vehicle is running.

The **Visibility Control** block is *composed* of a **generalization** of **User Input Modules** which includes: Button and Lever, **generalization** of **Sensors** which includes: (Light Sensor, Rain Sensor and Distance Sensor), **Window Wiper**, **Side Mirror**, **Window Actuator** and **generalization** of Lights which includes: (Interior Lights, Headlights and Fog lights). This Unit is responsible for allowing the user control over the visibility while on the road. Notably, this Unit *references* from the Hydraulic Brakes Module, which is not part of the CCU. Through the Block Definition Diagram, we are able to gather a High-level view of how the system is connected and composed.

3.4 Internal Block Diagram

The Internal Block Diagram elaborates on the internal functions of a block and its constituent parts by connecting the **internal blocks** via **ports** and **connectors** to visualize flows of different **items**.



(a) IBD of the CCU

(b) IBD of the Climate Control System

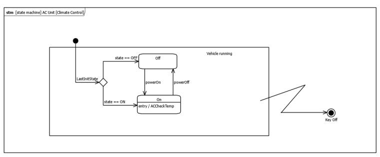
Figure 4: Internal Block Diagrams

4a The **CCU** is internally composed of a **port** allowing it to interface with the user interface connected to an **analog to digital converter**, which sends its **digital output** to the **central processor** of the CCU, which then decides if the signals it is receiving is relevant to the **Comfort** or the **Safety Unit**. The output of both will then be sent through the **A/D Converter** and then to the **Serial Port** that allows it to communicate with other subsystems.

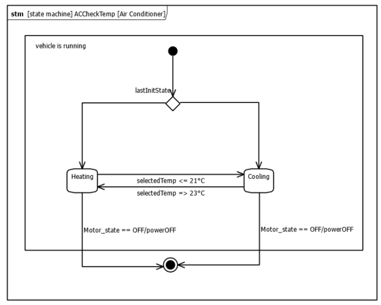
4b One of these subsystems is the **Climate Control**, which communicates via its **Serial Port**. The Signal from it goes through an **A/D Converter** into the **Processor**, which then forwards the signal to the **Air Conditioning Unit** and the **Airflow Control**. The **Central Processor** also takes an input from a **Thermometer**, which also inputs to the **Air Conditioning Unit**.

3.5 State Machine

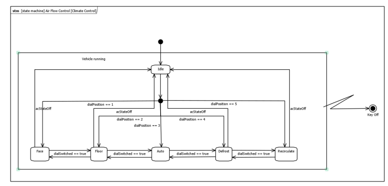
The state machine diagram models the behaviour of systems through the use of possible states and the interactions between them. It shows how the system reacts to moving from one state to another, and what actions are triggered during the transition between these states. In our example, it helps us visualize how the CCU manages different functions in a vehicle and its response to external triggers. The state machine diagrams for the various functions in the CCU will be explained below in detail.



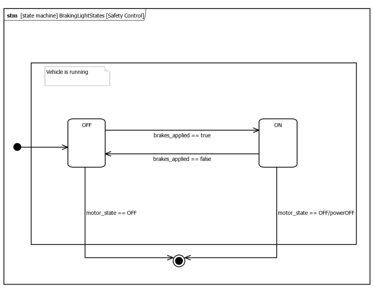
(a) AC Unit State Machine



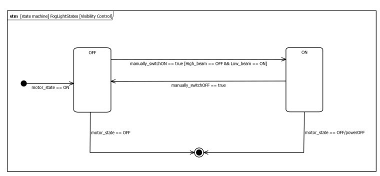
(b) AC Check Temperature State Machine



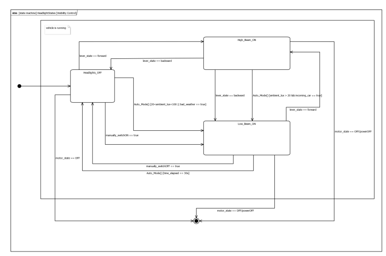
(c) Airflow State Machine



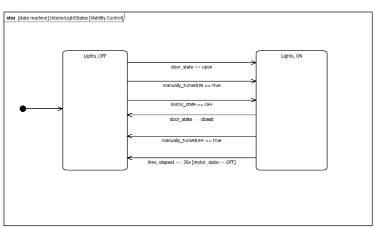
(d) Brake Lights State Machine



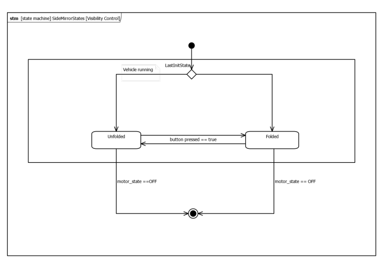
(e) Fog Lights State Machine



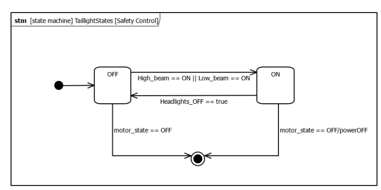
(f) Headlights State Machine



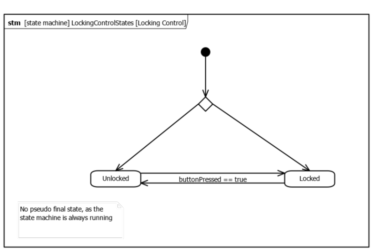
(g) Interior Lights State Machine



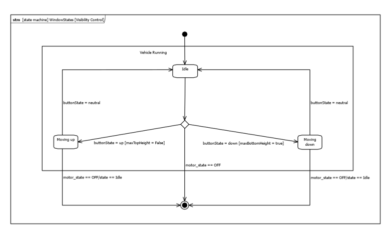
(h) Side Mirrors State Machine



(i) Tail Light State Machine



(j) Locking Control State Machine



(k) Window Control State Machine

Figure 5: CCU System State Machine Diagrams

In figure 5a, the state machine for a vehicle's ac unit is shown. After we start the vehicle, the **climate control**(a subsystem in the CCU) checks the last state of the the ac unit, which can be either **on** or **off**. If we start from off, we have the option to power on the ac unit and proceed further with its features. If we start from on, first the temperature is checked, which will be explained in more detail in 5b. Otherwise the ac unit can be switched off. Since we are assuming that the vehicle is running, there is an interrupt type behaviour to simulate what would happened if the key would be removed.

The ac temperature check state machine works similarly to the one in 5a. As it can be seen at 5b, we firstly check the last state the system was in, either heating or cooling air. If the last selected temperature was greater or equal to 23°C, then the system would be in the "heating" state, otherwise if the last selected temperature was lower or equal to 21°C, the system would be in the "cooling" state. If we turn off the motor, then the system would stop working, which is represented in the diagram by a final state.

The Airflow control function, shown in Figure 5c, is part of the **climate control** subsystem and is responsible for regulating the direction and intensity of air distribution inside a vehicle's cabin. This function is dependent on the **AC unit**, as it only operates when the AC is turned on. In the state machine diagram, the system begins in the *Idle* state. Depending on the dial position (1–5), it can transition between the following states: *Face*, *Floor*, *Auto*, *Defrost*, and *Recirculate*. These transitions occur based on user input via the dials. If the AC is turned off, the system returns to the *Idle* state regardless of the current airflow mode. An interrupt type behaviour to simulate what would happened if the key would be removed.

In the figure 5d, it shows the "Braking Light States" state machine starts in the **OFF** state, meaning the brake lights are off while the vehicle is running. It moves to the **ON** state when the driver applies the brakes, turning the brake lights on. The system returns to the **OFF** state when the brakes are released, and if the motor is turned off, it transitions to the final state, indicating the vehicle has stopped operating.

The "Fog Light States" state machine in figure 5e, it shows the begins in the **OFF** state when the vehicle's motor is turned on. The fog lights switch to the **ON** state when the driver manually turns them on, but only if the high beam is off and the low beam is on. The fog lights return to **OFF** when manually switched off, and the system reaches the final state when the vehicle's motor is turned off.

The "Head light States" state machine starts with the headlights turned **OFF** when the vehicle is running as you can see in figure 5f. The headlights can turn on to **high beam** directly if the driver pushes the lever forward. They can also turn on to **low beam** either when the driver switches them on manually or automatically when it becomes dark or the weather is bad. While the headlights are on, the system switches between low beam and high beam based on the lever position or if another vehicle is approaching. The headlights turn off either manually or automatically after a certain time, and the system reaches its final state when the vehicle is turned off.

The "Interior Light States" state machine controls whether the interior light is **OFF** or **ON**. The light turns **ON** when the door is open, the user switches it **ON**, or when the motor is turned **OFF**. It turns **OFF** when the door is closed, the user switches it **OFF**, or after 30 seconds has elapsed from the time it was turned **ON** after the motor was switched **OFF** as described in figure 5g.

As shown in figure 5h, this state machine diagram manages the side mirror behaviour with two main states: **Folded** and **Unfolded**. It starts at the initial state, then reaches a decision node labelled **LastInitState**, which checks the last known position of the mirrors before the system was turned off and transitions accordingly. When the vehicle starts running, depending on the last position, the user can toggle between **Folded** and **Unfolded**. When the motor is turned off, the system reaches the final state, preserving the current mirror position to restore it during the next initialization.

The "Taillights States" state machine diagram models the taillight behaviour in a safety control system. It has two states: **OFF**, where taillights are turned off, and **ON**, where taillights are active due to the headlights being in use. The system transitions to a final state when the motor is turned off, ending the taillight operation as modeled in figure 5i.

The "Locking Control States" state machine in figure 5j, represents the locking control system of a vehicle. It includes two states: **Locked**, when the doors are secured, and **Unlocked**, when the doors are accessible.

The "Window Control States" state machine models the window control system while the vehicle is running. It includes three states: **Idle**, when the window is stationary; **Moving Up**, when the window is rising; and **Moving Down**, when the window is lowering. The system ends in a final state when the motor is turned off, stopping window operations as it can be seen in figure 5k.

3.6 Activity

An activity diagram models the flow of interactions or operations in a system. It emphasises inputs, outputs, sequences, and conditions, showing how the system behaves in response to events or inputs. An activity diagram primarily consists of the following components, like initial nodes, action nodes, and decision nodes, which aim to represent the flow of control within a process. These elements help visualize the sequence of operations, possible decision points, and how the system transitions from one activity to another.

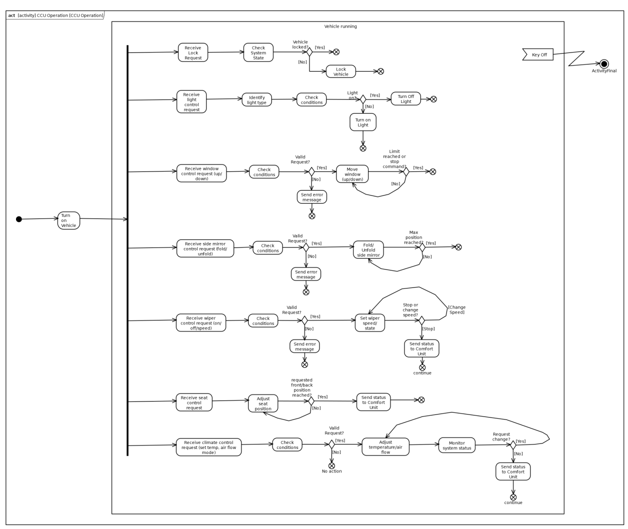


Figure 6: CCU System Activity Diagram

The system begins the activity sequence with **Turn on Vehicle**, afterwards, the system begins in parallel using the fork node, waiting for user input in the different Use cases of **Locking/Unlocking Vehicle**, **Light Adjustment**, **Adjusting Temperature**, **Changing Airflow**, **Adjusting Side Mirror** and **Adjusting Seat**. In most cases, we first receive an input from the User depending on what they want, then we check if their request is valid. For example, in the Changing Airflow Use Case where the user opens a Window, a valid request would only be where the button state is either upwards or downwards. An invalid request would be when it is upwards and downwards at the same time. The system then responds to the current conditions of the module, reacting differently depending on the current stored state. The activity only ends when the User turns off the motor of the Car using their key which interrupts the activity.

3.7 Sequence

The sequence diagram represents the interactions or exchanges between multiple system components, like actors and systems or parts of the system.This is done through ordered message exchanges to carry out a specific process or function, highlighting the order of operations.

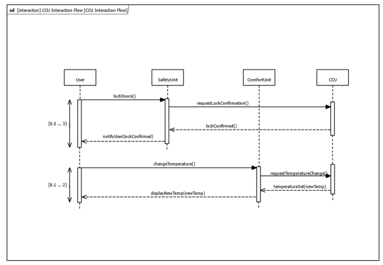


Figure 7: CCU System Sequence Diagram

In our case, the sequence diagram is meant to show the interaction between the user, control units (safety or comfort unit) and the CCU for handling various functions in a vehicle. To keep the model simple, this sequence diagram includes only two examples: one showing e message sent to the **comfort unit** and one to the **safety unit.** This illustrates the typical user interaction, as communication flow for the other features follow a similar pattern.

4 Conclusion

This technical report aimed to model and represent the Comfort Control Unit (CCU) of a vehicle using SysML. We focused on core functionalities split into two different systems focusing on **Safety** and **Comfort**, representing their structure and behavior through various SysML diagrams. The modeling process using Requirement Diagrams and Block Definition Diagrams revealed key relationships between components and clarified the overall objective of the system. Moreover, such views of the system were further refined by applying Non-Functional Requirements to normal Requirement blocks, such as the aspect of Reliability to the Locking Control Unit. SysML proved invaluable in offering a standardized approach to system representation, allowing for better traceability and understanding. While some parts of the system were simplified for clarity, future work could involve modeling error states, further safety integration, and broader vehicle subsystems.