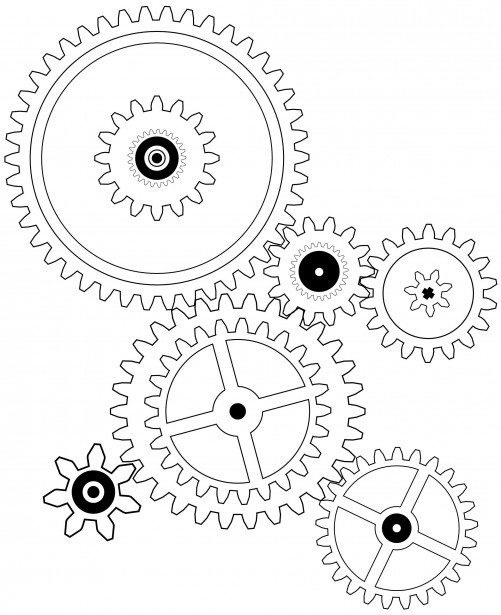
System Design

Highway Surfer

Status: Draft



Project period: 06-02-2017 to 01-07-2017

|  |  |
| --- | --- |
| Name | Student number |
| Lance Tijhuis | 564448 |
| Roy van Zuylen | 566455 |
| Nieck Busger op Vollenbroek | 558536 |
| Jordy Feiken | 546074 |
| Shreyas Patvardhan | 561179 |
| Bram Ribbers | 564818 |
| Lucas Smits | 567507 |
| Arthur Haverkort | 569277 |

Table of Contents

[1 Introduction 3](#_Toc475620466)

[2 General Overview and Design 4](#_Toc475620467)

[2.1 General Overview 4](#_Toc475620468)

[2.1.1 System Context 4](#_Toc475620469)

[2.1.2 Use Cases 4](#_Toc475620470)

[2.1.3 System Components 5](#_Toc475620471)

[2.2 Assumptions 6](#_Toc475620472)

[2.3 Constraints 6](#_Toc475620473)

[2.4 Risks 6](#_Toc475620474)

[3 Design Considerations 8](#_Toc475620475)

[3.1 Problem Statement 8](#_Toc475620476)

[3.2 Goals 8](#_Toc475620477)

[4 System Architecture and Design 9](#_Toc475620478)

[4.1 Logical View 9](#_Toc475620479)

[4.2 Hardware Architecture 10](#_Toc475620480)

[4.3 Software Architecture 12](#_Toc475620481)

[4.4 Internal communication Architecture 13](#_Toc475620482)

[Appendix A – References 14](#_Toc475620483)

[Appendix B – Diagram Conventions 15](#_Toc475620484)

# Introduction

In this document, you will find how the demo will be designed and developed. The hardware, software and the internal communication will be explained in the System Architecture and Design.

The idea of the demo is to let a car detect obstacles on the road and avoid them. To make this demo within the given requirements a small conveyor belt, with three different lanes marked on it, will be used. A small car will be attached to a linear actuator. The actuator will make the car switch lanes when an obstacle is detected. These obstacles can be fixed or be randomly placed on the belt and will be detected by ultrasonic sensor(s) placed above the car.

The purpose of this demo is for the HAN Automotive Research department to promote their software-tools (HANtune & HANcoder) which will be used to code and monitor the demo.

# General Overview and Design

## General Overview

A car runs on a conveyor belt and can run on three different lanes. When an obstacle is detected in the path of the car, the car will switch a lane to avoid a collision. It is possible to block all three lanes, but then the car will stop.

### System Context

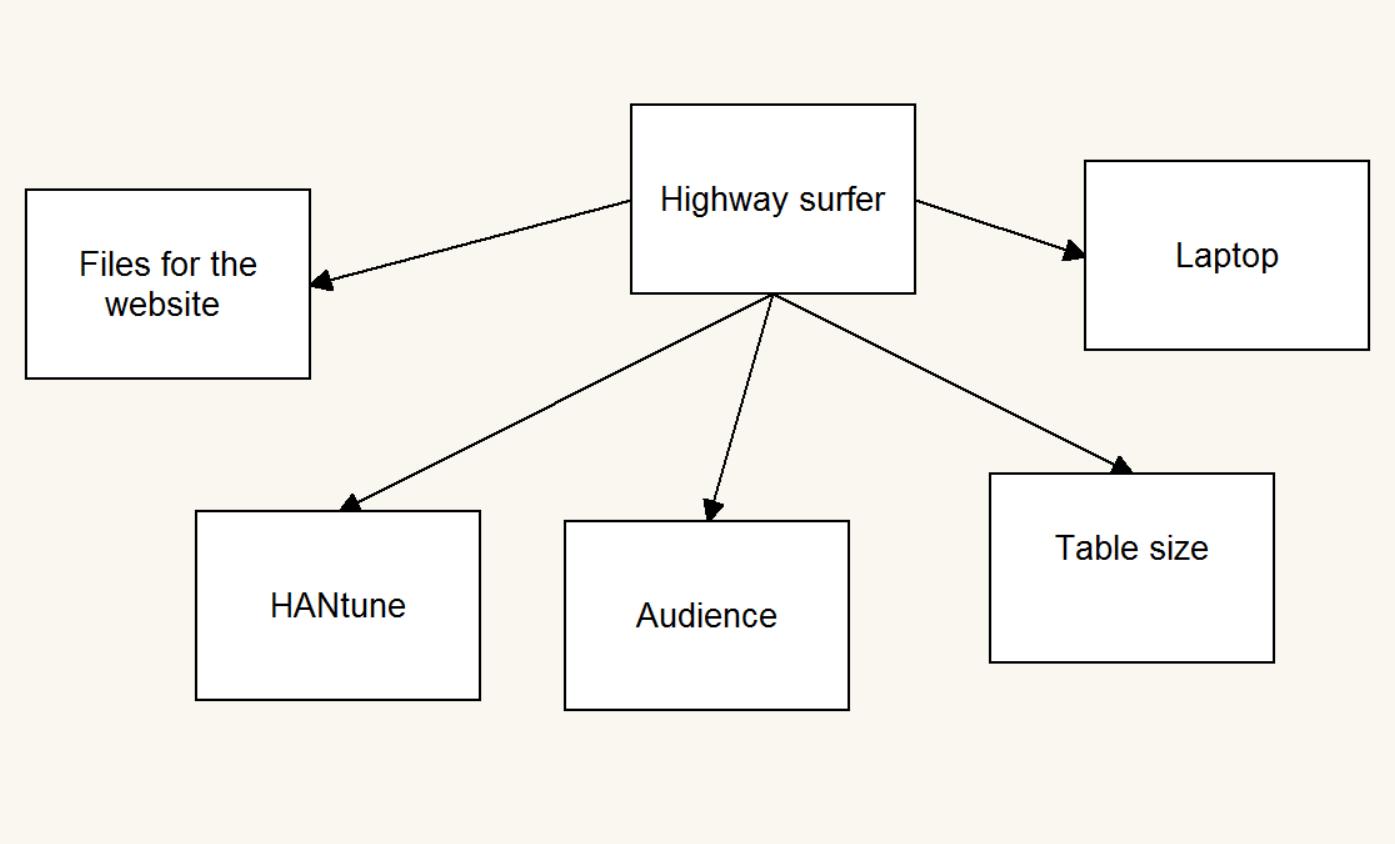


Figure : Contextual Diagram

### Use Cases

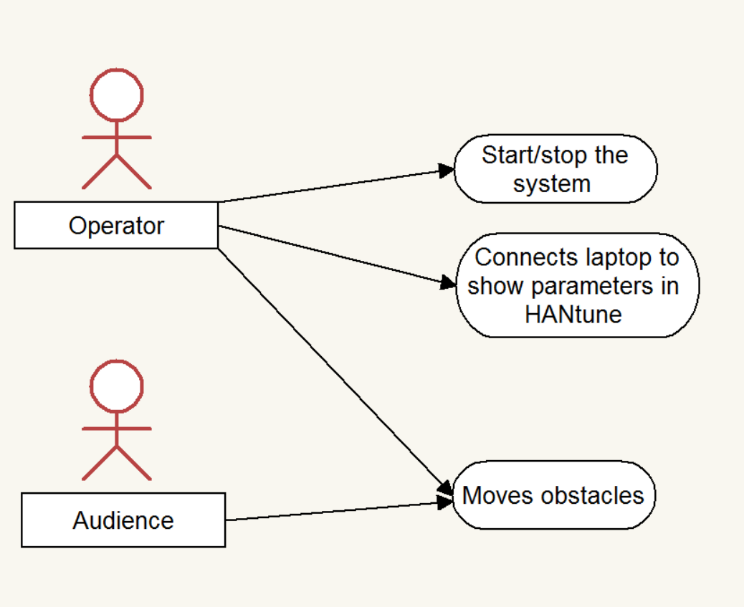


Figure : Use Case Diagram

### System Components

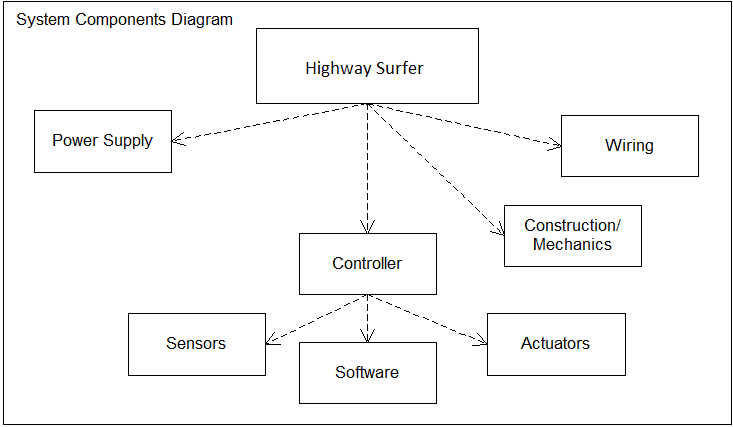


Figure : Component Hierarchy

## Assumptions

* The components required are ordered and delivered in time.
* The tools to write the software of the demo are always accessible.
* Everybody is on time.
* Everybody has the access to Microsoft Office, SolidWorks and MatLab Simulink.
* Every week there is a room that can be used as a workspace.

## Constraints

* The programming needs to be done with Simulink using the HANcoder blockset.
* The development board must be compatible with HANcoder.
* HANtune must be able to visualize all sensor output in physical values.
* HANtune must be able to visualize all actuator input in physical values.
* The demo interacts with the public.
* The product must not interfere with the Dutch constitution.
* The demonstrator must comply to IP20: (2: Protection against touching with a finger. Protects against penetration of objects larger than 12,5 mm; 0: No waterproofing).
* The demonstrator must have an easy to reach emergency button.
* The emergency button must be reachable from all around the demo
* The demonstrator must be at a total standstill within 1000ms.
* An FMEA should be performed to find and indicate hazards.
* The costs induced by the project must not exceed €300, -
* The project has to be finished before the 7th Wednesday of the regular lesson weeks of the 4th period.

## Risks

**The possible risks for the project are:**

1. The result of the demo will not be achieved
2. The project’s budget will be exceeded
3. The project will be cancelled
4. The deadlines will not be achieved

In the risk analysis each possible risk and its risk of failure has been determined.

**Risk:** The result of the demo will not be achieved

**Reaction**: limit

**Effect:** high

In the project, milestones have been used. With these milestones the progress will be reflected and only if the desired result has been achieved the next phase will begin, because of this it isn’t possible to intervene in time if the result deviated from the scope. Also the project contains a demarcation to close any deviations.

**Risk of failing:** Even though this project contains a correct milestone and demarcation to limit the risk, the risk of failing is higher than any of the other risks. This is because the project team has no experience in the field they are operating in and the deadline is too tight to absorb multiple let-downs, timewise.

**Risk:** The project’s budget will be exceeded

**Reaction:** limit

**Effect:** average

The client is part of a relatively small organization so the budget available is limited. Electronic components can be expensive and almost all components are of interest so it is not easy to economize on the number of parts. To limit the chance of exceeding the budget it’s important to use or reuse as much of the electronic devices that the ARLA has left.

**Risk of failing:** The risk of failing is not very high because there is a lot of equipment available from the ARLA. As long the available equipment is being used as much as possible the chances of exceeding the budget are small and If the budget is being exceeded the effect is still pretty slim.

**Risk:** The project is being cancelled by the client

**Reaction:** accepting

**Effect:** low

In this project, the effect of the project getting cancelled is really small. The client is a part of a big educational organization who can absorb the responsibilities the client had, so the project can successfully continue. This is a project where the HAN and the project implementers have mutual interest in succeeding the project.

**Risk of failing:** The chance of the project being cancelled is very small plus the effects are minimum. From all of the risks this one is probably the least important to note.

**Risk:** The deadlines will not be achieved

**Reaction:** limit

**Effect:** high

For each project there is a risk to anticipating the planning. The planning has a Gantt Chart so there is clarity over what is to be worked on. The Gantt Chart also has small margins to cover a few letdowns. This is how this risk is limited.

**Risk of failing**: The deadlines may not be reached if multiple letdowns occur. The schedule is pretty tight but with efficient working the deadlines should be achieved.

# Design Considerations

## Problem Statement

The HAN-AR has asked us to make a sensor-actuator demo. They would like to present our product at conferences and open days of the school. Our product needs to be made with HANtune and HANcoder. If our product is not usable with HANcoder and HANtune it will not get a go. It should be simple enough to be re-made by people with interest in HANtune and HANcoder.

## Goals

There are three main goals of this project. The first one is to provide students a project, where they can learn new skill. The students will learn about working together as a group and how a project like this is done with multiple people. Also, the students are learning new things about the electronics involved in this project.

The second goal of this project is to provide a new demonstration model for the HAN automotive research company. The demo model will be shown to people to peek their interest in what the company does. The demo model will show the people what the tools, developed by the HAN-AR, can do. This way the company and their tools get more attention, for example, at a tech convention, and gain more publicity.

The third goal is to provide good documentation on the website of the HAN-AR. On the website there is an option to download the tools, developed by HAN-AR, for the general public. This project will deliver a set of packages with all the documents needed to start building your own project. This is to provide new users a view on what is possible with the HAN tools, they can directly build their own device by following the documents.

# System Architecture and Design

*This chapter describes the different system aspects, only on a generic level.*

## Logical View



Figure State machine



Figure 5 Sensor-actuator

## Hardware Architecture

Figure Component diagram

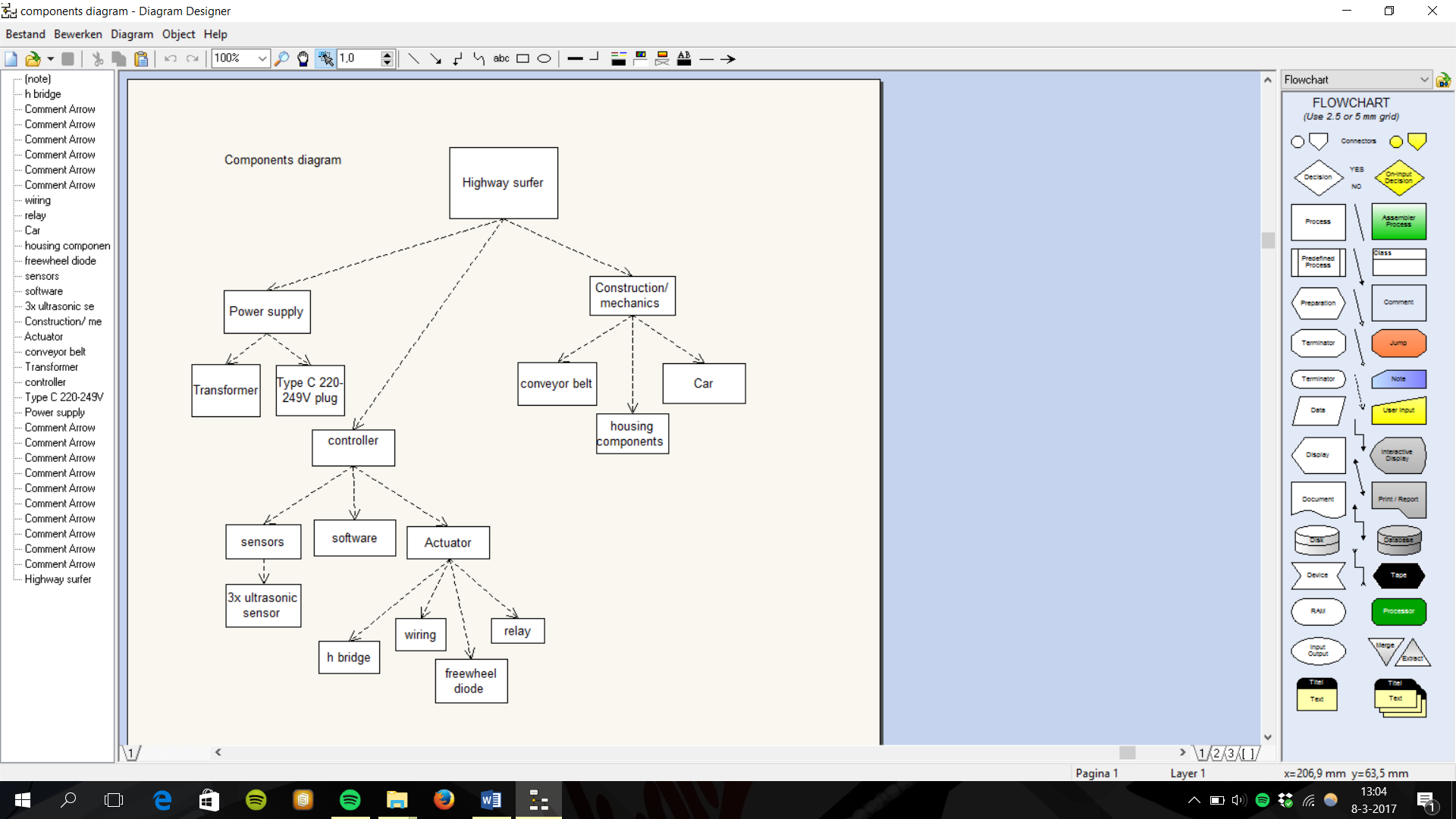
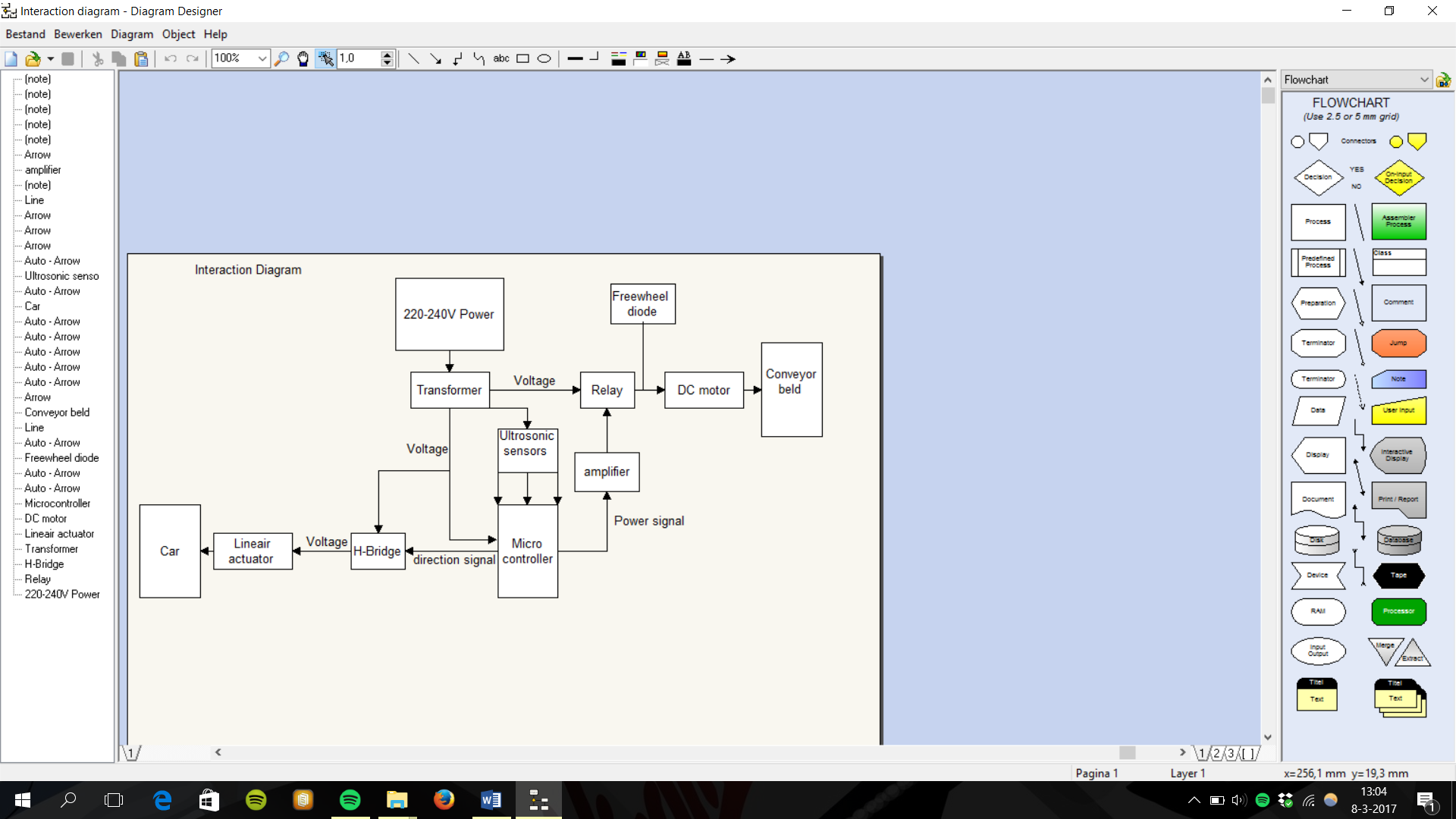


Figure Interaction diagram



*Explanation chosen components*

**Linear actuator:** Linear actuators can produce quick movement but little movement. In our demo the required force is low and the fast movement necessary. This is also the cheapest alternative.

**H bridge:** The H bridge is highly capable in this situation because it’s affordable, can work with signals what are relatively strong and negative.

**Amplifier:** The power output of any microcontroller is too low to power a relay. This makes an amplifier required.

**Relay:** The DC engine needs to be cut off in case the system recognizes an impossible situation. The cheapest and easiest way to do this is to set the engine apart from the microcontroller and connect it with a relay

## Software Architecture

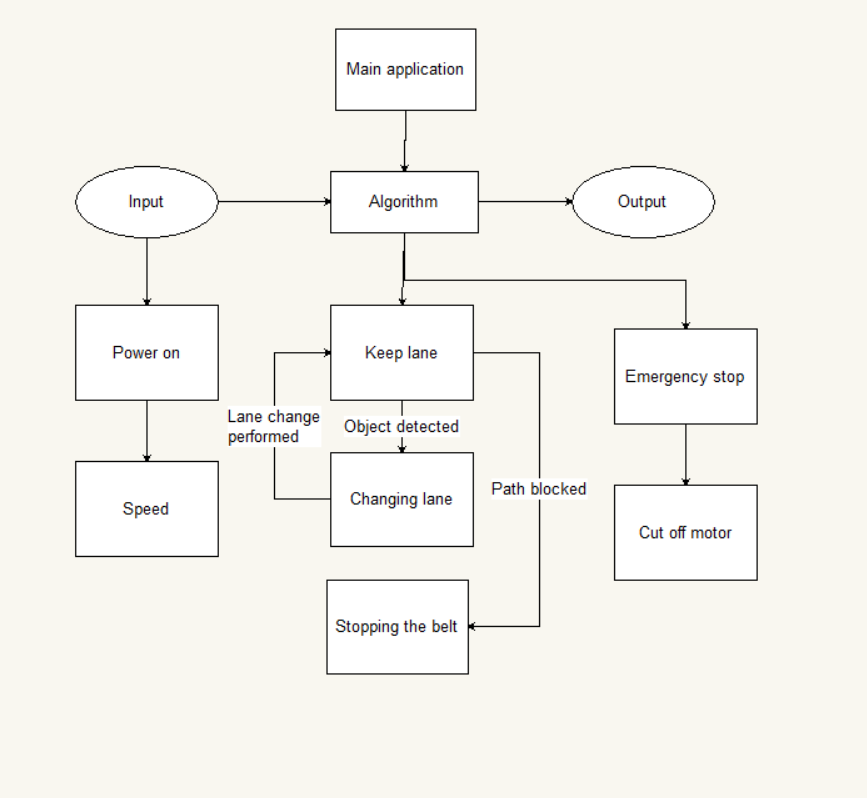
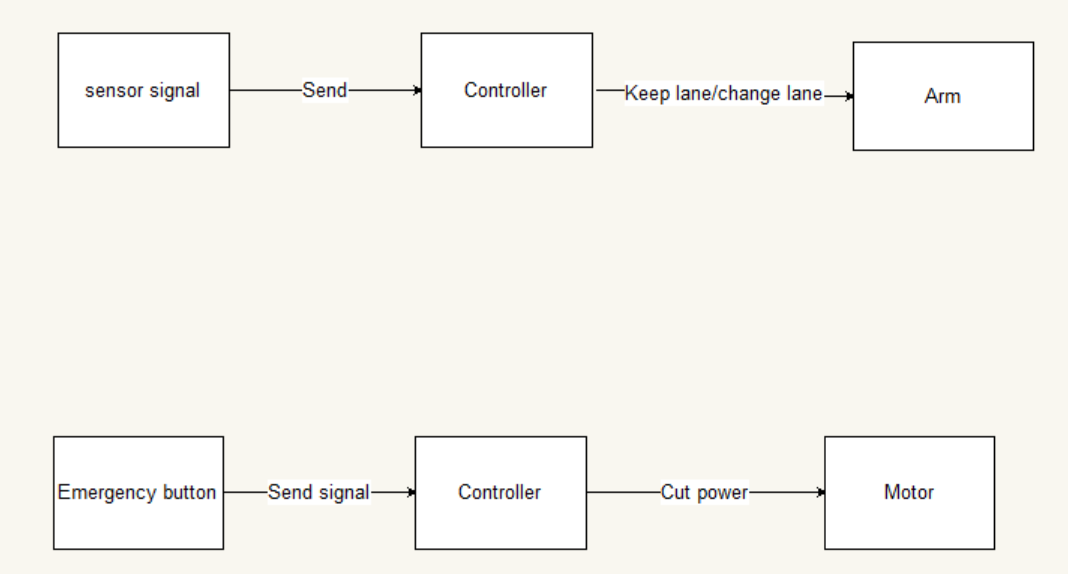


Figure Software component diagram

Figure 9 Software Interaction diagram



## Internal communication Architecture

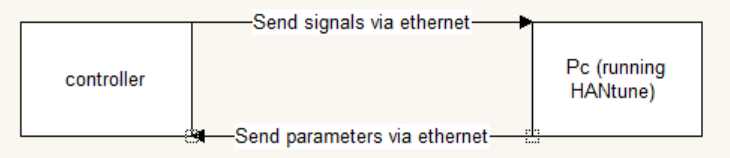


Figure 10 Interaction Diagram - HANtune Connection via Ethernet

# Appendix A – References

**Diagrams:**

Figure 1: Contextual Diagram 4

Figure 2: Use Case Diagram 4

Figure 3: Component Hierarchy 5

Figure 4: State machine 9

Figure 5: Sensor-actuator 9

Figure 6: Component diagram 10

Figure 7: Interaction diagram 10

Figure 8: Software component diagram 12

# Appendix B – Diagram Conventions

*These relations are a suggestion on how to use the diagrams. Feel free to add new features to your diagrams. However, if new features need to be added, please look at existing conventions in order to avoid confusion. At HAN-AR some knowledge and experience is present regarding formal modelling languages.*

**Component Diagram:**

|  |  |
| --- | --- |
|  | Component (Also used in Interaction Diagrams) |
|  | Component / Subcomponent relation (Also called: Parent/Child, or Super/Sub component) |

**Interaction Diagram:**

|  |  |
| --- | --- |
|  | Component (from component diagram) |
|  | Item flow (Anything that flows, e.g. Data, Energy, Fluid, Air) |

**State Machine Diagram:**

|  |  |
| --- | --- |
|  | Entry point |
|  | State |
|  | Exit point |
|  | State transition |
|  | Guard, entry or exit criterium |

**Use Case Diagram:**

|  |  |
| --- | --- |
|  | (Usually) Human interacting entity. Interaction may be both ways. Usually entails e.g. Operator, End-User, Maintenance staff, Audience, etc. |
|  | A use case. Could also have descendants in order to group use cases. |
|  | Interaction relation |
|  | Descending relation |