Robust and Adaptive Motion Control for Windscreen Wiping on Commercial Vehicles

PETER FJELLANDER
Robust and Adaptive Motion Control for Windscreen Wiping on Commercial Vehicles

PETER FJELLANDER

Master’s Thesis at ITM
Supervisor: Martin Edin Grimheden
Examiner: Martin Törngren

TRITA-ITM-EX 2018:162
Abstract

Windscreen wiping is an important part of driving safety and vehicle maneuverability. However, importance does not automatically imply progression, and the wiping functionality for heavy commercial vehicles have remained roughly the same through decades. When redesigning the cab for the latest truck generation at Scania, the thickness of the firewall was reduced to save weight. This reduced the stiffness of the cab, which made the vibrations in the throttle pedal from actuating the windscreen wiper rise to a critical level.

The problem definition in this thesis was to understand the root-cause and cooperation in the system by doing modelling and Model-Based Design (MBD), rather than starting with experimental verification. The task was to investigate what changes needed to be made in the controlling of the wiper motor and system specification of the ECU to reduce vibrations and ensure Scania's position as a premium brand in the future.

The windscreen wiping system was modelled in Simulink, with both Simulink blocks and Simscape models. A current-measuring voltage-controller for motion-profiles was developed and verified on real production hardware. Recommendation for future development of next ECU generation regarding sampling time and controller design was made and the importance of considering the whole system design was emphasized.

Results showed that controlling with current measurement of DC-motors as input parameter is a volatile approach due to disturbances. The algorithms depending on this measurement needs to be very robust, since filtering adds unwanted delay to the control loop.

Further investigations should be made in the component selection when mapping motors with the correct driver. The more logic placed in the motor, the less need for a complex ECU and vice versa.

Keywords: motion control, current measurement, windscreen wiping, dc motor, pwm, model-based design
Sammanfattning

För att kunna framföra ett fordon på ett säkert sätt är vindrutetorkning en viktig del. Men, bara för att det är en viktig del i användandet innebär det inte att det är en viktig del i utvecklingen. Detta har visat sig genom att funktionen och designen av vindrutetorkare på lastbilar har varit densamma i årtionden. När hytten till Scanias senaste lastbilsmodell designades så minskades tjockleken på torpedväggen för att spara vikt. Detta minskade även styvheten i hytten, vilket fick de vibrationer som inducerades vid körning av vindrutetorkarna att nå en kritisk gräns.

Problemställningen för detta exjobb var därför att förstå ursprunget till dessa vibrationer och hur delsystemen interagerar med varandra genom att utföra modellbaserad utveckling (MBD). Uppgiften var att undersöka vilka ändringar som behövde genomföras i styrningen av vindrutetorkarna och systemspecifikationen för den inbyggda styrenheten för att reducera vibrationerna och säkerställa Scanias position som premiummärke även i framtiden.

Vindrutetorkarsystemet modellerades i Simulink, med både Simulink-block och Simscape-modeller. En strömberoende spänningskontroller för rörelsestyrning utvecklades för att sedan verifieras på nuvarande hårdvara. Rekommendationer för framtida arbete på ECU gällande systemfrekvens för mätning samt algoritmdesign gjordes, samt helhetstänken vid design av ett nytt system poängterades.

Resultaten visar att styrning av en likströmsmotor med ström som ingångsparameter är komplicerat då strömmen varierar kraftigt på grund av störningar. Algoritmen som behandlar mätdatat måste därför vara väldigt robust eftersom filtrering påverkar systemet genom att lägga till fas i kontrollern, vilket ger eftersläpningar.

Kommande arbetsinsatser bör fokusera på hur man väljer komponenter som matchar varandra gällande likströmsmotor och ECU. Desto mer logik som placeras i motorn, desto mindre datorkraft behövs i den inbyggda styrenheten.

Nyckelord: rörelsestyrning, strömmätning, vindrutetorkning, dc-motor, pwm, modellbaserad utveckling
Acknowledgements

I would like to thank my academical supervisor Martin Grimheden for guidance during the project. I would also like to thank Scania and my industrial supervisor Ortwin Schlüter for giving feedback on the project. Thanks also goes to Anders Sundström for proofreading and giving second opinions.

Secondly, I would like to thank dSPACE GmbH and Petter Bellander at Fengco for providing the thesis with software to fully utilize the experimental setup.

Lastly, I would like to thank my colleagues at Scania. An extra thanks is directed to Tuhin Chowdhury and Pär Magnussson for their contribution beyond expectation in the definition phase of the thesis, and to Steffen Langhammer for help with elevation of privileges beyond expectation.
Contents

1 Introduction
   1.1 Background and problem description ................................. 1
   1.2 Purpose and definitions ............................................. 3
      1.2.1 Improving system characteristics ............................. 3
   1.3 Scope and definitions ............................................... 3
      1.3.1 Problem identification ......................................... 4
   1.4 Methodology .......................................................... 4
   1.5 Ethical considerations .............................................. 5

2 Windscreen wiping today .................................................. 7
   2.1 System architecture .................................................. 7
   2.2 Mechanical linkage ................................................... 8
   2.3 Principles of actuation ............................................... 9
   2.4 Drive unit and current control ..................................... 10
      2.4.1 Existing drive unit ........................................... 10
      2.4.2 Controllable parameters ...................................... 10
      2.4.3 Current-dependent algorithm ................................ 11
   2.5 About patents/CPC .................................................. 12

3 Modelling and parameters ................................................ 15
   3.1 Mechanical linkage .................................................. 15
      3.1.1 External load - naive approach ................................ 17
      3.1.2 External load - simple approach ................................ 19
   3.2 Electrical motor model ............................................. 20
      3.2.1 Park sensor ................................................... 20
      3.2.2 Design considerations ........................................ 21
   3.3 ECU ................................................................. 22
   3.4 Model verification .................................................. 22
   3.5 Current-sensing .................................................... 22
      3.5.1 Controller strategy ........................................... 22
   3.6 Summarizing of modelling and results ............................. 25

4 Experiments and verification ............................................. 27
# List of Figures

1.1 Windscreen wiper motor acceleration, with starting acceleration impulse .......... 1
1.2 Overview of physical system .................................................. 2
1.3 Proposed mechanical solution for damping throttle pedal vibration ....... 3

2.1 CAD model of four-bar linkage and wiper motor .............................. 8
2.2 Crank-rocker characteristic four-bar linkage. \( a + f < b + g \) .............. 8
2.3 System overview ................................................................. 10

3.1 System modelling overview ...................................................... 15
3.2 Linkage in Mechanics Explorer with labels. Joints in capital ................. 15
3.3 Mechanical linkage model ........................................................ 16
3.4 Naive load-estimation .............................................................. 17
3.5 Physical quantities of passenger side (PS) ...................................... 18
3.6 Modelled vs. measured motor torque ........................................... 18
3.7 Simple load-estimation .............................................................. 19
3.8 Windscreen wiper motor model .................................................... 20
3.9 Park sensor model ................................................................. 21
3.10 Current-controlled feed-forward model ........................................ 23
3.11 Voltage profile state machine ...................................................... 24
3.12 Voltage trajectory profile according to state machine in figure 3.11 ....... 24

4.1 Experimental setup ................................................................. 27
4.2 H-bridge block diagram ............................................................. 28
4.3 Evaluation motors ................................................................. 29
4.4 Real world simulator ............................................................... 29
4.5 dSPACE MicroAutoBox II .......................................................... 30
4.6 Circuit diagram for test-setup ..................................................... 30
4.7 MicroAutoBox II controller model. dSPACE-blocks coloured in cyan .. 31
4.8 Inverted step response at different PWM-frequencies ....................... 32
4.9 System performance. \( T_s = 1 \text{ ms} \) ............................................ 33
4.10 System performance with disturbances. \( T_s = 1 \text{ ms} \) ..................... 33
4.11 System performance. \( T_s = 2 \text{ ms} \) ............................................ 34
4.12 System performance with disturbances. \( T_s = 2 \text{ ms} \) ..................... 34
4.13 System performance. \( T_s = 10 \text{ ms} \) ........................................ 35
4.14 System performance with disturbances. $T_s = 10\text{ ms}$ .......................... 35
4.15 System performance. $T_s = 20\text{ ms}$ ................................................. 36
4.16 System performance with disturbances. $T_s = 20\text{ ms}$ .......................... 36
4.17 System performance. $T_s = 50\text{ ms}$ ................................................. 37
4.18 System performance with disturbances. $T_s = 50\text{ ms}$ .......................... 37
List of Tables

2.1 Patent tags B06S .................................................. 12
3.1 Linkage attributes .................................................. 16
3.2 Load parameters .................................................... 19
3.3 Worm gear attributes .............................................. 20
3.4 Motor attributes .................................................... 21
3.5 State machine parameters ........................................ 23
4.1 System performance evaluation ................................. 38
Glossary

**µC** microcontroller. 10, 20, 22, 27

**BDC** brushed DC-motor. 2, 9, 42

**BLDC** brush-less DC-motor. 9, 42

**CAD** computer-aided design. 16

**CAN** Controller Area Network. 10, 44

**CMOS** Complementary Metal Oxide Semiconductor. 10

**CPC** Cooperative Patent Classification. 12

**DS** driver side. 2

**ECU** electronic control unit. 2, 3, 7, 10, 11, 22, 23, 43, 44

**EMC** electro-magnetic compatibility. 11, 44

**EMI** electro-magnetic interference. 20

**HMI** human-machine interface. 7, 10

**MABX2** MicroAutoBox II. 30, 31

**MBD** model-based design. 7

**MOSFET** metal oxide semiconductor field effect transistor. 28

**NTG** New Truck Generation. 1

**PS** passenger side. 2

**PSU** power supply unit. 27

**PWM** pulse width modulation. 21, 23, 32
**RMS** root mean square. 22

**Scania** Scania CV AB. 1–4, 10, 11, 21, 42, 44

**SPI** Serial Peripheral Interface bus. 10

**SPST** single pole, single throw. 10

**VS-ABC** Variable Structure Adaptive Backstepping Controller. 11, 41, 43

**VS-MRAC** Variable Structure Model Reference Adaptive Control. 11, 41, 43

**WWM** windscreen wiper motor. 1–3, 29, 41, 43, 44
Chapter 1

Introduction

1.1 Background and problem description

In August 2016, Scania CV AB (Scania) released a new line of trucks, developed as the New Truck Generation (NTG) and marketed as Next Generation Scania. The windscreen wiper system design on the NTG is simple and has proven to be very reliable over the years, but nevertheless has some disadvantages, such as induced vibrations in the cab, which are coming more and more into focus.

The vibrations appeared shortly after introducing the new truck line to the public, so early on Scania realized that the problem had to be taken seriously. Therefore, an investigation was started to find the source of the vibrations. The root-cause of the vibrations was the actuation of the windscreen wiper motor (WWM), but a redesign of the firewall had created a weaker mount for the WWM, from which the vibrations propagated easier. Therefore, the vibration amplitude in the throttle pedal reached critical levels as shown in figure 1.1. [1]

![Windscreen wiper motor acceleration, with starting acceleration impulse](image)

Figure 1.1. Windscreen wiper motor acceleration, with starting acceleration impulse
Since the WWM is turning both driver side (DS) and passenger side (PS) wipers through a mechanical linkage, the load on the motor axle is a function of the lever angle $\Phi$ as seen in figure 1.2.

Since the brushed DC-motor (BDC) model is described as

$$U = R \cdot i + L \cdot \frac{di}{dt} + K_{emf} \cdot \dot{\phi}_r$$

(1.1)

and

$$J_r \cdot \ddot{\phi}_r = K_e \cdot i - T_{load} - d_r \cdot \dot{\phi}_r$$

(1.2)

a variable load ($T_{load}$) results in a variable current ($i$) fed into the motor since all other variables are motor-dependent and constant.

Due to the current and torque being linearly proportional in a DC-motor [2], the torque varies with alternating current. If the difference elapses within a short period of time, a shock is transferred to the system.

The windscreen wipers at Scania current truck generation can be activated at four different modes: high, low, fixed interval and controlled by rain sensor. However, only two different wiping speeds are used: high and low. The speeds are controlled by an electronic control unit (ECU) in an open-loop configuration.

Early on in this thesis, three methods for reducing the torque peaks were proposed:

1. Introducing a motor with integrated encoder
2. Using existing motor with sensor-less control
3. Make use of existing sensors and analyse the system to implement a controller

In consultation with both Scania and KTH, the first method was deemed unwanted, since the time to market for a possible solution was too long. The second
1.2 PURPOSE AND DEFINITIONS

method was academically promising, but was removed due to time constraints and for being too specific. Only the last method deemed sufficient for a thesis work and within the set time constraints.

1.2 Purpose and definitions

Before the start of the thesis work, the vibration problem has been thoroughly investigated mechanically. The current solution is to replace the bolt which transfers the force with a cylindrical vibration damper as seen in figure 1.3.

![Figure 1.3. Proposed mechanical solution for damping throttle pedal vibration](image)

However, both Scania and the company producing the vibration damper doubts the damper fulfills Scania’s tough premium-brand requirements to the same extent as the original bolt. Scania also wants something more easily implemented in the production.

In total, the goal was to optimize the motion profile of the WWM to avoid non-favourable combinations of force and levers.

1.2.1 Improving system characteristics

An overall goal for the thesis was to reduce the vibrations noticed by the driver. Since the truck generation where the problem is present is in production, no hardware modifications may be done. However, for those who experience the vibrations in the throttle pedal as a disturbing factor, there is always the possibility for after-market solution, i.e. service. Therefore, the result of this thesis aimed to be of such a magnitude that it is possible and cost effective to implement as an after-market solution by updating ECU software.

1.3 Scope and definitions

Since the research question was almost unlimited in question of possible solutions, some limitations had to be made. The thesis has focused on modelling and integrating systems of different disciplines into each other. Experiments has been done when necessary for verification, and implementation of everything into a production-ready
container-format is omitted due to time-constraints. More specific, the thesis has focused on current measuring on existing hardware and how to estimate rotor position with a limited amount of information. A control algorithm has been made and evaluated in terms of adaptability and robustness. Real-world test on production-made trucks was kept to a minimum because of lead time on hardware modifications. Instead, open-bench tests of hardware were favoured. Actuating was only made on the low-speed coil on the motor due to time-constraints. Any modifications of the system and the new solution was going to be compared to requirements imposed by Scania, but the intention is to keep the function of the system intact.

1.3.1 Problem identification

By concatenating the above described scope into a problem description, the following three points were deducted:

1. Understand the vibration origin
2. Propose new solution for vibration reduction
3. Verify proposed new solution

The first point aimed to answer if it was physically possible to reduce vibrations, or if the system design made it practically impossible with the desired system performance unaltered. To solve this problem, a descriptive study was made. Secondly, a new solution for windscreen wiping was to be developed. By theoretically assuming the torque is directly proportional to the current, a solution based on the measurability of current and reduction of the torque peaks was developed. Concurrently, a State-of-the-Art analysis in the field of current measurement was conducted. Lastly, an experimental setup to verify the implied gain from the developed solution was created.

1.4 Methodology

From an engineering point of view, the thesis goal was to improve the driver comfort. This by reducing the vibrations in the throttle pedal in terms of an experimental study with descriptive methods. To reduce the vibrations and get an academic thesis outcome, the main key was to understand the system and how the subsystems interact with each other. By doing theoretical analysis and modelling of mechatronic components, the foundation for a solution was made. By doing experimental verification the theories were verified. By understanding the interaction occurring in the system, the root cause was isolated and a proper solution was proposed.
1.5 Ethical considerations

In theory, the outcome of this thesis proposes a solution which eliminates a disturbance factor for the truck driver. This leads to more driving-environment aware drivers, but since the problem appears to quite few drivers, the overall outcome for society is small. However, every accident is adding unnecessary pain to society, and if this thesis contributes to reducing the risk of accidents for this small percentage of truck drivers, it does good.

Also, this thesis did include experiments on live hardware and was therefore required to taking human safety into account. All safety precautions and regulations were fulfilled; i.e. a sufficiently large safety distance was kept when operating windscreen wiper arms.

The largest academical ethical consideration was to make sure the thesis work was developed independent from bias from supervisors, stakeholders et al. Associated with this was also the consideration to be detailed in the thesis work, but respecting the intellectual property rights of involved companies.
Chapter 2

Windscreen wiping today

Regarding the given information in previous chapter, a frame of reference needs to be defined. The information gathered in this process is to be held as the literature study of this thesis.

2.1 System architecture

The windscreen wiping system on a manned vehicle usually contains five subsystems: human-machine interface (HMI), control unit, actuator, power transmitter, and wipers. The constraints of the system are usually set by external factors which needs to be weighted by the system architects. Such performance constraints may exist of, but not be limited to, performance - safety, manufacturability - volume, and performance - cost. [3]

Because the wiping system is a mechatronic system, a holistic approach in modelling is required. Mechatronic systems are complex by nature due to the sheer number of unique components and input parameters. Therefore, modelling the complete system requires the different subsystem models to be as simple as possible. One way of doing this is to use standardized continuous components for physical units and omit complex transient characteristics. [4]

During the last decades, the windscreen wiping system has stayed roughly the same, except for one component: the control unit. Prior to the control unit, the actuator was hard-wired to the HMI and possible to control only from the designed HMI interface. With the introduction of the ECU in the 1990s, the possibilities for large functional improvements and integrated functions emerged. [5] However, the wider toolbox also brought an almost infinite set of system states which were non-desirable and harder to model and predict. This set of non-desirable states created the need for model-based design (MBD). Model-based design is a way of capturing the dynamics of a system at a more holistic view early in the design process. By splitting the design into several unique models, the designer may easily re-use previously used models, which reduces time to market and increases quality. [6]
2.2 Mechanical linkage

A well-known invention of controlling motion is the mechanical linkage.

![CAD model of four-bar linkage and wiper motor](image1)

**Figure 2.1.** CAD model of four-bar linkage and wiper motor

Most common in the vehicle industry nowadays is to use a four-bar linkage with crank-rocker characteristics as seen in figure 2.1. The crank rotates a complete revolution while the rocker oscillates a fixed set of degrees for every crank revolution. The length of the couplers is adjusted so the desired motion characteristics are achieved.

![Crank-rocker characteristic four-bar linkage](image2)

**Figure 2.2.** Crank-rocker characteristic four-bar linkage. $a + f < b + g$

As classical Newtonian motion mechanics proposes, the most efficient setup is when the arm and lever are perpendicular. This is not the general case of the four-bar linkage. As shown in figure 2.2, the continuous motion of the crank results in different lever angles on the rocker. Hence, the internal damping and friction of the linkage is important for the system characteristics in terms of force distribution and
2.3 PRINCIPLES OF ACTUATION

thresholds. In a ideal world, free from friction and damping, the motion of each point in the linkage would be perfectly synchronized, i.e. one unit of motion of the crank gives the geometrical scaled one unit of motion at the rocker.

2.3 Principles of actuation

There are many types of actuators available for mechatronic systems. Commonly, they are referred to as electromagnetic actuators, fluid power actuators, piezoelectric actuators, thermal shape memory alloy actuators, and other actuators. Each type of actuator can be classified by different performance to simplify design. Volumetric power as a function of frequency/efficiency is the intuitive approach, but only after solid mechanic requirements are fulfilled, i.e. actuation stress as a function of actuation strain. [7] There are five types of criterion valid when designing a function with an actuator [8]:

1. Actuator type
2. Physical material properties parameters
3. Physical volumetric parameters
4. Input quantity
5. External load

When designing an actuator with maximum degree of freedom, the task is in practice impossible. Some limitations need to be set, especially since some of the design parameters are dependable. For example, all actuator types have ranges for linear scaling of power.

Common actuators used for commercial vehicles are electrical direct current (DC) motors of both brushed and brush-less type, and hydraulic/pneumatic motors. The BDC is cheap, simple, and easy to control. The maximum power is somewhat limited since the heat dissipation is conducted in the rotor. A brush-less DC-motor (BLDC) however, is expensive, needs a driver unit for correct commutation and is complex to control. Therefore the use cases for a BLDC motor are somewhat limited. However, since the heat dissipates through the stator, the power limit for a BLDC is significantly higher than the power limit for a BDC motor.

Hydraulic and pneumatic motors are the in principle the same, with only difference in hydraulic motors having a return path for the oil where the pneumatic actuators just release the used air into the premises. Since gas is a worse medium for kinetic energy than oil, the efficiency of pneumatic motors is worse than for hydraulic ones. However, the use of fluid power actuators is the most practical where high torque is needed and the infrastructure allows an additional component in the loop.
2.4 Drive unit and current control

2.4.1 Existing drive unit

The windscreen wipers at Scania’s current truck generation can be activated at four different modes: high, low, fixed interval and controlled by rain sensor. However, only two different wiping speeds are used: high and low. The speeds are controlled by an ECU in an open-loop configuration.

![System overview](image)

Due to legislation, the ECU interacts with HMI directly through I/O pins as shown in figure 2.3, and on a functional level with the rest of the truck through the Controller Area Network (CAN). [9]

Switching component

The switch used as peripheral switch in the ECU is an automotive switch from NXP Semiconductors. It is an integrated Complementary Metal Oxide Semiconductor (CMOS) chip for controlling loads in the industrial and automotive industry. Due to the legislation demands, the switch has a fall-back fail-safe mode. The switch communicates with the microcontroller (µC) through a Serial Peripheral Interface bus (SPI) bus and accommodates per channel measurement of open load, over current and short circuit.

A single pole, single throw (SPST)-switch is possible to model with a h-bridge. The h-bridge emulates a SPST-switch by using only one-quadrant due to the fact that the polarity is fixed on a switch and it is only high-sided switch characteristics. [2]

2.4.2 Controllable parameters

There are always constraints when designing a product aimed to be produced in tens of thousands of units. The current ECU is switching and measuring current, which in theory makes it capable of controlling both current and voltage. However, the
2.4. DRIVE UNIT AND CURRENT CONTROL

ECU:s currently in use by Scania is considered not good enough for high frequency operation because of the bandwidth of the I/O. Due to electro-magnetic compatibility (EMC), every I/O pin is equipped with a low pass filter when applicable, i.e. DC-characteristics usage. The cross-over frequency for the filter is of a magnitude of 50 Hz. This design choice also improves performance by reducing high frequency noise, such as switch bounce. [10]

Since the ECU have on-board flash, history may be utilised for each measured parameter. Both voltage and current history are utilized by trajectory algorithms.

2.4.3 Current-dependent algorithm

When controlling current in a motor, the desire is often to control a specific motion. When controlling a specific motion, a trajectory control is usually the desired option. When controlling actuators by current, the desire is often to remove jerk and smoothen the motion. However, most industrial systems are open-loop (without feedback), which may reduce controllability. A motion trajectory controller is robust to open-loop errors since the trajectory is a known variable. [11]

One of the most used algorithms is the sliding mode current control algorithm. Historically, this control technique has suffered from switching imperfections and sliding phenomena, but modern silicon hardware has eliminated previous problems.

A general system which is to be controlled by a sliding-mode controller could be described by

$$\frac{dy}{dx} = f(x) + B(x) \cdot u$$  \hspace{1cm} (2.1)

where $u$ is the input-vector, $x$ the system state and $f$ is the system state as a function of time. By constructing a switching function $S(x) = [S_1, S_2, ..., S_m]$ where the surface created is $S_i(x) = 0$, the surface created is called the sliding surface. The sliding function $S(x)$ could be implemented in both continuous and discrete time. [12, 13]

Controlling current by using sliding mode current controller is a robust way of controlling since it rejects high frequency noise. Robustness is per se a good property, since controlling actuators with unstable algorithms imposes unnecessary risks. However, to much robustness reduces adaptability, as Rohrs, Valavani, Athans and Stein proposed in the 80s. [14] Combining sliding mode current controls with adaptive schemes to improve adaptiveness is common when constructing modern controllers. Adaptive schemes are weak when handling transients, but for a process with a relative order of one in S-domain, a different method was approached called Variable Structure Model Reference Adaptive Control (VS-MRAC). [15, 16]

Queiroz, Araujo and Dias proposes that the VS-MRAC controller is inferior to the Variable Structure Adaptive Backstepping Controller (VS-ABC) due to the need of integral parts and complex controller design, which makes implementation cumbersome. [17]

Since robustness is defined as an ability to withstand changes [18], the characteristics of a robust current controller should take the shape of a low-pass filter
to smoothen the current profile. As shown previously, the derivate of the current should be low to minimize the impulses to the system, which requires the algorithm to limit the current in some cases.

2.5 About patents/CPC

The Cooperative Patent Classification (CPC) is a joint project between the European Patent Office (EPO) and US Patent Office (USPTO) to harmonize the intellectual property process, ensure compliance with international patent classification system and reduce the need for cross-verification of patents. [19]

From the European Patent office [20] the corresponding CPC tags relevant to this thesis were identified and a search for full-text patents was done. From the full-text search a number of patents was filtered out by reading the abstract, another assertion was done, and thereafter the remaining patents was read in full. Following CPC scheme was researched:

<table>
<thead>
<tr>
<th>Patent tags B06S</th>
</tr>
</thead>
<tbody>
<tr>
<td>B06S 1/00 Cleaning of vehicles</td>
</tr>
<tr>
<td>B06S 1/02 Cleaning windscreens, windows or optical devices</td>
</tr>
<tr>
<td>B06S 1/04 Wipers or the like, e.g. scrapers</td>
</tr>
<tr>
<td>B06S 1/043 {Attachment of the wiper assembly to the vehicle}</td>
</tr>
<tr>
<td>B06S 1/0441 {characterised by the attachment means}</td>
</tr>
<tr>
<td>B06S 1/0444 {comprising vibration or noise absorbing means}</td>
</tr>
<tr>
<td>B06S 1/06 characterised by the drive</td>
</tr>
<tr>
<td>B06S 1/08 electrically driven</td>
</tr>
<tr>
<td>B06S 1/0814 {using several drive motors; motor synchronisation circuits}</td>
</tr>
<tr>
<td>B06S 1/16 Means for transmitting drive</td>
</tr>
<tr>
<td>B06S 1/166 {characterised by the combination of a motor-reduction unit and a mechanism for converting rotary into oscillatory movement}</td>
</tr>
</tbody>
</table>

As table 2.1 shows, several different problems are present in the construction of a windscreen wiping system. The squealing friction-noise from the screen has been addressed by Karcher whom proposes a method of motion control of the wiper arm to eliminate squealing noise. [21]

Wiping of special or curved windows is also a topic where several different solutions has been invented. Garrastacho and Hussaini proposes a design differentiation to overcome the problem of uneven curvature of windscreen. [22] Luik and miller proposes an improved spring-loaded solution for cleaning highly curvature wind-screens. [23] Bichler solves the problem of operating windscreen wiper on an openable window. [24] Closely related to the curvature is the spherical wiping where
Google has developed a system for passively wipe their roof-mounted LiDAR:s in a 360-degree angle. [25]

After concluding the window figure is of no relevance, the actuation force is of interest. Tatsuya et al. has developed an alternative way of mounting the four-bar linkage for improved performance [26] Tetsuya and Takamichi proposes a design where the wiper motor is positioned in between the linkage. [27] Ikeda has developed a solution to utilize a high-performance brush-less motor for driving the linkage. [28] Wegner et al. has invented a method of mounting the wiper arm directly on the motor, eliminating the need for a linkage, but requiring two motors. [29] Högner and Wagner invented a method for programming two separate wipers where one is configured as master and the rest as slaves. [30] Prskawetz et al. continues the work on separate wiper arms by developing an algorithm for self-determination of master/slave. [31]

Lastly, the control of the wiper arms have been investigated. Hospital and Jackson proposes a method for adaptive adjusting of a software end-stop to provide self-adaptation to the wipers. [32] Boland proposed an improved yoke-free wiper blade where the end-caps are properly fastened to the wiper. [33] IBM has developed a method to prevent melted snow on a windscreen to refreeze during shut-down phases of heating windscreens. [34] Braun et al. proposes a method for determining system properties regarding the load. [35] Ernst and Kraemer has developed a wind blocking add-on for the wiper arm. [36]
Chapter 3

Modelling and parameters

The system is mainly modelled in Simulink to keep the intuitive connection to the real world hardware. [37] By keeping this connection, the model works with parameters such as position, angular velocity and acceleration instead of the mathematical Laplace transforms of each physical quantity. [38]

![Figure 3.1. System modelling overview](image)

A complete Matlab and Simulink model representing figure 3.1 can be found in appendix A, B and C.

3.1 Mechanical linkage

As stated in section 2.2, a mechanical linkage is simple, on a conceptual level. The model in Matlab Mechanics Explorer is built as shown in figure 3.2

![Figure 3.2. Linkage in Mechanics Explorer with labels. Joints in capital.](image)

In Simulink, the mechanical domain is used to model the linkage, hence the mechanical solver, world frame and solver in the bottom of figure 3.3. The model is derived from the Simulink example *smdoc_four_bar*. [39]
CHAPTER 3. MODELLING AND PARAMETERS

Figure 3.3. Mechanical linkage model

The world frame is attached to the three fixed points of a dual four-bar linkage with co-joined centre point: $B_{PS}$, $B_{DS}$ and $A$. Each of those fixed points is attached to a block representing a computer-aided design (CAD) description in Mechanics Explorer in Matlab. Then the model shows that the joints are connected by levers to form the linkage.

Table 3.1. Linkage attributes

<table>
<thead>
<tr>
<th>Signal</th>
<th>Type</th>
<th>Value</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>damping stiffness</td>
<td>attribute</td>
<td>$1.5 \times 10^{-2}$</td>
<td>N m/deg</td>
<td>Joint damping</td>
</tr>
<tr>
<td>density</td>
<td>attribute</td>
<td>$1 \times 10^{-9}$</td>
<td>N m/deg</td>
<td>Joint stiffness</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8000</td>
<td>kg/m³</td>
<td>Bar density</td>
</tr>
</tbody>
</table>
3.1. MECHANICAL LINKAGE

Material parameters deducted from the model of the linkage is described in table 3.1. Those parameters were the ones that gave the most successful system response when verifying against measured data as in figure 3.6.

3.1.1 External load - naive approach

Good enough is a mantra when modelling system loads. Since the motion of the system is static (except when ice on screen [40]) and the forces from the environment are dynamic, the resulting outcome of system load is hard to predict with methods available for a common student. A good model may be achievable, but to analytically determine such model is a relevant subject for a stand-alone thesis work.

Model

As an introduction before dynamic modelling, a semi-static load model was tested as shown in figure 3.4. The approach was to always apply a torque corresponding to the air-force at highway speeds. Since the motion of the wipers makes the lever the most favourable in the middle of the rocker motion, a sinusoidal modification to the torque was applied.

Results

As the results show, the load profile is working according to instructions. The overall characteristics are promising, however, the frequency is not correct and a mysterious peak appears at every turning-point as shown in figure 3.5 and 3.6.

The aim of this modelling approach was that the modelled motor torque in red in figure 3.6 should behave the same as the measured motor torque in blue. E.g. the torque-curve should behave similar, which is not the case.
CHAPTER 3. MODELLING AND PARAMETERS

Physical Quantities, Passenger side

Figure 3.5. Physical quantities of passenger side (PS)

Motor torque

Figure 3.6. Modelled vs. measured motor torque
3.1. MECHANICAL LINKAGE

3.1.2 External load - simple approach

As the naive approach for modelling the load torque resulted in unexplainable errors too big to be ignored, a simplification of the model was done. The Karnop friction was removed and the model was made less dependent on the position differences. The resulting model can be seen in figure 3.7.

![Figure 3.7. Simple load-estimation](image)

where input and output is accordingly to table 3.2.

<table>
<thead>
<tr>
<th>Signal</th>
<th>Type</th>
<th>Value</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ext_load</td>
<td>input</td>
<td>3</td>
<td>Nm</td>
<td>External load factor</td>
</tr>
<tr>
<td>vel</td>
<td>input</td>
<td>N/A</td>
<td>rad/s</td>
<td>Joint velocity</td>
</tr>
<tr>
<td>m_pos</td>
<td>input</td>
<td>N/A</td>
<td>rad</td>
<td>Crank angle</td>
</tr>
<tr>
<td>Load</td>
<td>output</td>
<td>N/A</td>
<td>Nm</td>
<td>Joint load</td>
</tr>
</tbody>
</table>

The load on the joints can be mathematically described as:

\[
load = (1 - |0.7 \cdot \cos(m_{pos})|) \cdot sgn(vel) \cdot ext_{load}
\]  

(3.1)
3.2 Electrical motor model

The electrical motor model is a fundamental part of the model, since the real-world usability of the results depends on the accuracy of the model.

The DC-motor itself is low-pass filtered by input- and output inductances as shown in figure 3.8. This is to increase the robustness against electro-magnetic interference (EMI). [9] The DC motor internally connected to a viscously loss-less, right-handed worm gear with parameters according to table 3.3.

<table>
<thead>
<tr>
<th>Description</th>
<th>Type</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gearing</td>
<td>attribute</td>
<td>73</td>
<td>-</td>
</tr>
<tr>
<td>Worm-gear efficiency</td>
<td>parameter</td>
<td>0.74</td>
<td>-</td>
</tr>
<tr>
<td>Gear-worm efficiency</td>
<td>parameter</td>
<td>0.65</td>
<td>-</td>
</tr>
<tr>
<td>Power threshold</td>
<td>attribute</td>
<td>0.001</td>
<td>W</td>
</tr>
</tbody>
</table>

The rest of the mechanical motor model is the ideal rotational motion sensor which feeds the rotor angle to the park sensor, ideal velocity feedback from the load on the output shaft, and ideal output torque for actuation of output shaft.

3.2.1 Park sensor

The park sensor on the motor is a sheet metal located on the output gear, which indicates parking position. Approximately 2 degrees every revolution the sheet metal is grounded, and therefore sending a pulse to the $\mu$C to brake the motor. For the sake of simplicity, the starting position of the wiper motor is also when it is in contact with parking sheet metal. Where to put the parking mode could be a topic
3.2. ELECTRICAL MOTOR MODEL

of interest to this thesis, but it has shown previously that parking position is not a critical cause for vibrations in the throttle pedal. [41]

![Figure 3.9. Park sensor model](image)

As seen in figure 3.9, the output of the parking sensor is binary in the model, but may be any value when implemented on the motor since it is referenced to chassis ground.

3.2.2 Design considerations

The physical and electrical design of the motor is decided by the supplier, since Scania buys the complete windscreen-wiping subsystem as a single product. [40] Therefore, the motor parameters available for modelling are somewhat uncertain since no data sheet exists.

<table>
<thead>
<tr>
<th>Description</th>
<th>Type</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Armature resistance</td>
<td>attribute</td>
<td>1.91</td>
<td>Ω</td>
</tr>
<tr>
<td>Armature inductance</td>
<td>attribute</td>
<td>3.75</td>
<td>mH</td>
</tr>
<tr>
<td>Filter inductance (each)</td>
<td>attribute</td>
<td>5.7</td>
<td>µH</td>
</tr>
<tr>
<td>Back-emf constant</td>
<td>attribute</td>
<td>0.113</td>
<td>Vs/rad</td>
</tr>
<tr>
<td>No-load current</td>
<td>input</td>
<td>0.58</td>
<td>A</td>
</tr>
<tr>
<td>DC supply voltage</td>
<td>input</td>
<td>28</td>
<td>V</td>
</tr>
</tbody>
</table>

As shown in table 3.4, the armature inductance is high. This is a design constraint, because the electrical time constant is

$$\tau_e = \frac{L}{R} = \frac{3.75 \text{ mH}}{1.91 \text{ Ω}} \approx 1.96\text{ ms} \quad (3.2)$$

which gives a theoretical maximum bandwidth of

$$f_{3db} = \frac{1}{2\pi\tau_e} = \frac{1}{2\pi \cdot 1.96\text{ ms}} \approx 81.1\text{ Hz} \quad (3.3)$$

A minimum pulse width modulation (PWM) frequency of

$$f_{PWM} = \frac{1}{\tau_e} \approx 510\text{ Hz} \quad (3.4)$$

is required to theoretically ensure smoothness in the actuation of the wiper motor. For sure, such a low PWM frequency would emit audible noise, which reduces the over-all performance of the system.
3.3 ECU

The ECU is represented by a h-bridge driven in one-quadrant mode, i.e. simulation of the currently implemented switch. This part is possible to make more dynamic to use the flexibility of the h-bridge, the only thing to adjust is to make the reference voltage a controlled parameter instead of static.

3.4 Model verification

The common approach when modelling the system was verification by design of subsystems. This means each subsystem’s behaviour was verified against theory. However, when combining all subsystems with the mechanical linkage, the situation becomes more complex. Therefore, the simulated torque output is compared to motor data supplied by the manufacturer. However, this does not guarantee that each individual component performs exactly accordingly to the suppliers design specification. But since the overall system performance for each load is acceptable, the model is deemed sufficient.

3.5 Current-sensing

The key parameter to control the wiper motor is the current. The armature current is measured in the switch described in 2.4.1, which then is communicated to the µC. Three parts are identified as crucial in order to achieve desired performance: current limiter, turning point detection, and soft start.

To ensure smooth performance, the deviation from the root mean square (RMS) value should be kept at a minimum. Only in occasional cases should the physical peak current be allowed.

To detect the turning point of the wiper helps reducing shock transmission into the truck, since a custom motion profile may be applied to counter such behaviours.

Lastly, to ensure a steady transmission of an accelerating system into a steady-state system, a soft start may be introduced. An soft-start algorithm acts upon the difference between the output and reference and limits the maximum acceleration to a pre-determined accepted value.

3.5.1 Controller strategy

The current current-controller is based upon a voltage-trajectory state-machine, which determine the behaviour depending on the input variables as discussed by both Teixeira and Revol [12, 13].
3.5. CURRENT-SENSING

![Diagram of current-controlled feed-forward model](image)

**Figure 3.10.** Current-controlled feed-forward model

As figure 3.10 shows, the controller is modelled in high level language in both physical and mathematical domain of Simulink. The parameters of the state machine is according to table 3.5.

**Table 3.5. State machine parameters**

<table>
<thead>
<tr>
<th>Signal</th>
<th>Type</th>
<th>Value</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>park</td>
<td>input</td>
<td>N/A</td>
<td>N/A</td>
<td>Park signal</td>
</tr>
<tr>
<td>period</td>
<td>input</td>
<td>N/A</td>
<td>s</td>
<td>Time for half a period</td>
</tr>
<tr>
<td>i_real</td>
<td>input</td>
<td>N/A</td>
<td>A</td>
<td>Motor current</td>
</tr>
<tr>
<td>i_lim</td>
<td>input</td>
<td>N/A</td>
<td>A</td>
<td>Minimum motor current</td>
</tr>
<tr>
<td>Ts</td>
<td>input</td>
<td>0.1</td>
<td>s</td>
<td>Sampling time</td>
</tr>
<tr>
<td>enable</td>
<td>input</td>
<td>1</td>
<td>N/A</td>
<td>State machine enable</td>
</tr>
<tr>
<td>AccVal</td>
<td>input</td>
<td>0.7</td>
<td>N/A</td>
<td>Soft-start parameter</td>
</tr>
<tr>
<td>TurnVal</td>
<td>input</td>
<td>0.7</td>
<td>N/A</td>
<td>Soft-turn parameter</td>
</tr>
<tr>
<td>y</td>
<td>output</td>
<td>N/A</td>
<td>N/A</td>
<td>PWM output</td>
</tr>
</tbody>
</table>

The period subsystem takes time as input and calculated once every falling edge of the park signal. It takes the current time, subtracts the value from last calculation one period ago, and multiplies by a half. The switch in line with the signal decides that if the period is below 0.1 s (which is physically impossible with current hardware), the period is set to 1 s.

The current subsystem takes current and time as input and outputs real and minimum current. Both of the input are then transferred the discrete domain with Zero-Order-Hold with 20 ms sampling time, to simulate a real ECU. For convenience, the current is set to zero during the first 0.3 s of the simulation to avoid transient errors in the starting process. The current minimum is a sliding minimum value of the discretised current with a sliding period of 50 samples. Lastly, the sliding minimum is compared to the no-load current of the motor and replaced if less than or equal, and multiplied with the \( i_{lim \_gain} \) variable (3\%) to set the threshold for the minimum current detection.
As figure 3.11 shows, the state machine starts in an idle state, doing nothing. When switching the enable flag, the machine starts to run and moves into the state *Soft_start* where it ramps up during $Ts$ time to maximum system voltage.

Since $Ts$ is a magnitude smaller than the period, the machine continues to the *RUN* state automatically where it continues to run with maximum voltage according to 3.12. All this until it reaches the $Ts$ before period time, where it continues if the real current is lower than the threshold. After deceleration in the turning point, the system enters next state and accelerates into the *RUN* state where it continues until the prediction of a full revolution enters. Then it decelerates.
3.6. SUMMARIZING OF MODELLING AND RESULTS

If the enable flag is raised, the process continues. The voltage trajectory profile as shown in figure 3.12 lowers the armature voltage on the motor when approaching turning points. As stated in equation 1.1 and 1.2, the reduced voltage results in lowered current, and therefore lowered torque as mentioned in section 1.1. The trapezoidal voltage reference implemented has worse jerk-reduction than the s-curve, but is easier to implement. [42] If better performance is desirable, a better jerk-reduction profile may be implemented instead of the trapezoidal ramp.

3.6 Summarizing of modelling and results

In short, the most important understanding of chapter 3 was that the external load (i.e. wind load) on the system is negligible in comparison with the internal dynamics (damping and stiffness). This leads to the conclusion that the system is relatively well documented, as proposed in the first point in the problem description in section 1.3.1. If the external load had not been deemed negligible, tests had to be made on the magnitude and impact of different loads on the system. However, with this concluded, the experimental phase may be conducted.
Chapter 4

Experiments and verification

Focus on this chapter is to verify the correctness of previously discussed methods, models and results from chapter 2 and 3.

4.1 System overview

The lab assembly consists of a straight chain of hardware, with a computer monitoring the \( \mu \mathrm{C} \).

![Experimental setup](image)

**Figure 4.1.** Experimental setup

4.2 Hardware

4.2.1 Power supply and measurement tool

The power supply unit (PSU) used in the experiments is a variable supply unit with an upper limit of 30 V and 10 A. The output is set to 28 V ±0.02 V, measured with a Fluke 85 multimeter. [43] Even though the battery voltage is 24 V, the maximum system voltage is achieved when running the generator at maximum power, hence
CHAPTER 4. EXPERIMENTS AND VERIFICATION

the 28 V output. All instruments were regularly calibrated to ensure conformity to the specification.

4.2.2 Driver

To adjust the voltage, a h-bridge is needed. As this is not a core part of the thesis, a driver evaluation board was purchased. The board chosen was an evaluation board of (EV-VNH7100AS) the h-bridge VNH7100AS from ST Microelectronics. The power driver can handle and input voltage of 28 V and a current of 15 A with limited external cooling.

![Figure 4.2. H-bridge block diagram](image)

The h-bridge is symmetrically constructed with two identical sides for both outputs. As figure 4.2 shows, the internal bridge with four metal oxide semiconductor field effect transistor (MOSFET), are controlled through a combined input for all MOSFET:s.

An internal interlocking feature is present to avoid activating low- and high side at the same time, creating a short-circuit. [44]
4.2. HARDWARE

4.2.3 Motor

A wide selection of motors have been supplied for testing the hypothesis of the thesis. Pictured below are three generation of motors, with minor adjustments between each generation. Motor data is according to table 3.3 and 3.4.

![Figure 4.3. Evaluation motors](image)

The connection for the power supply to the motor is done through a TYCO HDSCS vehicle connector, highlighted by the red circle in figure 4.3. Each motor has dual coils for low- and high speed actuation. Only low-side actuation was done in this thesis as discussed in section 1.3.

4.2.4 Load and disturbance

As load simulator for the WWM, a lifetime-testing equipment is used. In short, it is a 3 mm aluminum plate which mounts the motor with the axle rotating in the vertical plane.

![Figure 4.4. Real world simulator](image)
As shown in figure 4.4, the system setup at $t = 0\, s$ is with arm rotated to $270^\circ$. This position allows the motor to initially accelerate with zero counter-reacting forces from gravity.

The disturbance force is produced by manually braking or accelerating the load during run-time.

### 4.2.5 Microcontroller

To combine all subsystem together while controlling and measuring data, an MicroAutoBox II (MABX2) from dSPACE was used. The MABX2 is equipped with all major automotive communication buses, high-performance I/O and the ability to monitor the run-time environment from a standard PC. [45]

![Figure 4.5. dSPACE MicroAutoBox II](image)

The system is connected according to the circuit diagram in figure 4.6, where all electrical connections (pull-up resistors etc.) are handled internally in each device.

![Figure 4.6. Circuit diagram for test-setup](image)
4.3 Test plan

To evaluate controller performance, the following test run scheme was used: Five sampling times for current measurement; 1, 2, 10, 20 and 50 ms. Each test run was approximately twenty-five seconds to ensure test length to approximately 10 samples for each setup. For each sampling time, two tests were run. One run with load only, and one run with a large disturbance torque added periodically. The disturbance torque was made by braking or acceleration the load manually. The code running on the MABX2 was C-code generated from a modified controller model in Matlab. To compensate for real-world imperfection, the model was based on proposed model in chapter 3.

![MicroAutoBox II controller model](image)

**Figure 4.7.** MicroAutoBox II controller model. dSPACE-blocks coloured in cyan.

The MABX2 was set to trigger on the positive flank of main start switch, and the recording was stopped manually after the set run-time. Due to large disturbances in the motor current as seen in figure 4.18, a moving average filter was implemented as shown in figure 4.7 where an approximate unit-delay of one eighth of the system frequency was introduced.

Since the h-bridge had integrated over-current protection, a relatively high starting voltage was needed to get the motor running without engaging the over-current protection. Therefore, the minimum motor voltage was set to 35% of the system voltage.
4.4 Results

4.4.1 PWM performance

Before any motor-current tests were conducted, an inverted step response with PWM was conducted at different sampling times, $T_s$.

![PWM performance graph](image)

**Figure 4.8.** Inverted step response at different PWM-frequencies

As shown in figure 4.8, a PWM-frequency of 100 kHz is enough to smoothly actuate the motor. This is accordingly to the rule of thumb, which says that the controlling parameter should be 10 to 30 times faster than the actuating one, as theory presented in section 3.2.2 predicts. Also, 100 kHz is not in the audible spectrum, which makes it proper for long-term usage in terms of driver comfort.

The suitability of PWM-frequency to smoothly actuate the motor is shown by the relatively mild derivative occurrence when applying the inverted step response as shown in green in figure 4.8. Both PWM-frequencies of 1 kHz and 0.01 kHz results in a great deviation from predicted current when comparing to the current controlled with a PWM-frequency of 100 kHz. In reality; this phenomena results in noise, vibrations and excessive heating of motor coil.

4.4.2 Data charts

The tests were conducted by driving the motor with the proposed voltage profile while measuring the motor current, park signal, and monitoring the voltage profile.
4.4. RESULTS

Figure 4.9. System performance. $T_s = 1\, \text{ms}$

Figure 4.10. System performance with disturbances. $T_s = 1\, \text{ms}$
CHAPTER 4. EXPERIMENTS AND VERIFICATION

System performance, normal load

Figure 4.11. System performance. $T_s = 2 \text{ ms}$

System performance, disturbed load

Figure 4.12. System performance with disturbances. $T_s = 2 \text{ ms}$
4.4. RESULTS

![System performance, normal load](image)

**Figure 4.13.** System performance. $T_s = 10$ ms

![System performance, disturbed load](image)

**Figure 4.14.** System performance with disturbances. $T_s = 10$ ms
**Figure 4.15.** System performance. $T_s = 20$ ms

**Figure 4.16.** System performance with disturbances. $T_s = 20$ ms
4.4. RESULTS

![System performance, normal load](image)

**Figure 4.17.** System performance. $T_s = 50$ ms

![System performance, disturbed load](image)

**Figure 4.18.** System performance with disturbances. $T_s = 50$ ms
4.5 Summarizing

To summarize figure 4.9 to 4.18, assessment of performance was done in terms of park prediction success and turn prediction success. The full data set of 20 s of data was used, even if only 10 s is shown in the graphs for clarity.

The assessment was done on the following criteria:

1. When a park signal is sent, the motor voltage should be in system state \textit{RUN.turn_point_dec} as shown in figure 3.12.

2. Between each park signal, on the same position on the current signal, the motor voltage should be in system state \textit{RUN.turn_point_acc} as shown in figure 3.12.

3. Only one occurrence of \textit{RUN.turn_point_acc}, \textit{RUN.turn_point_dec}, \textit{Soft_start} and \textit{Soft_stop} is allowed between each park signal.

Failure to comply to above mentioned points is counted as an error in table 4.1, where assessment of first criteria is grouped as park prediction and assessment of second criteria is grouped as turn prediction. As an example, the absence of an deceleration phase at $t = 5$ s in figure 4.15 counts as an error when assessing the algorithm.

As mentioned in section 1.3.1, a new solution for reduction of vibrations should be proposed. The current tested solution should be improving the driver comfort since the derivative of the current is reduced and functionality is intact.

<table>
<thead>
<tr>
<th>$T_s$</th>
<th>Load profile</th>
<th>Park prediction</th>
<th>Turn prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ms</td>
<td>Normal</td>
<td>5/10</td>
<td>6/10</td>
</tr>
<tr>
<td>1 ms</td>
<td>Disturbed</td>
<td>6/11</td>
<td>6/10</td>
</tr>
<tr>
<td>2 ms</td>
<td>Normal</td>
<td>5/10</td>
<td>6/10</td>
</tr>
<tr>
<td>2 ms</td>
<td>Disturbed</td>
<td>5/10</td>
<td>9/10</td>
</tr>
<tr>
<td>10 ms</td>
<td>Normal</td>
<td>7/10</td>
<td>6/10</td>
</tr>
<tr>
<td>10 ms</td>
<td>Disturbed</td>
<td>7/10</td>
<td>9/10</td>
</tr>
<tr>
<td>20 ms</td>
<td>Normal</td>
<td>9/9</td>
<td>6/10</td>
</tr>
<tr>
<td>20 ms</td>
<td>Disturbed</td>
<td>9/10</td>
<td>6/10</td>
</tr>
<tr>
<td>50 ms</td>
<td>Normal</td>
<td>9/9</td>
<td>9/10</td>
</tr>
<tr>
<td>50 ms</td>
<td>Disturbed</td>
<td>9/9</td>
<td>9/10</td>
</tr>
</tbody>
</table>

As table 4.1 shows, an increased sampling rate reduces the robustness of the implemented algorithm and the number of false negatives increases in terms of turn predictions. The first revolution is always counted as a fail by algorithm design, which is clearly visible in the charts. As one can see in each chart in chapter 4.4.2, and as described in figure 4.7, the current is filtered with 0.125 s to make graphs readable and algorithm more robust. This sample time was chosen because it is
4.5. SUMMARIZING

a magnitude faster than the system revolution time, but the most efficient filter
design is an optimization problem, and therefore not covered in this thesis. Worth
to mention is that the slowest sampling time is not performing correctly in figure
4.17 and 4.18. This due to the deceleration phases happens after the park signal
and not on it.
Chapter 5

Discussion and conclusions

As stated in the Problem identification chapter, the task of this thesis was to:

1. Understand the vibration origin
2. Propose a new solution for vibration reduction
3. Verify proposed new solution

The problem intended to solve with this thesis is how to control the motion of the wipers to reduce the torque peaks and vibrations. The usual engineering way is to develop a solution, do some tests and then iterate until the requirements have been fulfilled. This thesis aimed to develop a model of the system to solve the problem academically by proposing how it should be. When modelling the system, the number of parameters where the value was unknown were in the beginning quite many. However, during the modelling it turned out which parameters were critical to the system performance as discussed in chapter 3. By modelling the system in Simulink, an understanding of the system and the dependable parameters was made. As previously stated, the external load model was of minor importance since the internal damping was magnitudes greater. It is also to be mentioned that the goal was not to create a perfect model, but a sufficient good one to detect the turning-point of the wipers. By understanding the system, the preparations for making a theoretical controller was made, since understanding of the system and the controllable parameters are crucial for a correct design.

The controller was made based on knowledge gathered about sliding mode controllers in 2.4.3. Compared to the VS-MRAC and VS-ABC controllers, the sliding mode controller offered the greatest robustness, but lacked in adaptability. A more tuned controller design is an optimization problem, and not covered in this thesis, but since the tests conducted in section 4.4.2 show a distinct dependability to the sampling frequency, the problem is solvable. Comparing of the algorithms may be conducted as future work.

Since the thesis has focused on the single setup of dry wiping with cold motor, an inductive hypothesis about other weather conditions may be done if the WWM
can be interpreted as linear, which is not an obviously bad hypothesis. Connected with this hypothesis is the generality of the motor. Since the motor is a BDC-motor, assumptions made of system performance are valid when talking about electrical motors in general which can be driven in the same configuration as in section 4.1. More specific, the results may be achieved with a BLDC-motor if the driver block is correctly emulated by the BLDC-driver.

If the thesis had tried to investigate the torque-reducing problem by using sensorless control, i.e. by measuring the imperfections of commutation brushes in the rotor, the scope had been out of bound for a thesis work, and the main problem would have been in the engineering scope since measuring accurate with fast sampling frequency is difficult, even for experienced personnel at Scania.

In the tests conducted, the test setup described in section 4.1 is dependable on the gravity since the worm gear in the motor is not self-locking and the centre of mass is not positioned in the centre of motion. However, this was also deemed to be of minor importance since this an adjustment of controller parameters due to the same circular motion. This affected the results, as shown in figure 4.15 where there sometimes appears a straight line as motor current. This in contrast to the noise motor current. The challenge is to make the controller robust enough, without malfunctioning as it did with the 50 ms sampling time. To compare the results with the possibility to implement on current Scania hardware, the controller is only possible to implement with a sampling time of 50 ms, which were the only sampling time which were disregarded due to faulty behaviour. Last, by not least, measurement errors had to be taken into account. Since Scania is experienced in experimental testing, all instruments used were calibrated to meet the manufactured specifications, even if the hardware is dated.

As table 4.1 shows, the results are somewhat surprising. The slowest sampling time is disregarded due to not follow specifications of the controller. E.g. too much delay before actuating. However, the two fastest sampling times also predicted quite similar which imposes that the acquired low pass filter when running slow sampling frequencies is affecting the system in a desirable way and needs to be implemented if faster sampling times are to be used. Omitted from this thesis is the case of running the measuring and the computing processes at different frequencies, which may improve system and control stability, especially when dealing with filters. Also, the effect of aliasing needs to be taken into account and is affecting the motor current and the threshold values. E.g. when the ripple in figure 4.18 is around 0.5 A (20% of max current) and the threshold for state change is 3% of minimum current, the robustness of the controller is compromised by design.
Chapter 6

Recommendations and future Work

To combine sensor data to create a greater sum than sum of the individual inputs is not a breaking new discovery. One of the most interesting further developments of this thesis is to compare the method and algorithm with the implementation of a Kalman-filter.

Since the thesis shows that the performance of the controller is sufficient with the implementation of a voltage driver, the way forward may take three directions.

The first one, and probably the most expensive and flexible approach, is to implement two electronically geared encoder-equipped motors actuating each wiper independently. This has the side effect of getting rid of the linkage. However, since this is cost-driving, it may not be the best overall solution.

As previously discussed in section 1.1, a closed loop system (i.e. motor with encoder) would probably outperform charts shown in section 4.4.2, but instead the model error would have been introduced as a system parameter. As previously stated, a closed loop controlled motor with integrated encoder will significantly increase system cost. This makes this solution only feasible if the additional functionality is demanded by the market, since the customer buys a function (windscreen wipping), not a construction.

As the experimental results shows in section 4.5, higher sampling time does not always equals better performance per se, but requires a more delicate controller design. However, the current sampling time of 20 ms is shown to be the best trade-off in terms of performance and robustness as shown in table 4.1. However, as current solution is limited by hardware design and as a faster sampling time is desirable, a more complex filter solution may be implemented in the design of the next ECU and WWM, and good control may be achieved with limited resources. If the hardware is developed to be able to use with a higher frequency system, the current filter design may be further optimized by comparing the controller implemented in this thesis with VS-MRAC and VS-ABC controllers.

Lastly, in terms of design considerations, the number of end-point switches may be increased to approximately four to create a poor-man’s encoder, which in the current user case should be sufficient to improve the vibration handling in the
throttle pedal. This reduces the amount of work needed to implement the future, but is probably the most limited solution.

In this design consideration problem, the lead-time of developing a new truck should be taken into account. If any decision is taken today it will take several years before the solution is being implemented on the line of production and the customer demands may have changed. However, Scania is on a development trend of doing the truck more decentralized, which sort of confirms the next step of the system to be transformed to a distributed one which only communicates with the truck through the CAN-bus. This also helps with the EMC-certification, which is an area of interest where the solutions seldom are easy.

Since the power requirements of the WWM is roughly 72 W, the possibility of using an automotive motor constructed for car voltages should also be considered to utilize the power of purchasing from the Volkswagen Group and stream-lining the product portfolio.

The implemented controller uses three types of information: time, current and history of both. Those parameters are possible to use in any combination, but there are more information available which have not been used. For example, the rain sensor may be possible to utilize to define current thresholds, since rain lowers the external load from the friction on the windscreen. In the same way could the speedometer information be used to calculate wind load. Especially the history is interesting to highlight. How long duration should history be saved? With micro-controllers and flash-memory, the possibilities to store data are almost infinite, but should one store every data point since the first power-up of the ECU or only the data from the last wiping cycle? As the results in section 4.5 shows, the effect of a sudden disturbance (history) diminishes after approximately one motor revolution with developed controller. Since the environment also may change quite dramatically, i.e. when travelling from the coast to highlands, data may soon be outdated.

The experimental setup was simple, but there is a lot of data accessible on the CAN-buses in the truck. I.e. one could also use the GPS and date to predict the season, in addition to using the temperature- and rain sensor. Such approaches of sensor fusion was omitted from this thesis due to time constraints. The environmental parameter was set static to indoor conditions. However, this sensor comparison is probably enough for a stand-alone thesis.
Bibliography


BIBLIOGRAPHY


# Appendix A

## Matlab toolboxes

Installed toolboxes in Matlab 2017b.

### A.1 Mathworks-developed toolboxes

<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Date</th>
<th>Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aerospace Blockset</td>
<td>3.20</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Aerospace Toolbox</td>
<td>2.20</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Antenna Toolbox</td>
<td>3.0</td>
<td>26-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Audio System Toolbox</td>
<td>1.3</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Automated Driving System Toolbox</td>
<td>1.1</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Bioinformatics Toolbox</td>
<td>4.9</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Communications System Toolbox</td>
<td>6.5</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Computer Vision System Toolbox</td>
<td>8.0</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Control System Toolbox</td>
<td>10.3</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Curve Fitting Toolbox</td>
<td>3.5.6</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>DO Qualification Kit</td>
<td>3.4</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>DSP System Toolbox</td>
<td>9.5</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Data Acquisition Toolbox</td>
<td>3.12</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Database Toolbox</td>
<td>8.0</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Datafeed Toolbox</td>
<td>5.6</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Econometrics Toolbox</td>
<td>4.1</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Embedded Coder</td>
<td>6.13</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Filter Design HDL Coder</td>
<td>3.1.2</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Financial Instruments Toolbox</td>
<td>2.6</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Financial Toolbox</td>
<td>5.10</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Fixed-Point Designer</td>
<td>6.0</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Fuzzy Logic Toolbox</td>
<td>2.3</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>GPU Coder</td>
<td>1.0</td>
<td>27-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>Global Optimization Toolbox</td>
<td>3.4.3</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
<tr>
<td>HDL Coder</td>
<td>3.11</td>
<td>24-Jul-2017</td>
<td>(R2017b)</td>
</tr>
</tbody>
</table>
### APPENDIX A. MATLAB TOOLBOXES

<table>
<thead>
<tr>
<th>Toolbox Name</th>
<th>Version</th>
<th>Date</th>
<th>Release</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDL Verifier</td>
<td>5.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>IEC Certification Kit</td>
<td>3.10</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Image Acquisition Toolbox</td>
<td>5.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Image Processing Toolbox</td>
<td>10.1</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Instrument Control Toolbox</td>
<td>3.12</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>LTE HDL Toolbox</td>
<td>1.0</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>LTE System Toolbox</td>
<td>2.5</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>MATLAB</td>
<td>9.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>MATLAB Coder</td>
<td>3.4</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>MATLAB Compiler</td>
<td>6.5</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>MATLAB Compiler SDK</td>
<td>6.4</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>MATLAB Report Generator</td>
<td>5.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Mapping Toolbox</td>
<td>4.5.1</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Model Predictive Control Toolbox</td>
<td>6.0</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Model-Based Calibration Toolbox</td>
<td>5.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Neural Network Toolbox</td>
<td>11.0</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>OPC Toolbox</td>
<td>4.0.4</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Optimization Toolbox</td>
<td>8.0</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Parallel Computing Toolbox</td>
<td>6.11</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Partial Differential Equation Toolbox</td>
<td>2.5</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Phased Array System Toolbox</td>
<td>3.5</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Polyspace Bug Finder</td>
<td>2.4</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Polyspace Code Prover</td>
<td>9.8</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Powertrain Blockset</td>
<td>1.2</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>RF Blockset</td>
<td>6.1</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>RF Toolbox</td>
<td>3.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Risk Management Toolbox</td>
<td>1.2</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Robotics System Toolbox</td>
<td>1.5</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Robust Control Toolbox</td>
<td>6.4</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Signal Processing Toolbox</td>
<td>7.5</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>SimBiology</td>
<td>5.7</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>SimEvents</td>
<td>5.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simscape</td>
<td>4.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simscape Driveline</td>
<td>2.13</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simscape Electronics</td>
<td>2.12</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simscape Fluids</td>
<td>2.3</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simscape Multibody</td>
<td>5.1</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simscape Power Systems</td>
<td>6.8</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simulink</td>
<td>9.0</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simulink 3D Animation</td>
<td>7.8</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simulink Check</td>
<td>4.0</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simulink Code Inspector</td>
<td>3.1</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simulink Coder</td>
<td>8.13</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
<tr>
<td>Simulink Control Design</td>
<td>5.0</td>
<td>24-Jul-2017</td>
<td>R2017b</td>
</tr>
</tbody>
</table>
### A.2. dSPACE-developed Toolboxes

<table>
<thead>
<tr>
<th>Name</th>
<th>Version</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>dSPACE FlexRay Configuration Blockset</td>
<td>4.0</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE MATLAB Connection</td>
<td>2.4</td>
<td>01-Jun-2012</td>
</tr>
<tr>
<td>dSPACE MATLAB Integration Main</td>
<td>2.2</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE Model Comparison Tool</td>
<td>2.8</td>
<td>07-Nov-2017</td>
</tr>
<tr>
<td>dSPACE Model Port Block Library</td>
<td>3.5</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE Model Separation Block Library</td>
<td>3.5</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE MotionDesk Blockset</td>
<td>2.5.2</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI Bypass Blockset</td>
<td>3.9</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI CAN Blockset</td>
<td>3.4.5</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI CAN MultiMessage Blockset</td>
<td>4.6</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI EMC Blockset</td>
<td>1.4.1</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI ETHERNET Blockset</td>
<td>1.2.3</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI ETHERNETUDP Blockset</td>
<td>1.4.3</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI FPGA Programming Blockset</td>
<td>3.4</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI FlexRay Configuration Blockset</td>
<td>4.0</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI Gigalink Blockset</td>
<td>2.4</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI LIN MultiMessage Blockset</td>
<td>2.9</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI MultiMessage Shared Utilities</td>
<td>1.9.5</td>
<td>02-Nov-2017</td>
</tr>
</tbody>
</table>
### APPENDIX A. MATLAB TOOLBOXES

<table>
<thead>
<tr>
<th>Toolbox</th>
<th>Version</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>dSPACE RTI RPCU Blockset</td>
<td>2.2.3</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI USBFLIGHTREC Blockset</td>
<td>1.2.2</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI WATCHDOG Blockset</td>
<td>2.1.1</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI XCP on Ethernet Blockset</td>
<td>1.2.10</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE RTI for Multiprocessor Systems RTI-MP</td>
<td>7.9</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE Real-Time Interface (RTI1401)</td>
<td>7.9</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE Real-Time Interface Shared Utilities</td>
<td>7.9</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE Run-Time Target</td>
<td>3.5</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE Run-Time Target Shared Utilities</td>
<td>3.4</td>
<td>02-Nov-2017</td>
</tr>
<tr>
<td>dSPACE TargetLink Data Dictionary</td>
<td>4.3</td>
<td>26-Oct-2017</td>
</tr>
<tr>
<td>dSPACE TargetLink Production Code Generator</td>
<td>4.3</td>
<td>26-Oct-2017</td>
</tr>
</tbody>
</table>
Appendix B

Matlab code

Complete Matlab code used for thesis work.

B.1 Matlab scripts

B.1.1 Matlab main code

1  addpath (’/functions’);
2  addpath (’/scripts’);
3  addpath (’C:\Users\PFJVFB\Desktop’);
4
5  cleanup (; %Wipes workspace
6  run real_world.m; %Import coordinates from Catia
7  run four_bar_calc.m; %Calculate 4–bar linkage constraints
8
9  %Run–time settings
10  plott = 0;
11  motordata_exists = 1;
12
13  run wiper_calc.m; %Calculate wiper points
14  if (plott == 1)
15      analytic_plot (A, A1, A2, B1, B2, B_DS, b_DS, B_PS, b_PS, C_DS,
16                      C_PS, D, a, space, points, resolution) %Plot analytical
17            version
18  end
19
20  if (motordata_exists == 0)
21      extract_motor();
22  else
23      run motor.m; %Handles motor data
24  end
APPENDIX B. MATLAB CODE

```matlab
run simulink_para.m %Initiate Simulink parameters
%
% Print modelled response
run custom_plot.m
desc = [];
plot(PS_B.Time,PS_B.Data(:,1)); desc = [desc; 'Position PS '];
plot(PS_B.Time,PS_B.Data(:,2)); desc = [desc; 'Velocity PS '];
plot(PS_B.Time,PS_B.Data(:,4)); desc = [desc; 'Torque PS '];
legend(desc);
title('Physical Quantities, Passenger side')
run custom_plot.m
desc = [];
plot(DGCML.t,filter(b,a,DGCML.T));
plot(MA.Time,MA.Data(:,4));
plot(PS_B.Time,sin(PS_B.Data(:,1))*30);
plot(MA.Time,sin(MA.Data(:,1))*30);
legend('Measured data, Dry glass, cold motor, low speed ','Modelled motor torque','Wiper position','Motor position');
title('Motor torque and position')

B.1.2 Real world measurements

%Physical dimensions of actual Scania wiper system
%
%Drawing 2371908_1
PT_A = [1457.8 -913.16 1496.54];
PT_B = [1400.34 144.92 1522.5];
PT_C = [1479.8 -336.99 1318.4];
PT_D = [1379.35 -826.37 1487.3];
PT_E = [1480.25 -808.17 1486.42];
PT_F = [1354.91 82.32 1494.36];
PT_G = [1456.45 82.26 1480.18];
%
%Drawing 2371908_2
lever_DS = 76.44;
lever_PS = 67.11;
lever_M = 55;
%
%From Catia
```

54
B.1. MATLAB SCRIPTS

PT_DS = [1376.26 32.13 1446.37];
PT_PS = [1416.18 -876.37 1434.76];
PT_MDS = [1383.33 -428.93 1418.80];
PT_MPS = [1372.36 -429.21 1419.58];
angle_offset = 2.2;
PT_M = [PT_MDS(1)+lever_M*sind(angle_offset)...
PT_MDS(2)+lever_M*cosd(angle_offset)...
PT_MDS(3)];

B.1.3 Mathematic parameters of linkage

corr3D = [2 -26 15]; %Due to simplification from 2D to 3D
a = lever_M;
b_DS = lever_DS;
b_PS = lever_PS;
f_DS = norm(PT_MDS-PT_DS)+corr3D(2);
f_PS = norm(PT_MPS-PT_PS)+corr3D(3);
g_DS = norm(PT_M-PT_E);
g_PS = norm(PT_M-PT_F)-corr3D(1);
angle_xi_PS = pi/4; %Experimental
angle_xi_DS = pi/5; %Experimental
angle_phi_DS = atan((PT_E(3)-PT_M(3))/abs(PT_E(2)-PT_M(2)));
angle_phi_PS = atan((PT_F(3)-PT_M(3))/abs(PT_F(2)-PT_M(2)));

B.1.4 Mathematical calculation of linkage

duration = 6; %s
resolution = 20; %fps
points = duration*resolution; %no of points
space = 50;

%Starting values
A = [0 0];
B_DS = g_DS*[cos(angle_phi_DS) sin(angle_phi_DS)];
B_PS = g_PS*[-cos(angle_phi_PS) sin(angle_phi_PS)];

%Calculations:
i = 1:points;
angle_theta = (i./10);
D = [A(1) + a*cos(angle_theta);A(2) + a*sin(angle_theta)]; %
[X;Y]

55
APPENDIX B. MATLAB CODE

% Intersections
[C_DS, th] = intersect_circle(D, B_DS, f_DS, b_DS); % First solution is valid

[th2, C_PS] = intersect_circle(D, B_PS, f_PS, b_PS); % Second solution is valid

clear th*; % Delete unused solutions

% Error calculation
error(1,:) = sqrt((C_DS(1,:)−D(1,:)).^2+(C_DS(2,:)−D(2,:)).^2)−f_DS;

error(2,:) = sqrt((C_PS(1,:)−D(1,:)).^2+(C_PS(2,:)−D(2,:)).^2)−f_PS;

% Calculate wiper arms
angle_alphaB_DS = angle_xi_DS + atan((C_DS(1,:)−B_DS(1))/(C_DS(2,:)−B_DS(2)));

angle_alphaB_PS = angle_xi_PS + atan((C_PS(1,:)−B_PS(1))/(C_PS(2,:)−B_PS(2)));

Q1 = 118.6;
Q2 = 941.9;
Q3 = 151.1;
Q4 = 947.5;

% Test
A1 = [B_PS(1)+Q1−cosd(44+rad2deg(angle_alphaB_PS)); B_PS(2) +Q1*sind(44+rad2deg(angle_alphaB_PS))];

B1 = [B_PS(1)+Q2−cosd(3.2+rad2deg(angle_alphaB_PS)); B_PS(2)+Q2*sind(3.2+rad2deg(angle_alphaB_PS))];

A2 = [B_DS(1)+Q3−cosd(53+rad2deg(angle_alphaB_DS)); B_DS(2) +Q3*sind(53+rad2deg(angle_alphaB_DS))];

B2 = [B_DS(1)+Q4−cosd(7.1+rad2deg(angle_alphaB_DS)); B_DS(2)+Q4*sind(7.1+rad2deg(angle_alphaB_DS))];

B.1.5 Motor data from supplier

dir = 'motor';

DGCMH = load(fullfile(dir, 'Dry Glass Cold Motor HS corrected.mat'));

DGCLM = load(fullfile(dir, 'Dry Glass Cold Motor LS corrected.mat'));

DGWMH = load(fullfile(dir, 'Dry Glass Warm Motor HS corrected.mat'));

56
B.1. MATLAB SCRIPTS

corrected.mat');
6 DGWML = load(fullfile(dir, 'Dry Glass Warm Motor LS corrected.mat'));
7 WGCML = load(fullfile(dir, 'Wet Glass Cold Motor LS corrected.mat'));
8 WGCML = load(fullfile(dir, 'Wet Glass Cold Motor LS corrected.mat'));
9 WGWML = load(fullfile(dir, 'Wet Glass Warm Motor HS corrected.mat'));
10 WGWML = load(fullfile(dir, 'Wet Glass Warm Motor LS corrected.mat'));
11
12 windowSize = 55;
13 b = (1/windowSize) * ones(1, windowSize);
14 a = 1;
15
16 if (plot == 1)
17 run custom_plot.m
18
19 desc = [ ];
20 plot(DGCMH.t, filter(b,a,DGCMH.T)); desc = [ desc, 'Dry glass, cold motor, high speed'];
21 plot(DGWML.t, filter(b,a,DGWML.T)); desc = [ desc, 'Dry glass, cold motor, low speed '];
22 plot(DGWML.t, filter(b,a,DGWML.T)); desc = [ desc, 'Dry glass, warm motor, high speed '];
23 plot(DGWML.t, filter(b,a,DGWML.T)); desc = [ desc, 'Dry glass, warm motor, low speed '];
24 % plot(WGCML.t, filter(b,a,WGCML.T)); desc = [ desc, 'Wet glass, cold motor, high speed '];
25 % plot(WGCML.t, filter(b,a,WGCML.T)); desc = [ desc, 'Wet glass, cold motor, low speed '];
26 % plot(WGWML.t, filter(b,a,WGWML.T)); desc = [ desc, 'Wet glass, warm motor, high speed '];
27 % plot(WGWML.t, filter(b,a,WGWML.T)); desc = [ desc, 'Wet glass, warm motor, low speed '];
28 legend(desc);
29 axis([-0.15 4 -20 40]);
30 end

B.1.6 Simulink parameters

gy = 0; %-9.80665;
density = 8000;
ext_load = 3; %External load [Nm]
damping = 1.5e-2; %joint [Nm/(deg/s)]
stiffness = 1e-9; %joint [Nm/deg]
n = 73; %Gearing
voltage = 28; %V
f = 20; %Hz

%Controller parameters
ilim_gain = 1.03; % [%]
Ts = 0.1; %Period time [s]
uc_enable = 0.35; %Controller start
VCC = 5; %System voltage scaling factor
AccVal = 0.7; % [%]
TurnVal = 0.7; % [%]
sim ('four_bar')

B.1.7 Plot style

h=figure();
hold on; grid on;

/animate position and size
x0=100;
y0=100;
width=1000;
height=600;
set(gcf,'units','points','position',[x0, y0, width, height])

B.1.8 Trajectory figure generation

cleanup();
x = [0 0.1 0.9 1 1.1 1.9 2];
y = [0.7 1 1 0.3 1 1 0.7];
f = 14;
figure();
hold on; axis equal;
title('Voltage trajectory profile','Font-size',f*1.5)
xlabel('Time [s]','Font-size',f)
ylabel('Voltage [V]','Font-size',f)
axis([-0.1 4.2 0 1.1])
B.2. MATLAB FUNCTIONS

14 area(x(1:2),y(1:2));
15 area([x(2:3) x(3) x(5) x(5:6)],[y(2:3) 0 0 y(5:6)]);
16 area(x(3:4),y(3:4));
17 area(x(4:5),y(4:5));
18 area(x(6:7),y(6:7));
19 legend('Soft start', 'RUN.RUN', 'RUN.turn\_point\_dec', 'RUN.turn\_point\_acc', 'Soft\_stop')

B.2 Matlab functions

B.2.1 Workspace cleanup

1 function [] = cleanup()
2  %Makes the workspace tidy
3     clear; clc; clf; close all;
4     format compact;
5 end

B.2.2 Calculate circle intersections

1 function [ sol_1, sol_2 ] = intersect_circle(A,B,aa,bb)
2  %Calculates intersection between circles
3  
4     %sym helper
5     len = length(A);
6     sol_1 = zeros(1,len);
7     sol_2 = zeros(1,len);
8     
9     N = [1 0;0 -1];  %Negate second element
10     F = [0 1;1 0];  %Flip elements
11     for i=1:len
12         d = norm(B-A(:,i).');
13         if d > aa + bb
14             disp('No solution! Separate circles');
15         elseif d < abs(aa - bb)
16             disp('No solution! A contains B');
17         elseif d == 0 & & aa == bb
18             disp('No solution! Same circle');
19         end
20         
21         a = (aa^2-bb^2+d^2)/(2*d);
22         h = sqrt(aa^2-a^2);
APPENDIX B. MATLAB CODE

\[ P2 = A(:,i)\cdot + a*(B-A(:,i)\cdot )/d; \]
\[ Q = h*(B-A(:,i)\cdot )/d; \]
\[ \text{tmp}_A = P2 + Q*F*N; \]
\[ \text{tmp}_B = P2 + -Q*F*N; \]
\[ \text{sol}_1(1,i) = \text{tmp}_A(1); \]
\[ \text{sol}_1(2,i) = \text{tmp}_A(2); \]
\[ \text{sol}_2(1,i) = \text{tmp}_B(1); \]
\[ \text{sol}_2(2,i) = \text{tmp}_B(2); \]

B.2.3 Extract motor data

function [] = extract_motor()

%extract_motor() Extracts data from Excel files from supplier
directory = 'motor';
files = dir(fullfile(directory, '*xlsx'));
for file = files'
    filename = fullfile(directory, file.name);
    [NUM,TXT,RAW]=xlsread(filename, 'A1:N5001');
    V_meas = NUM(:,1);
B.2. MATLAB FUNCTIONS

```
A_calc = NUM(:,3);
A_max = max(A_calc);
A_avg = NUM(:,7);
XYZ = NUM(:,2);
t = XYZ.*0.002;

% Torque and speed calculations
Tp = NUM(1:2,10);
Sp = NUM(1:2,13);
T = (A_avg-Tp(2))/Tp(1);
S = Sp(1)*T+Sp(2);

sample_len = 100;
T(isnan(T))=0;
noise_level_T = 0.3;
for i=1:length(T)
    if (mean(T(1:i+sample_len))>=T(i)+noise_level_T)
        if (T(i) ~= 0)
            V_meas = V_meas(i:end);
            A_calc = A_calc(i:end);
            A_avg = A_avg(i:end);
            t = t(i:end)-t(i);
            T = T(i:end)-T(i);
            S = S(i:end);
            break
        end
    end
end

str = erase(filename,'.xlsx');
save(str,'V_meas','A_calc','A_max','A_avg','XYZ','t','Tp','Sp','T','S');
end
```

B.2.4 Analytic plot

```
function [] = analytic_plot(A,A1,A2,B1,B2,B_DS,b_DS,B_PS,b_PS,C_DS,C_PS,D,a,space,points,resolution)
% Plots analytic version of wiper system
    % Draw once
    axis('equal',[min(B1(1,:))-space max(B2(1,:))+space A(2))
```
APPENDIX B. MATLAB CODE

```matlab
% Plot trajectory circles
traj_D = viscircles(A,a,'Linestyle','--');
traj_C_DS = viscircles(B_DS,b_DS,'Linestyle','--');
traj_C_PS = viscircles(B_PS,b_PS,'Linestyle','--');
plot([A(1) B_DS(1) B_PS(1)],[A(2) B_DS(2) B_PS(2)],'*');
% Plot traj centers
clear traj*; %Delete unused solutions

for t=1:points
    linkage = line([A1(1,t) B_PS(1) C_PS(1,t) D(1,t) NaN A(1) D(1,t)
                     C_DS(1,t) B_DS(1) A2(1,t)]),...
                   [A1(2,t) B_PS(2) C_PS(2,t) D(2,t) NaN A(2) D(2,t)
                     C_DS(2,t) B_DS(2) A2(2,t)]);
wipers = line([A1(1,t) B1(1,t) NaN A2(1,t) B2(1,t)
                 ],[A1(2,t) B1(2,t) NaN A2(2,t) B2(2,t)]);
pause(1/resolution);
delete(linkage);
%delete(wipers);
end
```

Appendix C

Inventory list Simulink

Model was used in Simulink [37] with toolboxes according to appendix A.

C.1 Analytical model

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>PS link</td>
<td>SubSystem</td>
</tr>
<tr>
<td>A</td>
<td>SubSystem</td>
</tr>
<tr>
<td>Base-Rocker Revolute Joint</td>
<td>SimscapeMultibodyBlock</td>
</tr>
<tr>
<td>Base-Rocker Revolute Joint1</td>
<td>SimscapeMultibodyBlock</td>
</tr>
<tr>
<td>Controlled PWM Voltage</td>
<td>SimscapeBlock</td>
</tr>
<tr>
<td>Controlled Voltage Source</td>
<td>SimscapeBlock</td>
</tr>
<tr>
<td>Controller</td>
<td>SubSystem</td>
</tr>
<tr>
<td>Controller/park</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/Clock</td>
<td>Clock</td>
</tr>
<tr>
<td>Controller/Constant1</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Constant2</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Constant3</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Constant4</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Constant5</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Period calc</td>
<td>SubSystem</td>
</tr>
<tr>
<td>Controller/Period calc/t</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/Period calc/Trigger</td>
<td>TriggerPort</td>
</tr>
<tr>
<td>Controller/Period calc/Gain</td>
<td>Gain</td>
</tr>
<tr>
<td>Controller/Period calc/Memory</td>
<td>Memory</td>
</tr>
<tr>
<td>Controller/Period calc/Subtract</td>
<td>Sum</td>
</tr>
<tr>
<td>Controller/Period calc/period</td>
<td>Outport</td>
</tr>
<tr>
<td>Controller/Switch1</td>
<td>Switch</td>
</tr>
<tr>
<td>Controller/To VCC</td>
<td>SubSystem</td>
</tr>
<tr>
<td>Controller/To VCC/PWM</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/To VCC/Gain</td>
<td>Gain</td>
</tr>
</tbody>
</table>
APPENDIX C. INVENTORY LIST SIMULINK

Controller/To VCC/Simulink-PS Converter
Controller/To VCC/Unit Delay
Controller/To VCC/VCC
Controller/Voltage profile1
Controller/i-limit
Controller/i-limit/t
Controller/i-limit/Constant
Controller/i-limit/Constant4
Controller/i-limit/Gain1
Controller/i-limit/Moving Minimum
Controller/i-limit/PS-Simulink Converter
Controller/i-limit/Switch
Controller/i-limit/Switch2
Controller/i-limit/Zero-Order Hold
Controller/i-limit/Zero-Order Hold1
Controller/i-limit/i
Controller/i-limit/i_real
Controller/i-limit/i_lim
Controller/current
Controller/PWM ref
Current Sensor
DC Motor
DS B
DS C
DS D
DS lever
DS link
Electrical Reference
Electrical Reference1
H-Bridge1
Ideal Angular Velocity Source
Ideal Rotational Motion Sensor
Ideal Torque Sensor
Inductor
Inductor1
Load
Load/Abs
Load/Add
Load/Constant
Load/Constant1
Load/Cos
Load/From
Load/Gain2
Load/PS-Simulink Converter1

SubSystem
UnitDelay
PMIOPort
SubSystem
SubSystem
Inport
Constant
Constant
Gain
MATLABSystem
SubSystem
Switch
Switch
ZeroOrderHold
ZeroOrderHold
PMIOPort
Outport
Outport
PMIOPort
PMIOPort
SimscapeBlock
SubSystem
SimscapeMultibodyBlock
SimscapeMultibodyBlock
SubSystem
SubSystem
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
SimscapeBlock
Abs
Sum
Constant
Constant
Trigonometry
From
Gain
SubSystem
C.1. ANALYTICAL MODEL

Load/Product
Load/Product1
Load/Sign
Load/Simulink-PS Converter1
Load/vel
Load/Load
Load2
Load2/Abs
Load2/Add
Load2/Constant
Load2/Constant1
Load2/Cos
Load2/From
Load2/Gain2
Load2/PS-Simulink Converter1
Load2/Product
Load2/Product1
Load2/Sign
Load2/Simulink-PS Converter1
Load2/vel
Load2/Load
Mechanical Rotational Reference
Mechanism Configuration
Motor axle
Motor lever
Origo
PS B
PS C
PS D
PS lever
PS-Simulink Converter1
Park sensor
Park sensor/Constant
Park sensor/Constant2
Park sensor/Constant3
Park sensor/Mod
Park sensor/PS-Simulink Converter2
Park sensor/Switch
Park sensor/angle
Park sensor/pulse
Rocker-Base Transform
Rocker-Base Transform1
Scope
Scope1
Product
Product
Signum
SubSystem
PMIOPort
PMIOPort
SubSystem
Abs
Sum
Constant
Constant
Trigonometry
From
Gain
SubSystem
Product
Signum
SubSystem
PMIOPort
PMIOPort
SimscapeBlock
SimscapeMultibodyBlock
SimscapeMultibodyBlock
SubSystem
SimscapeMultibodyBlock
SimscapeMultibodyBlock
SubSystem
SubSystem
SimscapeMultibodyBlock
SimscapeMultibodyBlock
SubSystem
SubSystem
SimscapeMultibodyBlock
SimscapeMultibodyBlock
SubSystem
SubSystem
SimscapeMultibodyBlock
SimscapeMultibodyBlock
SubSystem
SubSystem
SimscapeMultibodyBlock
SimscapeMultibodyBlock
SubSystem
SubSystem
SimscapeMultibodyBlock
Switch
PMIOPort
Outport
SimscapeMultibodyBlock
SimscapeMultibodyBlock
Scope
Scope
APPENDIX C. INVENTORY LIST SIMULINK

Solver Configuration SubSystem
Solver Configuration1 SubSystem
To WS1 SubSystem
To WS1/Mux1 Mux
To WS1/PS-Simulink Converter SubSystem
To WS1/PS-Simulink Converter4 SubSystem
To WS1/PS-Simulink Converter5 SubSystem
To WS1/PS-Simulink Converter6 SubSystem
To WS1/To Workspace ToWorkspace
To WS1/Position PMIOPort
To WS1/Velocity PMIOPort
To WS1/Acceleration PMIOPort
To WS1/Actuator Torque PMIOPort
To WS2 SubSystem
To WS2/Mux1 Mux
To WS2/PS-Simulink Converter SubSystem
To WS2/PS-Simulink Converter4 SubSystem
To WS2/PS-Simulink Converter5 SubSystem
To WS2/PS-Simulink Converter6 SubSystem
To WS2/To Workspace ToWorkspace
To WS2/Position PMIOPort
To WS2/Velocity PMIOPort
To WS2/Acceleration PMIOPort
To WS2/Actuator Torque PMIOPort
To WS3 SubSystem
To WS3/Goto Goto
To WS3/Mux1 Mux
To WS3/PS-Simulink Converter SubSystem
To WS3/PS-Simulink Converter1 SubSystem
To WS3/PS-Simulink Converter2 SubSystem
To WS3/PS-Simulink Converter3 SubSystem
To WS3/PS-Simulink Converter4 SubSystem
To WS3/PS-Simulink Converter5 SubSystem
To WS3/PS-Simulink Converter6 SubSystem
To WS3/PS-Simulink Converter7 SubSystem
To WS3/To Workspace ToWorkspace
To WS3/Position PMIOPort
To WS3/Velocity PMIOPort
To WS3/Acceleration PMIOPort
To WS3/Actuator Torque PMIOPort
To WS3/Force constrained PMIOPort
To WS3/Torque constrained PMIOPort
To WS3/Force total PMIOPort
To WS3/Torque total PMIOPort
### C.2. DSPACE MODEL

- Voltage Sensor: SimscapeBlock
- Voltage Source: SimscapeBlock
- World Frame: SimscapeMultibodyBlock
- Worm Gear: SimscapeBlock

#### C.2 dSPACE model

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADC VCC-level</td>
<td>SubSystem</td>
</tr>
<tr>
<td>ADC VCC-level/ADC_TYPE1_M1_CON1</td>
<td>SubSystem</td>
</tr>
<tr>
<td>ADC VCC-level/ADC_TYPE1_M1_CON2</td>
<td>SubSystem</td>
</tr>
<tr>
<td>ADC VCC-level/Gain_Current</td>
<td>Gain</td>
</tr>
<tr>
<td>ADC VCC-level/Gain_Park</td>
<td>Gain</td>
</tr>
<tr>
<td>ADC VCC-level/Current</td>
<td>Outport</td>
</tr>
<tr>
<td>ADC VCC-level/Park</td>
<td>Outport</td>
</tr>
<tr>
<td>Constant</td>
<td>Constant</td>
</tr>
<tr>
<td>Constant1</td>
<td>Constant</td>
</tr>
<tr>
<td>Constant2</td>
<td>Constant</td>
</tr>
<tr>
<td>Constant3</td>
<td>Constant</td>
</tr>
<tr>
<td>Constant4</td>
<td>Constant</td>
</tr>
<tr>
<td>Constant5</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller</td>
<td>SubSystem</td>
</tr>
<tr>
<td>Controller/i</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/park</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/enable</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/Clock</td>
<td>Clock</td>
</tr>
<tr>
<td>Controller/Constant1</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Constant3</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Constant4</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Constant5</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Current limit</td>
<td>SubSystem</td>
</tr>
<tr>
<td>Controller/Current limit/t</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/Current limit/i</td>
<td>Inport</td>
</tr>
<tr>
<td>Controller/Current limit/Constant</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Current limit/Constant4</td>
<td>Constant</td>
</tr>
<tr>
<td>Controller/Current limit/Gain1</td>
<td>Gain</td>
</tr>
<tr>
<td>Controller/Current limit/Moving Minimum</td>
<td>MATLABSystem</td>
</tr>
<tr>
<td>Controller/Current limit/Switch</td>
<td>Switch</td>
</tr>
<tr>
<td>Controller/Current limit/Switch2</td>
<td>Switch</td>
</tr>
<tr>
<td>Controller/Current limit/i_real</td>
<td>Outport</td>
</tr>
<tr>
<td>Controller/Current limit/i_lim</td>
<td>Outport</td>
</tr>
<tr>
<td>Controller/Period calc</td>
<td>SubSystem</td>
</tr>
<tr>
<td>Controller/Period calc/t</td>
<td>Inport</td>
</tr>
</tbody>
</table>
APPENDIX C. INVENTORY LIST SIMULINK

Controller/Period calc/Trigger
Controller/Period calc/Gain
Controller/Period calc/Memory
Controller/Period calc/Subtract
Controller/Period calc/period
Controller/Rate Transition
Controller/Rate Transition1
Controller/Rate Transition2
Controller/Scope1
Controller/Switch1
Controller/Voltage profile
Controller/PWM
Current filter
DIO_TYPE1_PWM_VP_M1_C1
DIO_TYPE1_PWM_VP_M1_C2
DIO_TYPE1_PWM_VP_M1_C3
DIO_TYPE1_PWM_VP_M1_C4
INA
INB
Main start
Mode
PWM
Park filter
RTI Data
SEL0
Switch1
Switch2
period

TriggerPort
Gain
Memory
Sum
Outport
RateTransition
RateTransition
RateTransition
Scope
Switch
SubSystem
Outport
MATLABSystem
SubSystem
SubSystem
SubSystem
Constant
Constant
ManualSwitch
ManualSwitch
Constant
Switch
SubSystem
Constant
Switch
Switch
Constant