

Truck Platooning System

Namik Mert Tunçbilek
Matric No : 7213712
Embedded Systems Engineering
Fachhochschule Dortmund
Dortmund, Germany
namk.tuncbilek001@stud.fh-dortmund.de
Contribution Percentage: 27

Veli Ateş
Matric No : 7213717
Embedded Systems Engineering
Fachhochschule Dortmund
Dortmund, Germany
veli.ates001@stud.fh-dortmund.de
Contribution Percentage: 8

Nijat Dashdamirov
Matric No : 7213892
Embedded Systems Engineering
Fachhochschule Dortmund
Dortmund, Germany
nijat.dashdamirov002@stud.fh-dortmund.de
Contribution Percentage: 23

Vignesh Arumugam
Matric No : 7213710
Embedded Systems Engineering
Fachhochschule Dortmund
Dortmund, Germany
vignesh.arumugam002@stud.fh-dortmund.de
Contribution Percentage: 30

Mirudhubashini Ramasamy Sridharan
Matric No : 7213893
Embedded Systems Engineering
Fachhochschule Dortmund
Dortmund, Germany
mirudhubashini.ramasamysridharan002@stud.fh-dortmund.de
Contribution Percentage: 12

Abstract—Truck Platooning is the future of transportation in which trucks drive agreeably at less than 1 moment separated made possible by automated driving innovation. Transportation companies' advantage from lower fuel utilization and enhancements in (driver) efficiency, whereas society benefits from less mishaps, more secure activity and less congested streets, and lower carbon outflows.

Keywords—Platooning, mechanized driving, lower carbon

I. INTRODUCTION

Transportation is vital to society and economy, and street cargo transportation accounts for about 60% of all surface cargo transportation. The request for street cargo transport is anticipated to extend within the coming a long time. As appeared within the American Trucking Association's 2015 report, the trucking industry comprises about 80% of a 1.33 trillion-dollar shipping and coordination's industry within the US. In any case, bounty of fuel utilization and nursery gas outflow have been created. For case, street transport speaks to around 27% of the vitality utilization of the European Union. Furthermore, indicated that vehicles account for 20% of the entire carbon emanation of which a quarter comes from overwhelming obligation vehicles (HDVs). Subsequently, the natural impacts amid the method of transport got to be diminished direly. In expansion, the fetched of fuel has an expansive share of add up to transportation costs. Fuel fetched spoken to about 30% of the life cycle taken a toll of owning and working a truck. Additionally, agreeing to the American Transportation Investigate Institute's (ATRI) later report, fuel is respected as the moment biggest taken a toll, where the most elevated is faculty cost. With a huge sum of HDVs and the expanding request for street cargo, it can be anticipated that indeed small advances in fuel efficiency can decipher into significant taken a toll diminishment [1]. And it is additionally advantageous to realize the objective of natural assurance due to less deplete gas. As a result, it is of awesome advantage to move forward fuel economy, and how to decrease fuel utilization amid traveling has turned into a prevalent theme in later a long time. Luckily, the improvements of shrewdly transportation frameworks (ITSs) have empowered strategies to improve the vitality productivity of transportation systems. A promising approach to managing with that issue is to decrease the crevice between vehicles on the street, which is ordinarily called truck companies. Truck companies,

moreover, known as caravans, are a set of vehicles shaping a street prepare by traveling closely in single record to encounter diminished discuss drag. This may altogether diminish fuel utilization since around one-fourth of the fuel utilization is important to streamlined drag. As a result, fuel economy can be progressed, and natural invitingness can be accomplished due to less nursery gas emanation in a unit. Separated from fuel reserve funds, truck units can contribute to an increment of street capacity and can ease activity blockage by a littler hole between vehicles. In later a long time, with the advancement of independent driving innovation, vehicles are prepared with a few sensors that empower them to watch their environment and choose in genuine time what activity ought to be taken, which are called "autonomous vehicles" or "driverless vehicles." Driverless vehicles can arrange their way when driving, and they can travel in a company with littler interims to diminish fuel utilization [2]. Moreover, when driving naturally in a company, it is conceivable to diminish the hazard of rear-end collisions and to move forward activity security. With incredible points of interest said over, vehicle companies have pulled in the consideration from numerous governments and inquire about teach. As a result, a few ventures related to companies were proposed. The primary thinks about on truck computerization were "Chauffeur" inside the EU venture T-TAP from the mid-1990s to the starting of 2000. Amid the extend, Cap and Fritz conducted an explore with two trucks coupled by an "Electronic Tow Bar" to measure the fuel reserve funds. A while later, the California Way program begun it inquire about on heavy truck platooning. Within the Way program, all vehicles were completely mechanized, counting the pioneer. For case, in 2004, the program performed a fuel utilization test with two pair trucks connected by an electronic control framework for diverse spaces.

II. EASE OF USE

A. Automated Driving Techonology

Computerized Driving innovation offers the plausibility of in a general sense changing transportation. The objective of innovation is to create vehicles drive independently in a secure and comfortably way. Preparing cars and large goods vehicles with this innovation will likely diminish mischances, fuel utilization, contamination, and blockage Many frameworks that are portion of innovation are as of now commercially

accessible, such as Versatile Voyage Control (ACC), Lane Keeping Help (LKA), Independent Crisis Braking (AEB) and Mechanized Stopping or stopping help platooning builds upon these innovations, by creating the Agreeable Versatile Voyage Control (CACC). The SAE Universal Levels of Robotization for On-road Vehicles records 5 levels from no robotization to full mechanization, where truck unit can be set from levels 2 to 4 [3]. In common, Advertisement innovation are mechanical frameworks that ‘sense’ the environment employing a combination of sensors, such as lidar, radar, and cameras. The sensors can make up for each other’s shortcomings and give repetition. For occasion, on the off chance that it is greatly foggy on the street, cameras are essentially futile. However, radar and lidar still work and compensate for the need of data given by the camera. For localisation, a mechanized vehicle can utilize worldwide situating frameworks (GPS) and inertial route frameworks (INS) [4]. Once more, if GPS comes up short briefly, INS can take over utilizing accelerometers (movement sensors) and spinners (turn sensors), making a difference the vehicle to orientate until GPS comes back online. This is often a likely situation when driving in tunnels where GPS does not work, but the car is still able to orientate itself utilizing INS. [5] For remote communication, a particular Wi-Fi standard has been endorsed.

B. Characteristics of the Application Domain

Reactive systems is a system in which the ECU continuously check with the voltage data for output failures [open & short circuit] triggered by external environment and temperature readings for burnt failures. Real-time systems come under Hard-real time. Continuous/discrete/hybrid systems has discrete character in time. Values of output voltage & temperature readings are acquired in a discrete manner in a pre-defined time. Dependable systems address reliability attribute by having certain number of life cycles or warranty periods. Availability is required, since this is needed to ensure its functions availability as soon as BCM [Body Control Module] gets powered up and need to exists till the power gets down. Also, this system has Safety standard ASIL “Automotive Safety Integrity Level” which highly tells the safety functions to avoid accidents or failures.

III. MODEL SELECTION

The analysis of the model has taken place an important role to design the required model. The tasks are done in the following alphabetical order to complete the whole project from analysing to testing.

A. Requirement Diagram[Author: VigneshArumugam]

The requirements are the base for any system’s development. It explains how the system wanted to be. And the requirement acquisition of this project started with elicitation process. During this part, one of the team members acted as the requirement engineering and others acted as customer. The requirement engineer gathered the domain, context, and problem knowledge before starting the elicitation process and it happened in the mode of interaction [LEC 1]. After elicitation, requirement analysis started with identifying the important requirements and iterated till finalizing. The overall requirement diagram was added to the package. Each individual requirement has been given more importance, because as per statistics the percentage of software errors and cost of rectifying those errors are 82%

and 56% respectively for the requirements [LEC 1]. Which is the highest of all other part of the life cycle.

Each requirement is created with unique ID, and they got incremented when the corresponding requirement get updated. A requirement block is refined by another requirement with more quantitative requirements or with behavioral diagrams. And the requirements can be satisfied by structural diagrams [LEC 1]. Corresponding test cases of the requirements are added as verifiable. Base and fundamental requirements are divided into multiple requirements by containment. For this truck platooning project 19 requirements blocks are created, and its structural appearance is shown in Fig.1

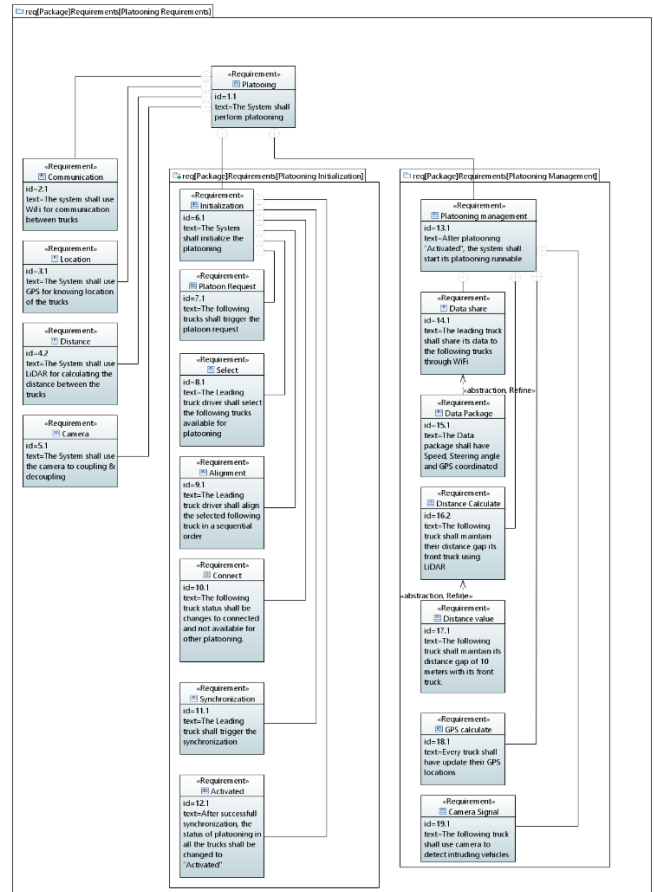


Figure 1: Requirement Diagram of Truck Platooning

B. Use Case Diagram[Author: Namık Mert Tunçbilek]

Use case diagram defines the operations that users want to perform on the system through a function and consists of four main elements: Actors, System, Use Cases, and their relations. Use case diagrams show the context and requirements of entire system or some parts of the system.

In Platooning System, four different use case diagrams were created for four different events. These cases are platoon administration, joining to the platoon, data sharing in platoon and collection data from sensors respectively.

In first case, platoon administration, actors are truck driver, leading truck and following truck. All trucks in a platoon have a truck driver like an operator who is responsible for platoon management. Trucks drivers perform creation, joining, leaving, and ending the platoon with for

instance HMI. Following truck is the second actor for joining and leaving, and leading truck is second actor for creating and ending the platoon.

Secondly, we have use case diagram to show operations in joining to the platoon. In this case, first actor is truck which ready to join to the platoon and the other actor is trucks in platoon. Waiting truck sends request, shares its GPS data to give instant coordinates and joins to wireless network of platoon. Also trucks in platoon shares their coordinates and related accept the request of waiting platoon.

Third use case diagram illustrates the data sharing in platoon system. All trucks in platoon are responsible for sharing these data with each other.

In the last diagram, truck in platoon and sensors which are GPS, camera and LIDAR are actors. GPS is for generating route plan, camera is responsible for recognizing environment such as intruding cars, traffic lights, and pedestrians during the driving, LIDAR is responsible for calculating distance continuously. In addition, camera, and LIDAR, both are used for coupling & decoupling during the driving.

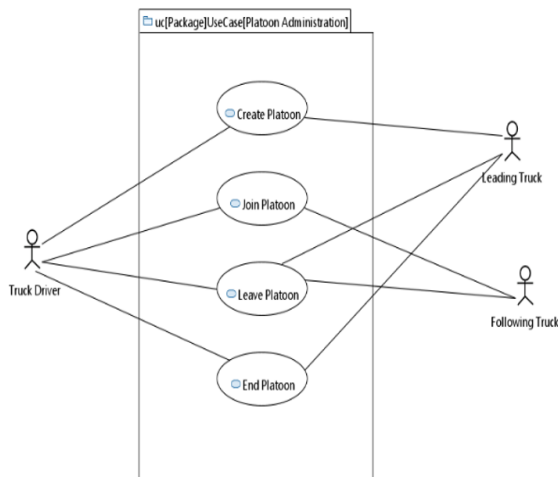


Figure 2: Platoon administration use case diagram

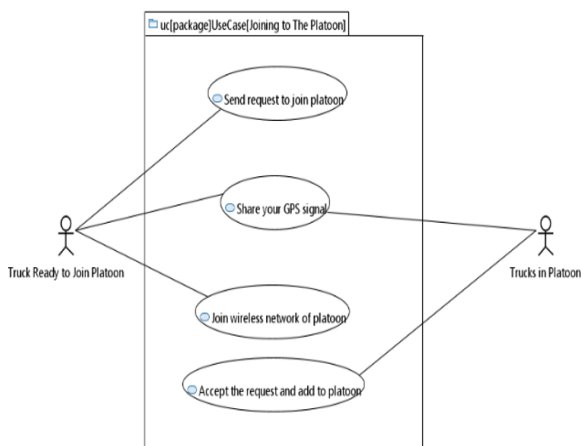


Figure 3: Joining platoon use case diagram

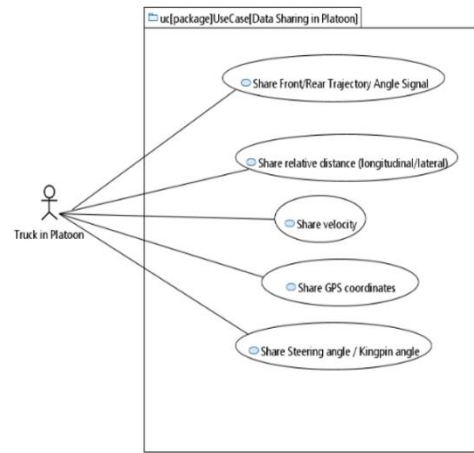


Figure 4: Platoon data sharing use case diagram

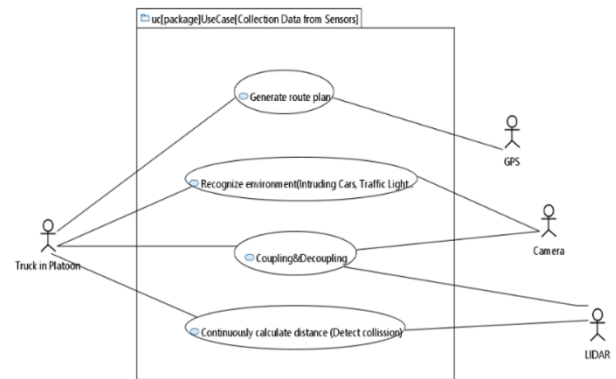


Figure 5: Platoon data collection use case diagram

C. Activity Diagram [Author: Mirudhubashini Ramasamy Sridharan]

An Activity diagram is a behavioral diagram (it depicts the entire behavior of the system). It shows the control flow from start point to a finish point showing the various decision paths that exists while the activity is being executed. Activity diagrams show both computational and organizational forms (i.e., workflows). Activity diagrams are developed from a restricted number of shapes, associated with arrows. The foremost vital shape types: ellipses speak to actions; diamonds speak to decisions; bars speak to the begin (part) or conclusion (connect) of concurrent activities; a dark circle speaks to the begin (introductory hub) of the workflow; an encompassed dark circle speaks to the conclusion (last node). Arrows run from begin towards the conclusion and speaks the order in which the exercises flow.

The activity diagram of Truck platooning is, initially the lead truck initiates the communication to the following truck, the following truck receives and send signals through the sensors used. The lead truck waits for the communication if the signal received it sends request to join and shares the GPS signal to the following truck. The following truck receives the data and accepts the request, and the platoon is added. If the lead truck didn't receive signal, then it waits for the communication from the following truck.

Once the following truck is added the lead truck detects the environmental intruders via LIDAR and camera located in the front of each truck. If any intruders detected it shares the coordinates like distance, speed, velocity and transmits the signal to following truck and truck either slows down or stops. If there are no intruders, then the cycle goes back to the data received from followers and the cycle continues again.

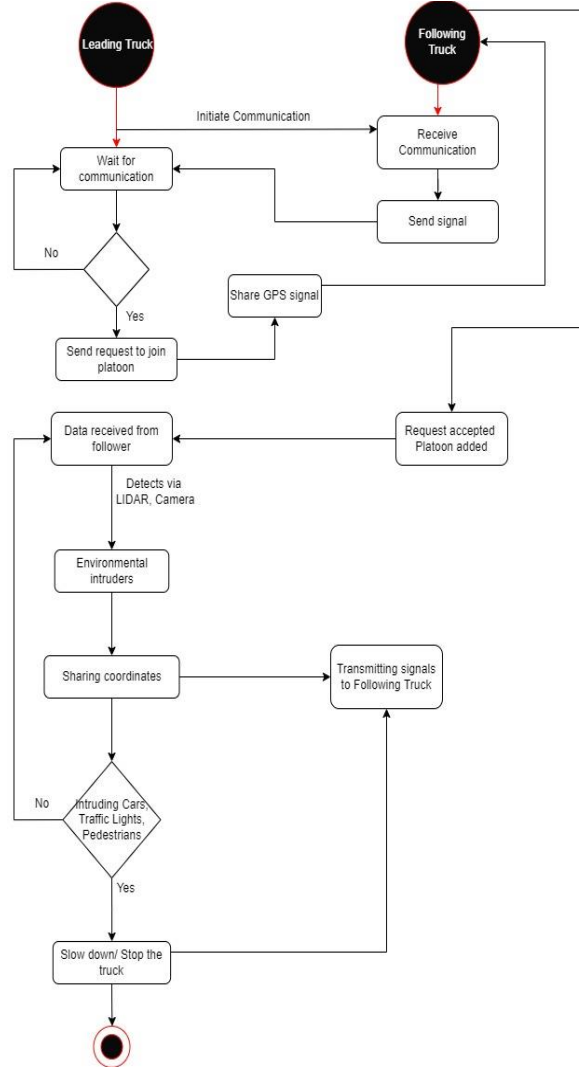


Figure 6: Activity Diagram of Truck Platooning

D. Block Definition Diagram[Author: VigneshArumugam]

The block definition diagram is the very first diagram to represent the structural appearance of the system. Elements which are satisfying the requirements are considered as blocks. The whole truck platooning requirements are analyzed, and the components of the systems are considered as the block and the relationship between the blocks are done with association connectors. The blocks are added with its corresponding attributes and operations.

Blocks and Parts classifications are followed in this truck platooning block definition diagram. For example, in Fig.7 the sensor is the “Block” and “GPS”, “LiDAR”, “Camera” management are the parts. Blocks are used for definition and parts are used for application [LEC 2]. As mentioned earlier, the relationship between the blocks is handled with association [Composite] with multiplicity.

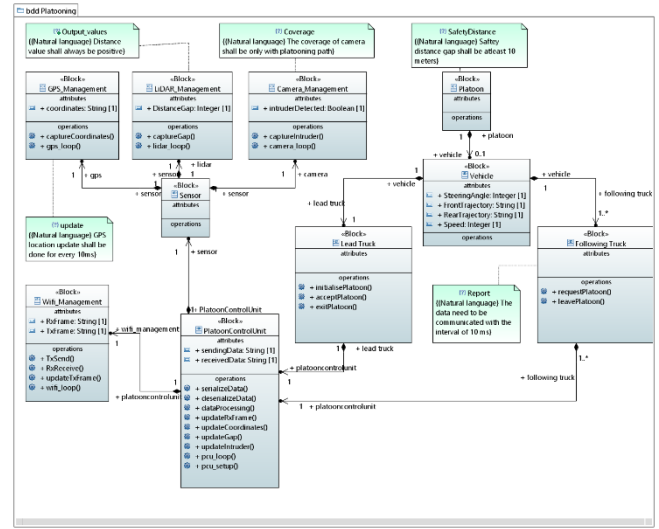


Figure 7: Block Definition Diagram of Truck Platooning

E. Sequence Diagram[Author: Nijat Dashdamirov]

Sequence diagram is used to describe types and orders of the interactions between the objects. In our case we have 4 main objects: “Platoon Central Unit”, “GPS”, “Camera” and “Wi-Fi”. “Process Central Unit” is used to control all the system and it is main object in the diagram. Firstly, PCU sends UpdateTXFrame() message to update credentials in the Wi-Fi object. Then UpdateRXFrame() is sent from Wi-Fi object to the PCU to confirm that message received successfully. Once it is done, PCU is getting and updating the coordinates of the leading truck via UpdateCoordinates(). Lidar object plays the role in getting the distance between the trucks and PCU receives information from UpdateGap() method. To check the platooning is working flawlessly and there is no un-platooned truck in the system, every time we are getting information from the camera. Once camera senses an intruder in the system, and it sends message UpdateIntruder() to the main block to inform controller.

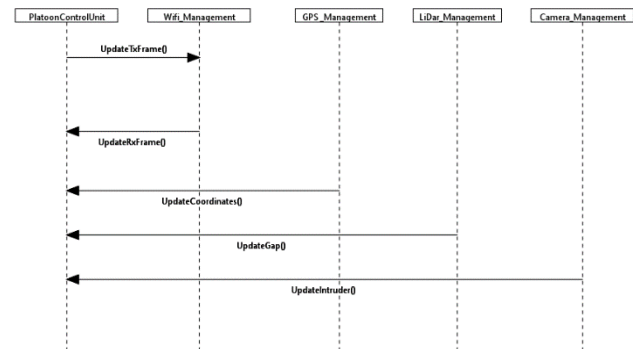


Figure 8: Sequence Diagram of Truck Platooning

F. Parametric Constraint Diagram[Author: VigneshArumugam]

Critical performance parameters are considered as constraints. Parametric constraint diagram has been added along with the block diagram. The constraints of the block are mentioned in the natural language. Fig.9 represent the constraint diagrams added in the block definition diagram.

Here, in the constraint of the GPS Management block talks about the periodicity of the GPS module of the system.

So, this constraint shall be the input for task scheduling and verifiable by scheduling test cases.

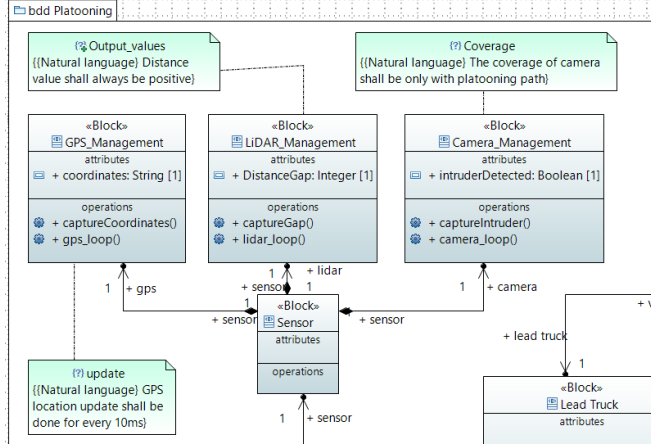


Figure 9: Constraint Diagram of Truck Platooning

G. Allocation Diagram[Author: Namık Mert Tunçbilek]

Allocation diagrams are typically used to define relationship of various parts of the model. Our platooning system basically consists of four main block and their allocation.

Control unit block processes data which are coming from sensors by its functions and use wireless communication to provide communications between trucks. Vehicle block and its functions are associated with all blocks to perform tasks. Wireless communication system has relation with sensors, control unit and vehicle block to receive and transmit necessary data for responsible units in the platoon.

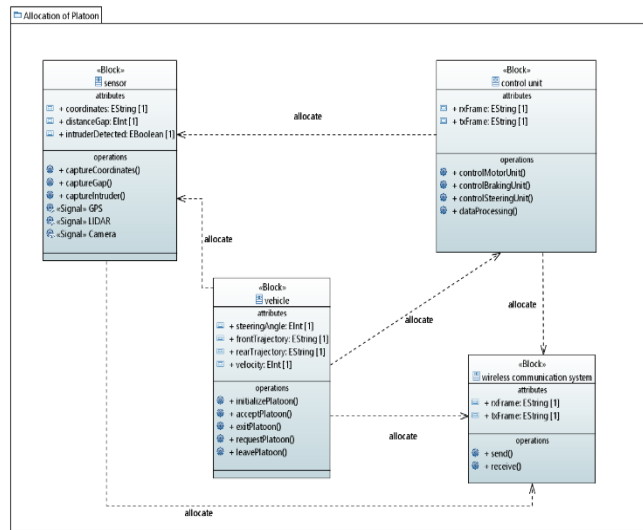


Figure 10: Allocation Diagram

H. Internal Block Diagram[Author: Namık Mert Tunçbilek]

First internal block diagram illustrates internal structure of the platooning system in terms of its subsystems. Platooning system basically consists of five different subsystems which are power subsystem, sensor subsystem, communication subsystem, control processing unit and safety unit. All subsystems are related to each other through communication system which is Wi-Fi between the trucks and serial communication between the units.

Second internal block diagram shows the fundamental internal units of the vehicle block. It consists of mainly two

units (control subsystems and sensor subsystems) to perform its tasks. Control subsystems consists of steering control unit, breaking control unit and motor control unit, while sensor subsystem is formed by HMI, camera, GPS and LIDAR. While HMI unit, GPS and LIDAR are used once, each truck must have more than one camera in order to perform necessary tasks.

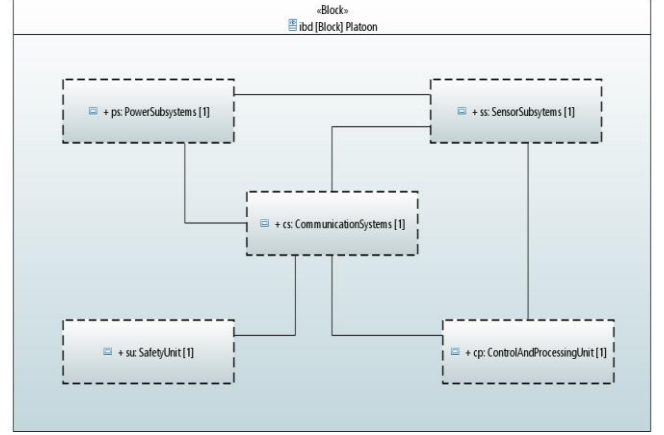


Figure 11: Platooning Internal block diagram

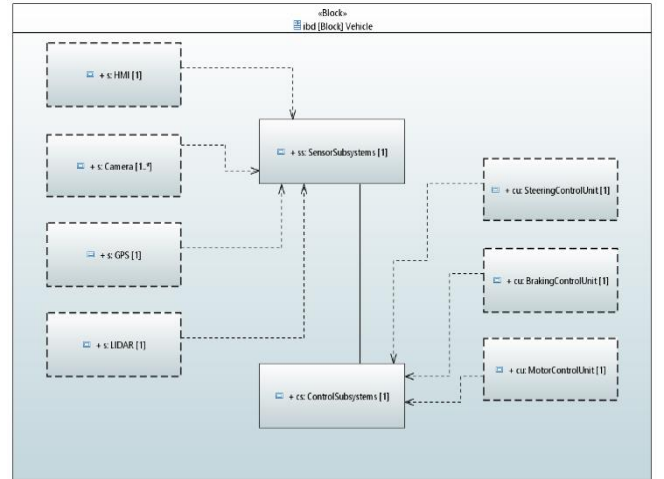


Figure 12: Vehicle Internal Block diagram

I. State Machine Diagram[Author: Veli Ates]

State machine diagrams are typically are used to describe state-dependent behavior for an object. An object responds differently to the same event depending on what state it is in. State machine diagrams can be shown uniquely depending on the system and components that are going to be chosen based on requirements. In this case, 3 different components(Vehicle, Sensor and Platoon) are chosen to show their behaviors in spesific circumstances.

- **Vehicle Component:** The vehicles in the platoon may vary according to the user's requirements. But every vehicle must have some states in the platoon. In our case, “vehicle component” has 9 states. Each state represents the current position of the component. The main purpose of the whole tasks is to make sure if the component works properly.

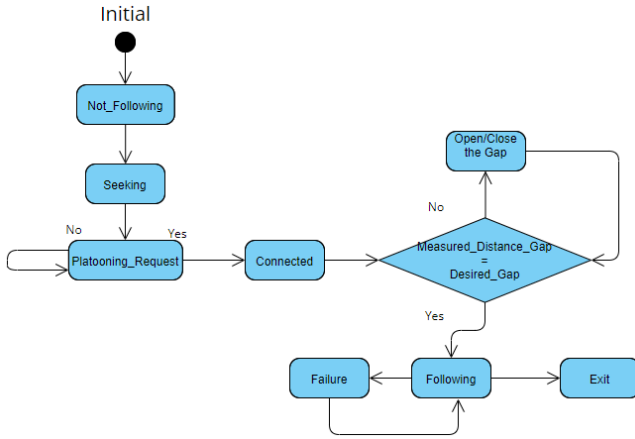


Figure 13: State Diagram of Vehicle

- **Sensor Component:** The sensor component is used to ensure if the distance between the vehicles is equal to the safe distance. In order to do that, sensor has to receive signals from leader vehicle to calculate the distance.

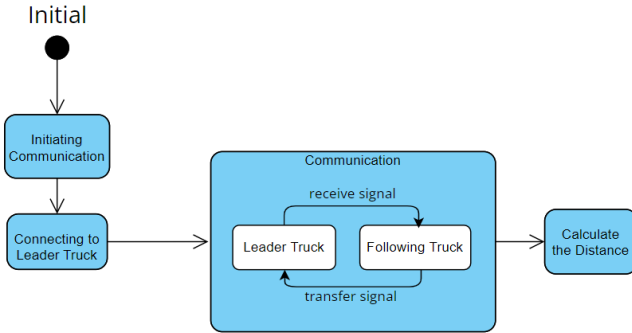


Figure 14: State Diagram of Sensor

- **Platoon Component:** Platoon is a component represents of the whole system. This component has 7 states in which all the tasks should be done properly. When if the missions don't go as expected and the "platoon" encounters an emergency, platoon goes into "Failure" state and tries to figure out the problem to make sure whole system is secured.

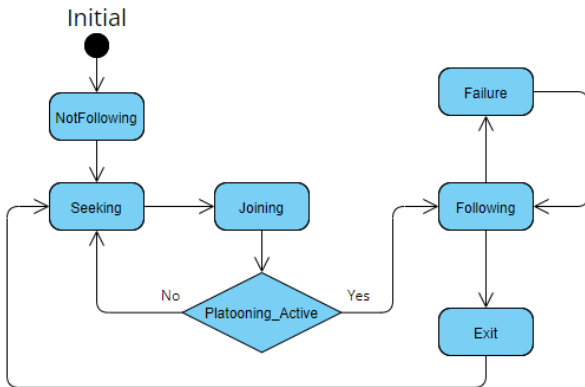


Figure 15: State Diagram of Platooning

IV. IMPLEMENTATION SECTION

After the analysis and design part, the realization has started with implementation section in the following order.

A. Code Implementation[Author: VigneshArumugam]

The code implementation is done in CPP language, and the hardware used is Arduino ESP32. The realization of platooning control unit component is performed with the input documents such as state machine diagram and other analysis diagrams. For witnessing a working prototype, the platooning control unit (PCU) is created with other important components which are required for its execution. The supporting components are added with only basic functions required for communicating with PCU component. So, the overall realization includes PCU, GPS, Lidar, Camera and Wi-Fi components.

Since the output is tried with real hardware ESP32, the software package and hardware for LiDAR and Camera not available. So, to replicate only the behavior a 3-Channel Wide FOV Distance Sensor based on the OPT3101 IC from Texas Instruments is used. This sensor helps to replicate the Lidar and camera component working scenario. Whenever an object is detected in the close range to the OPT3101 sensor, then the Camera and Lidar module consider it as "Vehicle intruded" and "Distance measured is less than Desired Value".

The PCU component is added with all the operations mentioned same as PCU block of the truck platooning bbd. Pcu_loop() operation is considered as the task which shall be considered as task during scheduling. The outputs of the console "without intruder" and "with intruder" are shown in the Fig. 16 and Fig. 17 respectively.

```

Front truck latitude before updating: 200.00
Front truck longitude before updating: 100.00
Front truck latitude after updating: 200.00
Front truck longitude after updating: 100.00
Increasing Speed : Measured_Distance = 401
Maintaining Speed
Front truck latitude before updating: 200.00
Front truck longitude before updating: 100.00
Front truck latitude after updating: 200.00
Front truck longitude after updating: 100.00
Increasing Speed : Measured_Distance = 402
Maintaining Speed
Front truck latitude before updating: 200.00
Front truck longitude before updating: 100.00
Front truck latitude after updating: 200.00
Front truck longitude after updating: 100.00
Increasing Speed : Measured_Distance = 402
Maintaining Speed

```

Figure 16: Output console without intruder maintaining Speed

```

Front truck latitude before updating: 200.00
Front truck longitude before updating: 100.00
Front truck latitude after updating: 200.00
Front truck longitude after updating: 100.00
Increasing Speed : Measured_Distance = 405
Intruder detected Decreasing Speed
Front truck latitude before updating: 200.00
Front truck longitude before updating: 100.00
Front truck latitude after updating: 200.00
Front truck longitude after updating: 100.00
Increasing Speed : Measured_Distance = 405
Intruder detected Decreasing Speed
Front truck latitude before updating: 200.00
Front truck longitude before updating: 100.00
Front truck latitude after updating: 200.00
Front truck longitude after updating: 100.00
Increasing Speed : Measured_Distance = 405
Intruder detected Decreasing Speed
Front truck latitude before updating: 200.00
Front truck longitude before updating: 100.00
Front truck latitude after updating: 200.00
Front truck longitude after updating: 100.00
Increasing Speed : Measured_Distance = 405
Intruder detected Decreasing Speed

```

Figure 17: Output console intruder detected decreasing speed

Form the outputs it can be clearly seen that without intruder detection the truck will maintain its speed, whereas during intruder detection it reduces its speed.

B. Scheduling[Author: VigneshArumugam]

This truck platooning system is a hard real-time system. So, all the task's deadline in this system needs to be completed within all circumstances. For this system only one CPU core is used, and five tasks are assigned to the core. The five tasks are gps_task, lidar_task, camera_task, pcu_task, and wifi_task shown in Fig. 18. The software called Simso has been used to show the schedulable of the tasks.

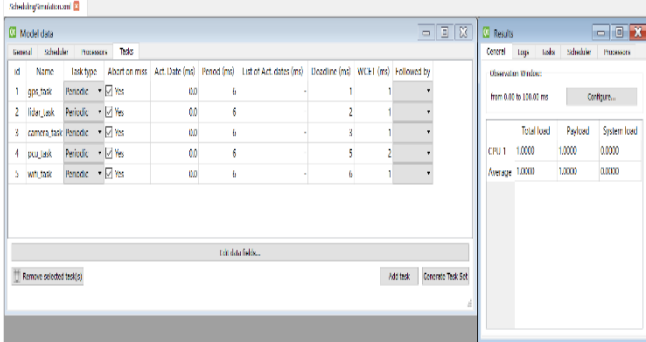


Figure 18: Tasks scheduling

While performing the scheduling, three main constraints are considered, they are timing, precedence, and resource constraints [LEC 2]. The gps_task, lidar_task and camera_task are having higher precedence than pcu_task. And all the tasks have their periodicity as timing constraints which are referred from constraint diagrams. Shared resources like DistanceGap, coordinates and intruderDetected need to be properly accessed and updated. So, considering all these constraints, the EDF (Earliest Deadline First) scheduling seems promising for this system. To achieve the schedulable, the gps_task, camera_task and lidar_task are given earlier deadline than pcu_task. The task simulation can be seen in the fig. 19

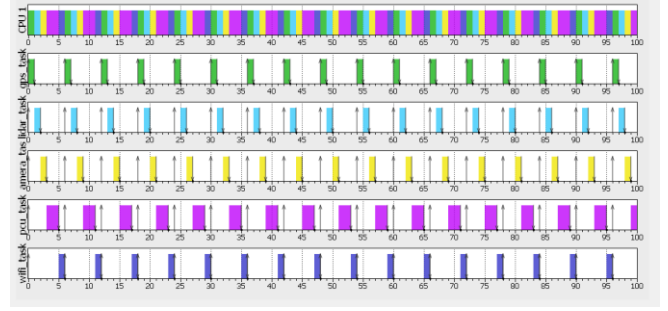


Figure 19: Scheduling Simso

This process also performed in pyCPA(Python Compositional Performance Analysis) tool. The same simso type of task configuration is done in the pyCPA format and achieved the similar output Fig. 20.

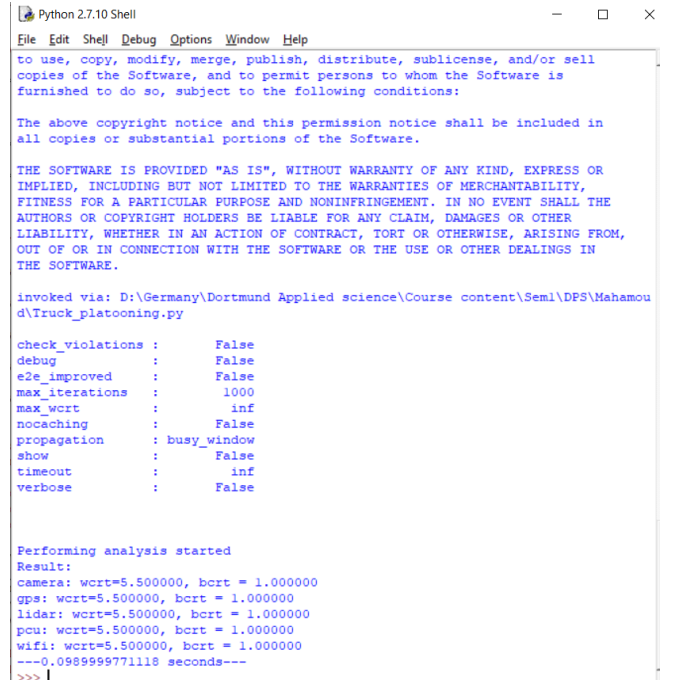


Figure 20: pyCPA Output

V. REVIEW AND TESTING

A. Inspection[Authors: All team members]

The review has been done for all the models, code, and testing phases. Proper iterations were performed to update the negative review commands. For documenting the review process, google sheets were used.

B. Unit testing[Author: Namık Mert Tunçbilek]

The main purpose of unit testing is showing that the units/functions of system work as expected. According to testing experience and common errors, test cases can be varied. Therefore, after the code implementation completed, unit testing is implemented to observe that functions work as expected. To test platooning control unit, five test cases were implemented. These are initial value test, update coordinates test, truck selection test, intruder detection test and gap update test. As a result of the entered reference parameters, it is observed that test cases passed, and the functions worked as expected.

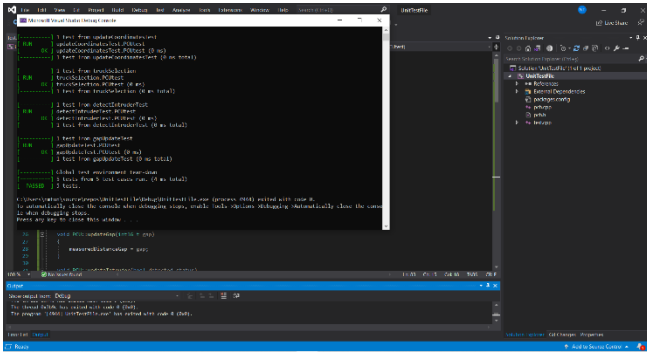


Figure 21: Unit testing

C. Component testing[Author: Nijat Dashdamirov]

Basically, component testing is used to test the all the object that is communicating with PCU.

- Firstly, I tested the Wi-Fi management. In this case I uploaded the credentials of the Wi-Fi and check that if it is updated correctly.
- As coordinates of the leading truck is important in our case, it should be tested. Initially, I give random value to longitude and latitude. If value of the longitude and latitude variables differ from previous value, it means that coordinates updated successfully.
- To check intruder in the system, I changed the intruder value to true. It means that there is un-platooned truck in the system and PCU send decrease speed message to the platooned trucks. By this command we achieve to increase distance between platooned and un-platooned truck to prevent collision from happening.
- Lastly, to verify Radar is working correctly, I increased the distance between the two adjacent truck and system increased the speed until the point that, distance between the trucks will be equal to predefined value.

```
Front truck latitude before updating:
0.Increasing Speed : Measured_Distance = 0Received data from the front truck
0
150
Front truck longitude before updating: Intruder detected Decreasing Speed
0.00
Front truck latitude after updating:
12.00
Front truck longitude after updating:
15.00
Increasing Speed : Measured_Distance =
150
Intruder detected Decreasing Speed
Front truck latitude before updating:
12.00
Front truck longitude before updating:
15.00
Front truck latitude after updating:
12.00
Front truck longitude after updating:
15.00
Increasing Speed : Measured_Distance =
150
Intruder detected Decreasing Speed
Received data from the front truck
Increasing Speed : Measured_Distance =
```

Figure 22: Component testing

CONCLUSION

Via this project, we explored the life cycle of system development. It started from the requirement elicitation process to the testing of those requirements. In each phase of this cycle, the solidity of attaining the system requirements increased gradually. Sometimes, later parts of the life cycle give more data to the former section. For instance, the initial design section documents had some inaccuracies, but they got more influences during the implementation section. That triggered the need for a second iteration for updating the design document. From this scenario, we understood the importance of the designing phase, which is very important to shape the requirements into a system. And this architect or designer needs to think in all aspects like implementation and testing feasibility, and even modifiability.

ACKNOWLEDGMENT

We sincerely thank Prof. Dr. Stefan Henkler, who supported us throughout the journey of this project.

REFERENCES

- [1] Ellwanger S, Wohlfarth E. Truck platooning application. In2017 IEEE Intelligent Vehicles Symposium (IV) 2017 Jun 11 (pp. 966-971). IEEE.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
 - [2] Calvert SC, Schakel WJ, van Arem B. Evaluation and modelling of the traffic flow effects of truck platooning. Transportation research part C: emerging technologies. 2019 Aug 1;105:1-22.K. Elissa, “Title of paper if known,” unpublished.
 - [3] Saeednia M, Menendez M. Analysis of strategies for truck platooning: Hybrid strategy. Transportation Research Record. 2016;2547(1):41-8.
 - [4] Janssen GR, Zwijnenberg J, Blankers IJ, de Kruijff JS. Truck platooning: Driving the future of transportation.
 - [5] Zhang L, Chen F, Ma X, Pan X. Fuel economy in truck platooning: a literature overview and directions for future research. Journal of Advanced Transportation. 2020 Jan 3;2020.
- [LEC 1] ESE_3_Requirements
[LEC 2] ESE_2_02_RT