EH2760 Management of Projects

ESS-CAR/ESS-NW

Stockholm, Sweden

Project Plan

Jonas Ekman
Yini Gao
Jacob Kimblad

Leon Fernandez
Fredrik Hyyrynen
Yifan Ruan

October 10, 2018

Background

Advanced Driver Assistance Systems (ADAS) is one of the hottest research areas in the automotive industry. ADAS should be capable of sensing the vehicle's surroundings, as well as vehicle's own status. To achieve those abilities, multiple sensors and Embedded Control Units (ECUs) are used in the ADAS and a network inside the vehicle is established to combine the data gathered from different nodes.

The two projects "ESS-Car Embedded Services for a Self-Adaptive Car" and "ESS-Network Embedded services for Self-Adaptive Networking" were provided by professor De-Jiu Chen at KTH Royal Institute of Technology. The idea of our projects are from Viktor Karlquist's Master's thesis [1]. He presented a design and implementation of an automotive experimental platform for ADAS, and the "ESS-Car" team from last year's MF2063 course round has already built a autonomous vehicle prototype based on the thesis [2]. Our projects will be focused on the development and extension of the prototype. Instead of traditional IP networks, software defined networking, which centralizes the control plane in the network and allows for a better granular and agile network management, will be implemented by "ESS-NW" team. "ESS-Car" team will implement error injectors on the new prototype which makes the car more autonomous.

Reference Documents

- EH2760 course-pm [3]
- ESS-CAR and ESS-NW entries in the MF2063 project catalog [4]

1 Goals

The goals of the project can be divided into two parts, goals concerning the network implementation on the car and goals concerning the remaining software and hardware system architecture of the car. As both parts are heavily tied into each other some of the sub-goals are dependent on sub-goals from the opposing . The overarching goal of the network implementation is of a more exploratory nature as it is a problem that has not been studied clearly, thus the project aims to explore what different network implementations can be beneficial in this type of infrastructure. The overarching goal can be stated as "The goal of the project is to reconfigure the network using SDN-switches and a Raspberry Pi as a SDN controller". The overarching goal for the software and hardware system architecture can be stated as "The goal of the project is to increase the robustness by revising the existing architecture and implementing additional features such as vision control and a display service." Not related to the technical aspects of the car, but rather the ambitions of the project members, are also the business goals of the project.

1.1 Technical Goals

The technical subgoals are divided up into two parts. Subgoals concerning the ethernet network which takes care of all of the communication between different parts of the system. Subgoals concerning the hardware and system architecture which implements all of the functionalities of the system except for the network communication.

1.1.1 Hardware and Software System Architecture

As discussed in the background, the project builds on an existing architecture. Thus most goals consists of improving the existing architecture or examining alternative implementations. The first subgoal consists of making the system more robust by extending the hardware architecture. This should be done such that there is one unique microprocessor for each sensor and actuator in the system while maintaining the existing functionalities of the car. The next subgoal is to monitor the operations of the ECU's such that the car can be made fail-safe by implementing fault-detection, watchdogs and battery level indication. The next subgoal is to develop a formal boot- and shutdown-sequence such the the car becomes reliable and fail-safe under normal operation. The final subgoal is to extend the functionality of the existing system by setting up a camera module and choosing an AI algorithm to detect some external stimuli such as a traffic sign such that the rest of the system can react to it (choosing the appropriate speed or steering for example).

1.2 Business Goals

The first subgoal is to assemble the car and set up the physical layer, which includes the SDN-switches and the SDN-controller. This is a dependency the other goals.

2 Organization

The requirements of the project were divided into a number of topics, with each team member being focused on two topics. Additionally, Leon and Fredrik have been appointed roles of Project Manager and Secretary, respectively.

2.1 People

2.1.1 Project Members

- Fredrik Hyyrynen, fhyy@kth.se, Software and Computer Vision, Secretary
- Jacob Kimblad, jacobki@kth.se, Software and Control
- Jonas Ekman, jonekm@kth.se, Network and Electronics
- Leon Fernandez, leonfe@kth.se, Network and Software, Project Manager
- Yifan Ruan, yifanr@kth.se, Control and Electronics
- Yini Gao, yini@kth.se, Computer Vision and Software

2.1.2 Stakeholders

- Dejiu Chen, chen@md.kth.se, Stakeholder, Supervisor
- Matthias Becker, mabecker@kth.se, Stakeholder, Supervisor

2.2 Topics and Roles Elaboration

Software - Consists of scheduling the tasks running at the different nodes in the system, managing the bootstrapping of the system upon startup and lastly, writing code for the arduino nodes

Computer Vision - Consists of setting up the Raspberry Pi + Camera subsystem and implementing the commputer vision algorithm

Control - Consists of devising and implementing the main control algorithm that will run on one of the nodes

Network - Consists of configuring the SDN switches and SDN controller as well as writing the application-layer programs that the nodes use to communicate

Electronics - Consists of doing the PCB design for the ”motherboard” in the car as well as mounting the sensors and wiring in the car

Project Manager - Consists of communicating with stakeholders, schedule and lead meetings and handling parts orders

Secretary - Consists of taking notes during meetings and being a temporary managar in the absence of the project manager

3 Project Model

4 Commentary: Time Plan

5 Commentary: Resource Plan

6 Risk Analysis

7 Communication

The main communications channel used is the kth e-mail. Meetings and scheduling is handled with Microsoft Outlook. For shorter and less important messages, Slack is used. Slack also interfaces with Trello, Google Drive and Github, which facilitates keeping the documentation up to date. Physical meetings takes place on a weekly basis during which progress and possible issuesand possible issues are discussed.

8 Documentation

Documentation is a critical part during software development project. It is a good way to communicate the implementation of everyone’s work. So, we need to keep documentation concise and interpretable, for example, overviews and roadmaps can be shown in our documentation to let other teammates know about the work in a short time. Only important information like what is the underlying communication architecture and what type of data it outputs should be elaborate in documentation. Another significant point is the accuracy of the documentation due to some future work will be done base on it. Indexing and linking should also be added to our documentations as references project. We use Google Drive and GitHub to manage our documentations since Google Drive allows us to work on the documentation simultaneously and GitHub provides control of builds in case of mis-deleting. Another reason we choose to upload our document to the cloud is that typically the well-known cloud drive will have good maintenance work. In this case, we normally won’t lose any important document if we store them on the server. In our project, documentation will be created throughout the entire project development lifecycle. And we should write our documents in a iterate way, which means that we should update our documents after getting some feedback during the whole development process. For instance, we are keeping changing the description of a programming code when we add new parts of it or if someone in the group gives feedback in case of not readable documents. Finally, we should treat our documentation like requirements. We need to estimate the priority of our documents and deal with the most important one first. This priority may change during the development process and some of documents can even be removed, which is fully depends on the project’s demand.

References

[1] Karlquist, "Design and implementation of an automotive experimental platform for ADAS," KTH, Skolan för informations- och kommunikationsteknik (ICT), 2017.

[2] Bark *et al.*, "Embedded service for self-adaptive cars," KTH, Instutitionen för Maskinkonstruktion, 2017. [Online]. Available: https://kth.box.com/s/iaabvkok3fsqgpbgaq10ymgqnnf3a135

[3] Lilliesköld, Gingnell, Petterson, and Varawala, "Management of projects eh2760," KTH, Management of Technology Research Group, 2018.

[4] Chen and Derbyshire, "The embedded systems HK 2018 project descriptions," KTH, Instutitionen för Maskinkonstruktion, 2018.

A Time Plan

B Resource Plan

C Work Breakdown Structure

The WBS was done digitally on Google Drive. It was later summarized into Figure 1. To help identify what was needed to be done code-wise, a breakdown of the main components of the program was done in the form of a "pseudo-code-breakdown". In this breakdown the topmost function was written first while assuming that all lower functions were already defined. Then, these lower functions were recursively broken down in a similar manner until the complexity of the subfunctions had been reduced sufficiently. These lowest-level subfunctions could then added to the Scrum board.

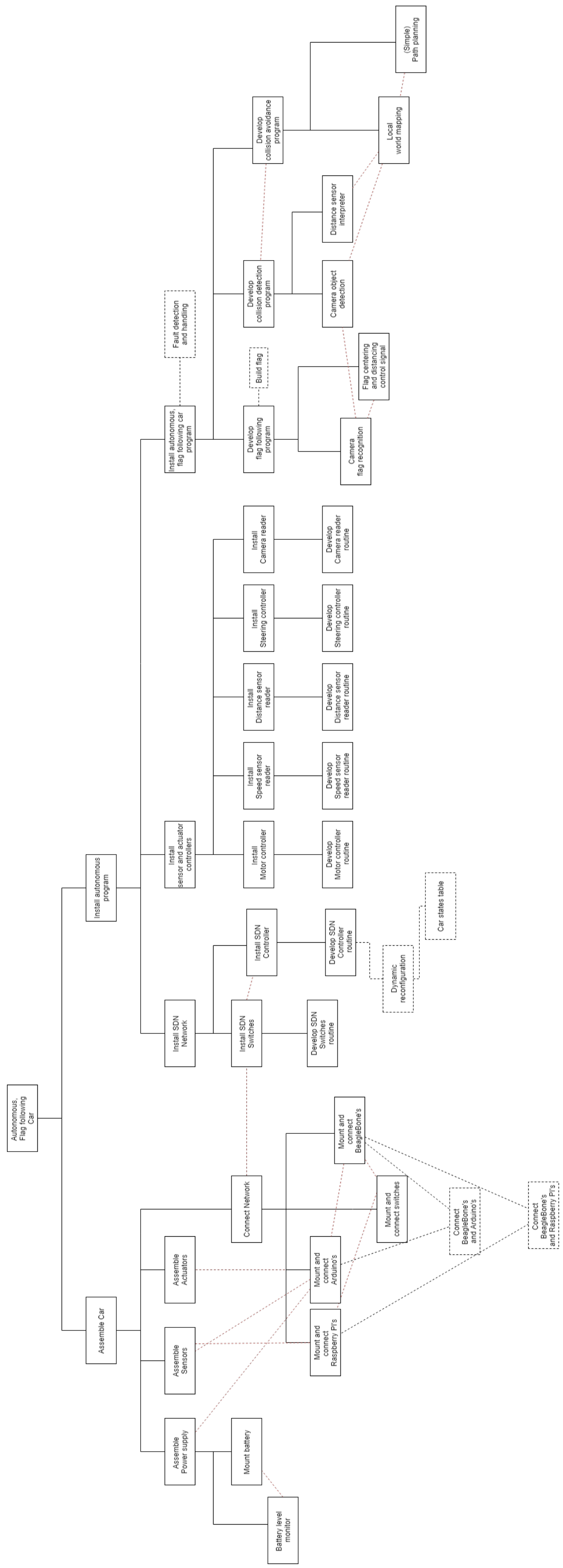

Figure 1: The Work Breakdown Structure for the autonomous car.

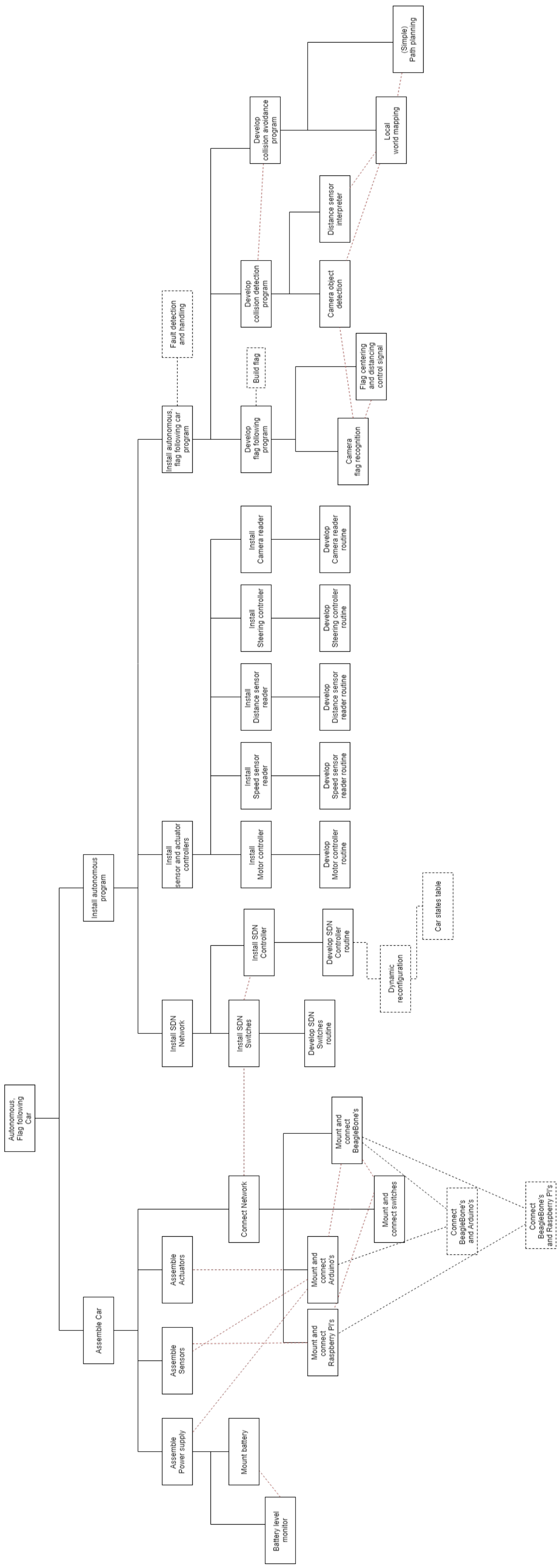

Figure 2: Pseudo-code from a brainstorming session to identify subtasks.