CODEN:LUTEDX/(TEIE-5309)/1-84/(2013)

Industrial Electrical Engineering and Automation

Dynamic Gear Shifting of an Automated Manual Transmission

Malin Reinholds
Steve Andreasson

Division of Industrial Electrical Engineering and Automation
Faculty of Engineering, Lund University

LUND UNIVERSITY

MASTER THESIS

Dynamic Gear Shifting of an Automated Manual Transmission

Authors:
Steve Andreasson
& Malin Reinholds

Supervisor:
Rasmus Andersson
Examiner:
Mats Alakūla

A thesis submitted in fulfilment of the requirements for the degree of Master of Electrical Engineering

in the department of

Industrial Electrical Engineering and Automation

July 2013

LUND UNIVERSITY

LUND UNIVERSITY

Abstract

LTH
Industrial Electrical Engineering and Automation

Master of Electrical Engineering

Dynamic Gear Shifting of an Automated Manual Transmission

by Steve Andreasson
& Malin Reinholds

To be able to change gears in a transmission it is necessary to remove the torque when disengaging and engaging gears in order not to damage the teeth of the cogs. If the driving torque from the engine to the wheels is interrupted for a longer period of time it can be perceived as very uncomfortable for both the driver and passengers in the vehicle. A gear change can be divided into three main stages; disengaging the current gear, synchronizing the countershaft and finally engaging the new gear. Engaging and disengaging a gear is a very fast process as it is and no attempts are made to improve upon this further. However, synchronising the countershaft is rather slow in conventional hybrid vehicles thus the potential for improving this stage is vast. One way of speeding up the synchronisation process is to actively steer the speed of the countershaft with the aid of an electrical machine. A long and slender machine is preferable since the acceleration is proportional to the inertia of the machine, and therefore increases with the radius. This thesis studies the potential of such an electric machine aided dynamic gear change. The investigation is carried out as a series of empirical tests in a testing rig comprised of a transmission, an electric machine, a flywheel, power electronics and a control system based on both hardware and software from National Instruments.

A key element to achieving a successful gear change is a quick yet stable motor control since this determines the synchronisation time. In a conventional vehicle the countershaft brake used for synchronisation can achieve a deceleration of 2000 rpm/s. Using the electrical machine instead a deceleration of 3347 rpm/s is recorded which corresponds to a 67 % time improvement. It is also shown that a full gear change from 3rd to 5th gear can be completed in 160 ms. These are very positive results and the main objectives of the thesis have been met.

Contents

List of Figures

List of Tables

Abbreviations

cRIO	**C**ompact **R**econfigurable **I**nput **O**utput
EM	**E**lectrical **M**achine
ExSAM	**E**xternal **S**tarter, **A**lternator **M**otor
I-SAM	**I**ntegrated **S**tarter, **A**lternator **M**otor
LUT	**L**ook **U**p **T**able
PMSM	**P**ermanent **M**agnet **S**ynchronous **M**achine
PMU	**P**ower **M**anagement **U**nit
PTO	**P**ower **T**ake-**o**ff
TECU	**T**ransmission **E**lectronic **C**ontrol **U**nit

Chapter 1

Introduction

The steady increase of oil prices together with a rising environmental concern has created a demand for more energy efficient transport solutions. One way of doing this is through hybridisation of vehicles. By hybridising a combustion engine vehicle the operating point of the combustion engine can be kept closer to the maximum efficiency point for a longer period of time compared to conventional vehicles. This results in lower fuel consumption and emissions. Another major benefit is that brake energy can be regenerated and stored instead of being wasted as heat energy as in a conventional friction brake. Volvo has designed such an electric hybrid drive train concept called I-SAM; Integrated Starter, Alternator Motor.

1.1 I-SAM

The I-SAM[1] concept is a parallel hybrid meaning that both the electrical motor and the combustion engine provide power to the wheels. This enables the vehicle to run in conventional mode, full electric mode or hybrid mode. The additional components compared to a conventional driveline are a permanent magnet synchronous machine (PMSM), a power converter, an energy storage system and a Power Management Unit (PMU). The electric machine is located between the clutch and the transmission as seen in figure 1.1. This placement restricts the length of the machine. In order to fit an electrical machine with sufficient power the design has a high diameter to length ratio. This results in a relatively high moment of inertia since the moment of inertia of a

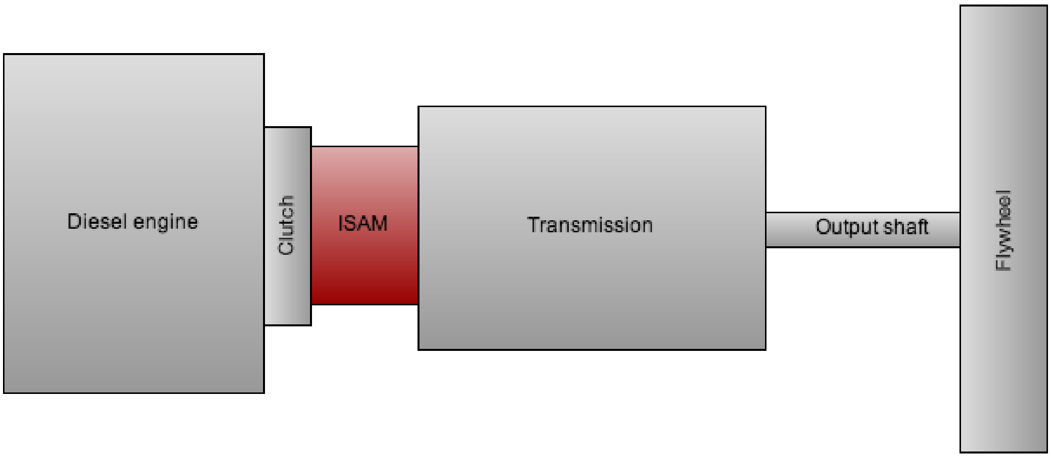

FIGURE 1.1: I-SAM configuration

cylindrical object grows with the radius raised to the power of four. This in turn causes longer acceleration times of the electrical machine since the acceleration time is directly proportional to the moment of inertia. Longer acceleration times have a negative effect on the time it takes to change gear since the electrical machine is used to synchronize the shafts in the transmission.

1.2 ExSAM

An alternative placement of the electrical machine is along the side of the transmission, rather than in line with it, as in the I-SAM project. Instead of connected at the clutch, the machine is connected to the drive train at the power take-off (PTO) as seen in figure 1.2. This allows for the implementation of a fixed gear step resulting in an electric machine speed higher than that of I-SAM. The concept is called ExSAM since the machine is mounted externally on the gearbox rather than integrated in the drive train as in I-SAM. The main benefit of this configuration that is relevant for this master thesis is that the design of the electrical machine is long and slender instead of short with a large radius. This means that the moment of inertia is heavily reduced without compromising the electric torque contribution to the drivetrain, allowing faster acceleration times. Other benefits with ExSAM include a lighter unit that is cheaper to produce and potentially easier to install [2].

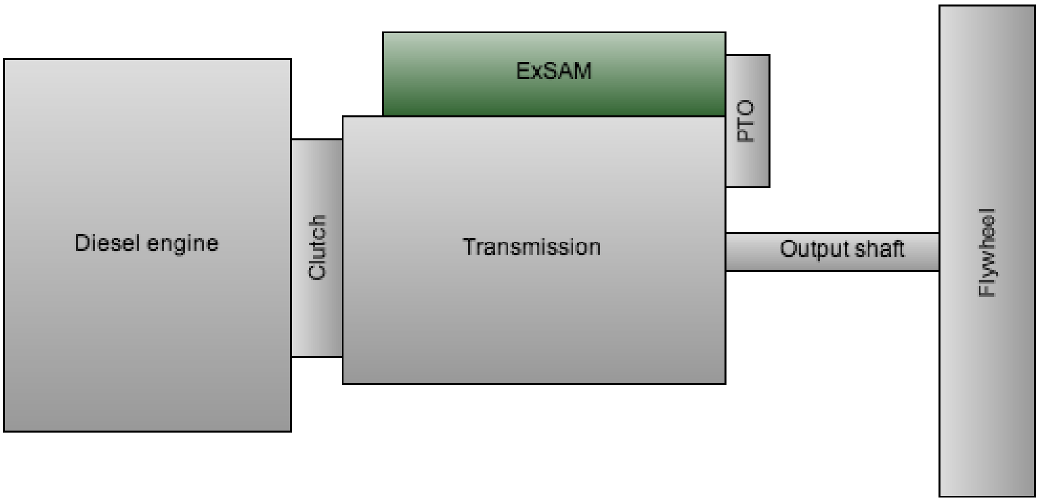

FIGURE 1.2: ExSAM configuration

1.3 Objectives

The main objective of this master thesis is to conduct a dynamic gear shifting sequence in a Volvo iShift gearbox connected with an ExSAM via the power take-off. Furthermore, there is an ambition to empirically prove that the time of torque interruption in the gear shifting process in a hybrid driveline can be significantly decreased with the assistance of the ExSAM. At present the synchronisation between the output shaft and the countershaft when performing a gear change is achieved with the help of the diesel engine and a friction brake. Decelerating the countershaft with the brake can be done with a speed of 2000 rpm/s on the input shaft. Accelerating the countershaft is done with the diesel engine and takes even longer. With the help of the electrical machine it is hoped that these synchronisation times can be improved. Note that the actual time for engaging gears is not improved in this project as the gear changing algorithms of the iShift robot are circumvented. The total gear change time is also likely to come out worse than before but with an improvement of the actual synchronization time. This is however not considered to be a problem if the concept is further developed at Volvo since the control can then be incorporated into the iShift robot rather than circumventing the existing algorithms. Nevertheless, when evaluating the result of this project it has to be taken into consideration. The programming is done in National Instruments' programming software LabVIEW. The power electronics unit comes from Vacon and is controlled by Vacon's own software, NC Drive. The work is structured into the following stages:

- Study of the concept of dynamic gear changing. This includes a study of a previous master thesis on the subject by Axel Bergman and Per Byrhult; *Automated shifting of a manual sequential transmission in a hybrid vehicle* (2009)
- Solving for a communication strategy between the computer software and the transmission
- Programming a gear shifting sequence
- Tuning a working motor control
- Setting up the power electronics including programming the power converter's software
- Basic testing on a small scale rig at Lund University
- Final testing on a full scale rig at Sibbhultsverken
- Processing and evaluating test results

1.4 Limitations

The AT2412D transmission used in this project has three different types of gears; two synchronized split gears and three unsynchronised main gears much like the ones found in a normal manual transmission. Finally it has two synchronized range gears comprised of a planetary gear. This results in a total of twelve gears ($2 \cdot 3 \cdot 2 = 12$). The split is located between the input shaft and the countershaft and increment or decrement the gear by one for a pair of gears e.g. $1st$ and $2nd$, $3rd$ and $4th$ etc. The main gears connect the countershaft to the main shaft and have four different settings including a reverse gear. By changing these gears the transmission gears up or down by two or four steps e.g. from $1st$ to $3rd$ or $1st$ to $5th$ and reversed. Finally the range gear has two settings that divide the gears into a lower set of gears ($1st$ to $6th$) and an upper set of gears ($7th$ to $12th$) and is located between the main and output shaft, see section 2.2 for further details.

Ideally the goal would be to change freely between all the gears. However due to the limited amount of time and the fact that the objective of the study only requires changes of the unsynchronized main gears this is where the focus of the thesis work is put. The split gear is included but not tested in detail since neither the clutch nor diesel engine is intended to be incorporated at this stage. A selector for the range gear is implemented allowing for tests to run either in the low set of gears or the high set of gears. However it is not implemented in such a way that dynamic range gear changes will be possible. Since it is a matter of very fast sequences the communication times are limiting. This would, as mentioned earlier, be solved by incorporating the control into the transmission control unit (TECU) instead of using external control as is the case in this thesis.

1.5 Previous work

This master thesis is a part of the ExSAM project that has been going on for more than three years. Since the start of the project several people have been involved, some from the very beginning to the present date and some who have made contributions along the way.

The most recent Master thesis done in this project is *Dynamic control of PMSM via profibus communication between NI-CRio and Vacon frequency converter* by Hammad Khan. It investigates the possibility to run the entire communication between LabVIEW (the computer software) and the software in the power electronics through a Profibus link. The conclusion of the report is that even at its best the Profibus communication takes about 150 ms [3, p. 45] which is much too slow for a stable motor control.

An alternative solution has been created by PhD student Yury Loayza, which circumvents the Profibus link for the motor control by reading a resolver signal from the motor and directly feeding it to LabVIEW. This gives a very fast but noisy reading of the motor speed that requires filtering. Filtering increases the accuracy of the signal at the cost of the time it takes to filter it. A compromise is made that gives an accurate enough signal at a sampling time of 0.1 ms. In order to update the speed reference signal as fast as or faster than the speed signal from the resolver an analog reference signal used. The signal is sent directly to the Vacon control unit instead of through the Profibus. Due to the fact that a current reference signal should in theory be more stable it is chosen over using a voltage reference.

Apart from the knowledge gained from Khan's Master thesis, a lot of the programming done in this thesis is a development of the code that was written by Khan and the PhD students that helped him. Some parts have been heavily altered, some have been left intact and some have been removed altogether as they proved obsolete but the structure is basically the same.

1.5.1 Publications in relation to ExSAM

- Andersson R., Reinap A., Alaküla M. (2012), *Design and Evaluation of Electrical Machine for Parallel Hybrid Drive for Heavy Vehicles.* International Conference on Electrical Machines (ICEM2012), Marseille, France, Sept.2-5, 2012.
- Andersson R., Högmark C., Reinap A., Alaküla M. (2012), *Modular Three-phase Machines with Laminated Windings for Hybrid Vehicle Applications.* International Electric Drives Production Conference and Exhibition (EDPC2012), Nürnberg, Germany Oct. 16-17, 2012.
- Hammad Khan (2012), *Dynamic Control of PMSM via Profibus Communication Between NI-CRio and Vacon Frequency Converter* MSc thesis, Lund University 2012.

1.5.2 Testing facility

As part of the ExSAM project a testing rig has been built at Sibbhultsverken. It will be explained further in chapter 3.

Chapter 2

Theory

This chapter outlines the theory used to throughout this thesis starting with an introduction to the stages required to complete a gear change as well as some basic theory regarding gear boxes. This is followed by gear ratio calculations and a description of how a theoretical limit for the gear change can be determined.

2.1 Shift sequence

The different stages in a shifting sequence are shown in figure 2.1. The first step is to disengage the combustion engine from the transmission input shaft by disengaging the

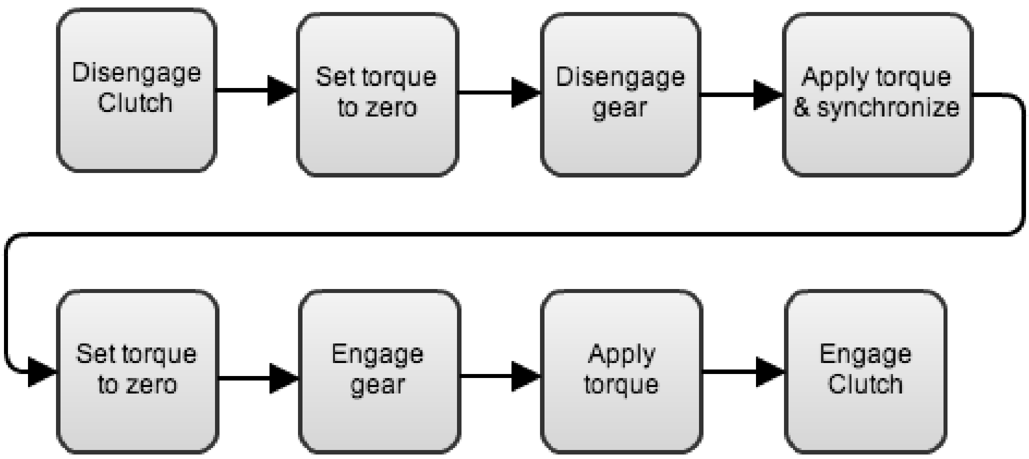

FIGURE 2.1: Shift sequence

clutch. The next step is to set the torque applied by the ExSAM to zero so that the gears are free to disengage without a driving force between gear teeth limiting movement. Once the gears are in neutral position the countershaft can be accelerated or decelerated to synchronous speed by applying torque from the ExSAM. Another short torque interruption follows to allow for the new gear to be engaged, after which driving torque is reapplied and the clutch can be engaged.

2.2 Transmission

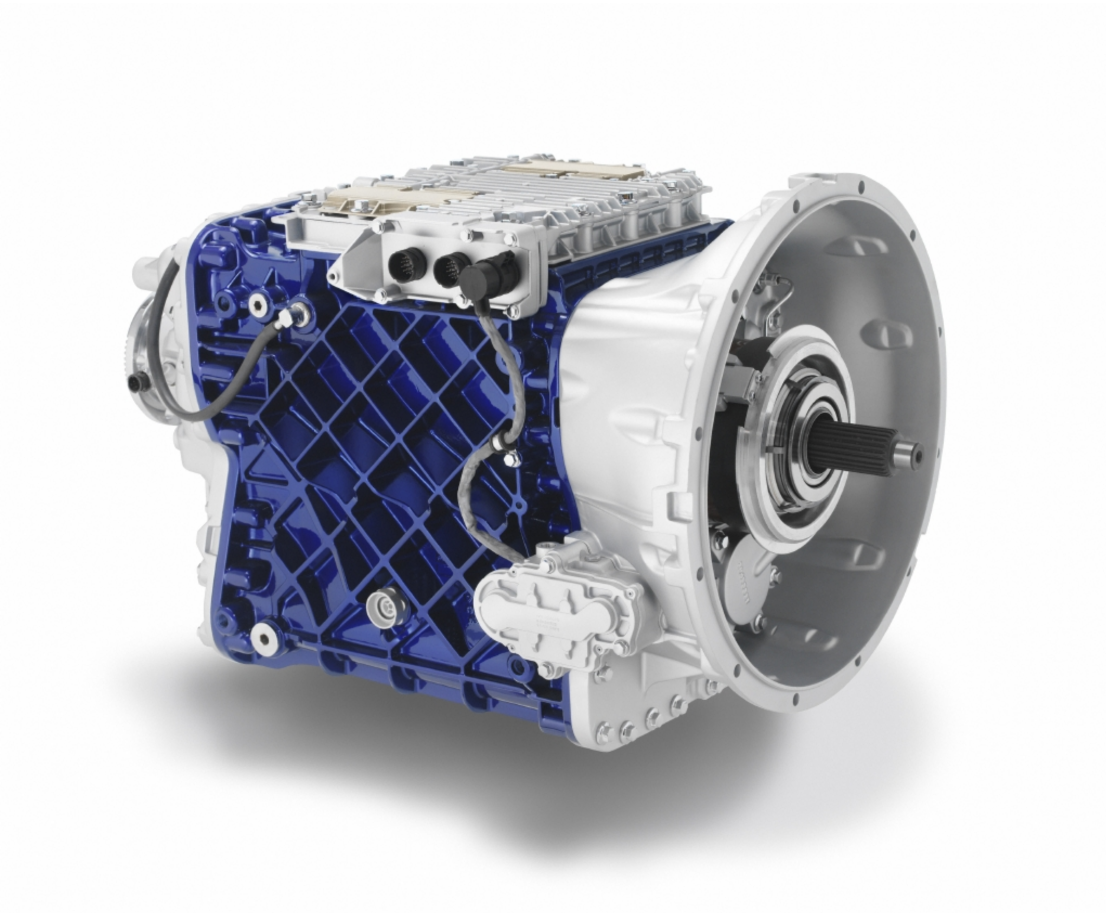

FIGURE 2.2: I-shift Transmission (Source: Volvo trucks [4])

This thesis uses a Volvo AT2412D transmission as shown in figure 2.2. Figure 2.3 shows the gears of the transmission and the four different shafts: Input shaft, countershaft, main shaft and output shaft. Connecting the input shaft and countershaft is a synchronized split gear (A). The main gearbox housing (B) houses the main gears. These are not synchronized and it is during a gear change involving the main gears that the gear

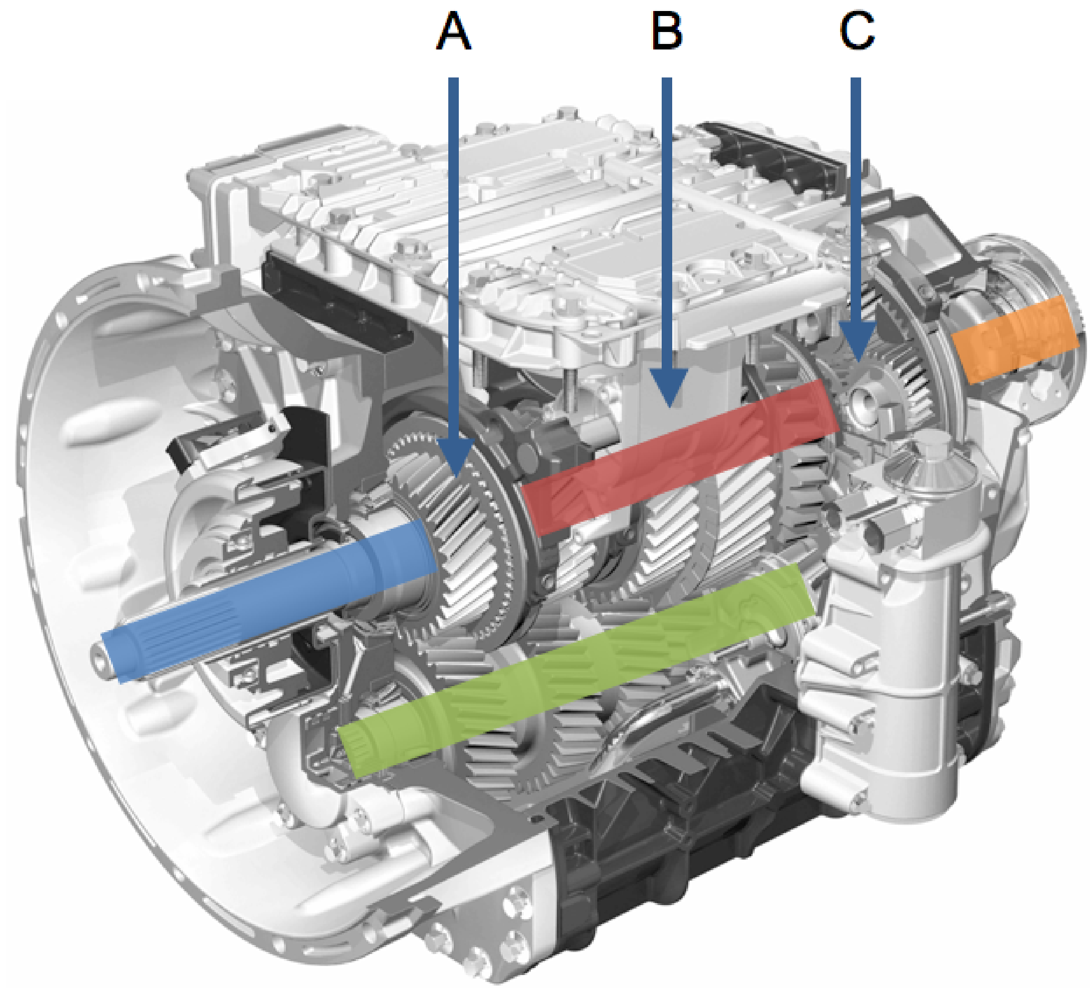

FIGURE 2.3: Cross section of the gearbox; *Blue*: Input shaft; *Green*: countershaft; *Red*: Main shaft; *Orange*: Output shaft
(Source: Volvo trucks[4])

shifting sequence can be improved by using the ExSAM to speed up the synchronization of the countershaft to the speed required by the new gear. The range gear (C) is a synchronized planetary gear set that is integrated with the output shaft. The split gear and main gears provide six different gear combinations (excluding reverse) that depending on the position of the range gear are gears $1st$ to $6th$ or $7th$ to $12th$.

Engaging a gear is done using a shifting fork whose position is controlled by pressurized air valves. There are four different forks in this transmission, two in the main section engaging either first or reverse gear and second or third gear, one engaging either high split or low split, and one engaging high range or low range. The position of each fork is controlled by two air valves releasing air pressure in opposite directions. By opening both valves at the same time the fork can be moved to neutral position. When either of the two main gear forks is engaged in a gear the other one is mechanically locked into neutral position. This is to prevent conflicting gears from being engaged at the same time which would be potentially disastrous.

2.3 Calculation of gear ratios and shaft speeds

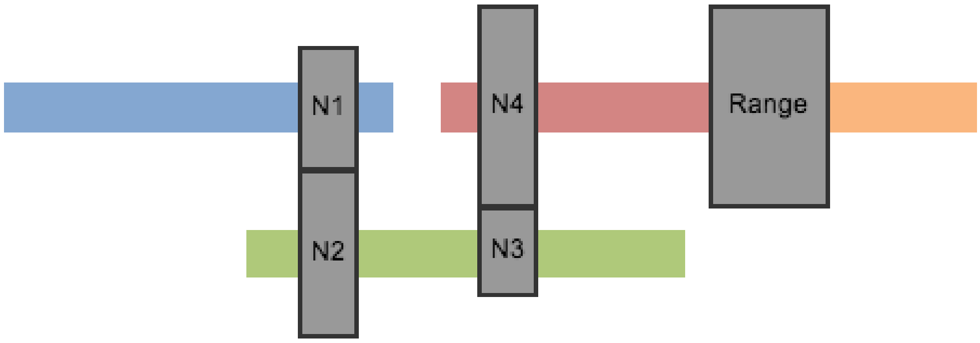

FIGURE 2.4: Block diagram of the power transfer within the gearbox

The figure 2.4 above is a simplified version of the transmission showing only the engaged gears and shafts. The notation N followed by a number represents the number of teeth on the cog wheel. Calculating the gear ratio of two engaged gears is done by dividing the number of teeth on the output gear by the number of teeth on the input gear. The speed of the output gear can then be calculated as follows:

$$\omega_2 = \frac{N_2}{N_1} \cdot \omega_1 \tag{2.1}$$

where ω_2 is the output gear speed and ω_1 is the input gear speed.

The speed of the main gear shaft can thus be determined by extending this principle:

$$\omega_4 = \frac{N_2}{N_1} \cdot \frac{N_4}{N_3} \cdot \omega_1 \tag{2.2}$$

The number of teeth on the engaged gears in AT2412D transmission are shown in table 2.1.

Table 2.1: Nr. of teeth on cog wheels

No. of teeth on cog wheel	Low split	High split / Third	Second	First
input/main shaft	31	35	46	41
countershaft	44	39	19	28

Calculating the speed of the output shaft requires the gear ratio of the planetary gear set that is the range gear. The gear ratio depends on which part of the planetary gear is locked to the output shaft at the time. Figure 2.5 below shows the basic structure of a planetary gear set which is comprised of an outer ring gear, three planet gears and a sun gear. The three planet gears are mounted on a Y-shaped planet carrier.

An equation based on the speeds of the ring gear (ω_R), the sun gear (ω_S) and the planet gear mount (ω_Y) as well as the number of teeth of the ring gear (N_R) and sun gear (N_S) can be set up as follows:

$$(N_R + N_S) \cdot \omega_Y = N_R \cdot \omega_R + N_S \cdot \omega_S \quad (2.3)$$

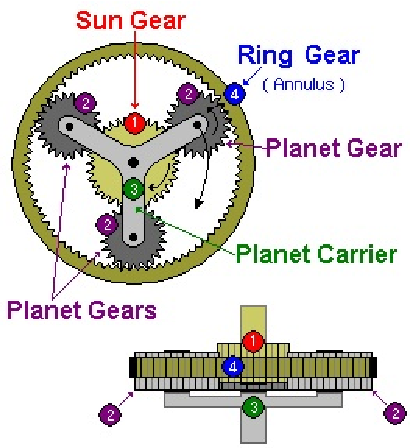

Figure 2.5: Planetary gear (Source: Wikipedia[5])

In the case of this thesis the ring gear is held stationary while the planet gears and sun gear rotate, which means that ω_R is zero. The equation can then be simplified to:

$$\omega_Y = \omega_S \cdot \frac{N_S}{N_R + N_S} \quad (2.4)$$

Thus resulting in the gear ratio $\frac{N_S}{N_R+N_S}$. In the AT2412D the sun gear has 23 teeth, the planet gears have 26 teeth and the ring gear has 77 teeth. Low range therefore has a gear ratio of 1 : 0.23. The high range position has gear ratio 1 : 1, which is achieved by locking the planet carrier to the ring gear.

2.4 Theoretical synchronization times

In order to calculate the theoretical synchronization times it is necessary to know the moment of inertia of all the rotating parts involved in the speed change of the counter-shaft. It is also necessary to know the performance of the electric machine, specifically the relationship between torque and speed. The result of these calculations sets a theoretical limit for a frictionless system, which provides an interesting reference value to compare with the test results.

Equation 2.5 shows the formula for calculating the moment of inertia of a cylinder shaped object, depending on its density (ρ), its inner radius (r), its outer radius (R), and its length (l).

$$J = \frac{1}{2} \cdot \pi \cdot \rho \cdot l \cdot (R^4 - r^4) \tag{2.5}$$

The gear ratios between interlocking gears are required to calculate the total moment of inertia of a system. To illustrate this, the total moment of inertia of two interlocking gears, gear 1 and gear 2, is calculated with equation 2.6. The moment of inertia of gear 1 is combined with the moment of inertia of gear 2 multiplied by the gear ratio squared.

$$J = J_1 + \left(\frac{N_1}{N_2}\right)^2 \cdot J_2 \tag{2.6}$$

Where J_1 and N_1 is the moment of inertia and number of teeth of gear 1, and J_2 and N_2 is the moment of inertia and number of teeth of gear 2.

Once the total moment of inertia of the system has been calculated the ideal time it takes to achieve a specific speed step using the ExSAM can be determined using equation 2.7.

$$\Delta t = \frac{J}{T} \cdot \Delta\omega \tag{2.7}$$

Where J is the total inertia of the system, T is the torque and $\Delta\omega$ is the size of the speed step.

Equation 2.7 can be derived from an adaptation of Newton's Second Law of physics as shown below, where a is the body's angular acceleration in rad/s^2, T is the total torque exerted on the body and J is the body's moment of inertia.

$$F = m \cdot a \Rightarrow T = J \cdot a \Rightarrow T = J \cdot \frac{d\omega}{dt} \Rightarrow dt = \frac{J}{T} \cdot d\omega \Rightarrow \Delta t = \frac{J}{T} \cdot \Delta\omega \tag{2.8}$$

2.5 Cascade PID-control

Speed control of the ExSAM is achieved through a cascaded PID control method as shown in figure 2.6 below.

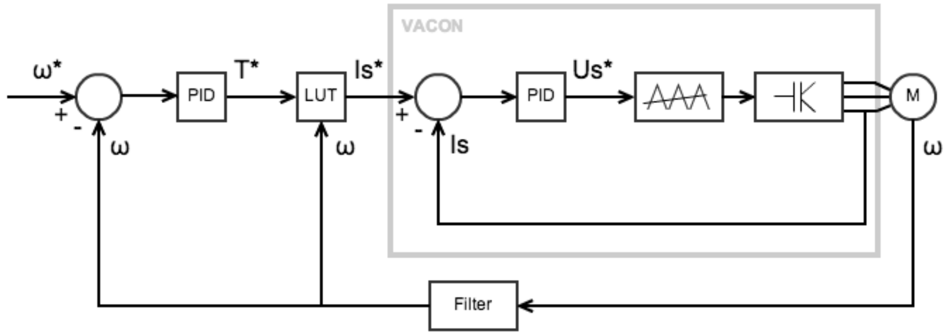

FIGURE 2.6: Control loop

Speed control takes place using a PID-control in the LabVIEW program that sends a motor torque reference to a Lookup Table. The torque reference (T*) together with the speed of the ExSAM (ω) is required as input for the lookup table (LUT), which returns two current references; the x-component and y-component of the stator current (Is). These current references are realized using a second PID-control in the Vacon control unit.

Lookup tables (LUT) is a method used to effectively implement field weakening by selecting a combination of x- and y-currents that fulfill both the torque requirements and the voltage limitations. This results in reduced flux linkage and the stator current runs along constant power curves. This allows the motor to run at speeds exceeding its base speed. The lookup tables for the ExSAM are completed and implemented in the LabVIEW program by Yury Loayza, PhD student at Lund University, however they have not yet been sufficiently tested. Consequently the gear shifting tests are limited to tests below the base speed of the ExSAM. Omitting the Lookup tables and thereby the possibility for field weakening means that the output from the speed PID-control is to be rescaled and sent directly as a current reference to the PID-control in the Vacon control unit. Without an x-component to the current the flux linkage is constant and it can be seen from equation 2.9 that if Is_x is 0 then torque is directly proportional to Is_y. This supports the method of simply rescaling the speed PID-control output to achieve a current reference.

$$T = \psi \cdot Is_y + (L_x - L_y) \cdot Is_x \cdot Is_y \Rightarrow T = constant \cdot Is_y \tag{2.9}$$

Here T is motor torque per phase and pole pair, L_x is the x-axis inductance, L_y is the y-axis inductance, ψ is the permanent magnet flux.

The dynamics of the electrical magnitudes in the system (voltage, current and the resultant torque) are at least one order of magnitude faster than the dynamics of the mechanical magnitudes (speed). Therefore, if the PID-controller in the Vacon inverter is properly tuned, it can be assumed that the set point sent to the Vacon control unit is already achieved by the time one sampling period for the LabVIEW control has passed. The inverter and motor can therefore be modeled as as a first order low pass filter with a time constant equal to the sampling period in the torque control loop [6]. This means that the symmetric optimum theory can be used to tune the speed controller[7].

The equation for a PID-control is shown in equation 2.10

$$I(t) = K_p \cdot e(t) + K_i \cdot \int_0^t e(t)\mathrm{d}t + K_d \cdot \frac{\partial e(t)}{\partial t} \tag{2.10}$$

A theoretical calculation is used to set starting points for manually tuning the proportional constant and integral constant of the PID-control (the derivative constant is set to zero). The symmetric optimum method with critical damping has calculations as follows:

$$K_p = a \cdot \frac{J}{T_i} \tag{2.11}$$

$$T_i = a^2 \cdot T_s \tag{2.12}$$

$$K_i = K_p \cdot \frac{T_s}{T_i} \tag{2.13}$$

where T_i is the integral time parameter, T_s is the sampling time and a is the symmetrical distance between $1/T_i$ to the cutoff frequency, ω_0, and $1/T_s$ to ω_0.

Chapter 3

Test rig setup

This chapter describes the different components included in the test rig at Sibbhultsverken. Figure 3.1 below is a simplified drawing of how these components are connected in relation to each other. A picture of the real test rig can be seen in figure 3.4.

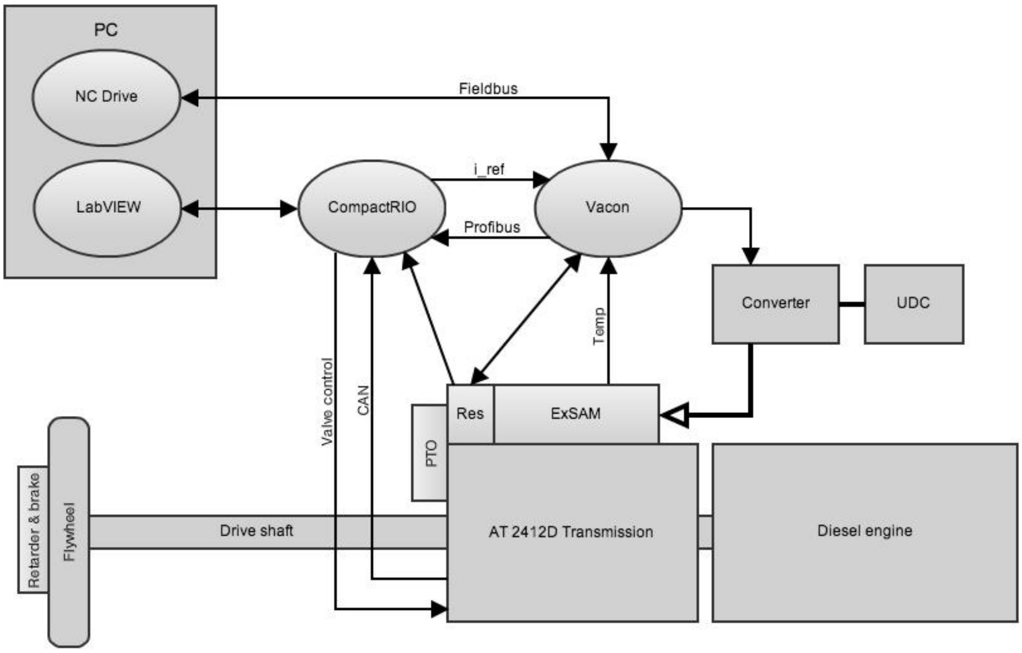

Figure 3.1: Project flowchart

The main parts that make up the test rig are as follows:

1. Flywheel on output shaft

2. Transmission AT2412D and electronic control unit

3. ExSAM
4. PTO-gear drive
5. Diesel engine
6. CompactRIO with extra chassis and modules
7. Vacon control unit
8. PC with LabVIEW and NC Drive

3.1 Flywheel on output shaft

The flywheel at the end of the driveshaft simulates the momentum of a vehicle in motion. This is important because it keeps the main shaft of the transmission at a relatively constant speed during the torque interruption in a shifting sequence as it would in a moving vehicle. The flywheel is not large enough to accurately replace the moment of inertia that an actual moving vehicle has, but it is sufficient in its purpose of maintaining speed on the output shaft during the short time that it takes to complete the gear shifting sequence.

3.2 Transmission AT2412D

The transmission used in this project is an AT2412D automated manual transmission from Volvo. It has synchronized split and range gears which together with the unsynchronized main gears can provide a total of 12 forward and 4 reverse gears. The transmission is equipped with an electronic control unit (TECU) with an advanced gear changing strategy already implemented. This presents possible communication problems as no access is allowed to the existing control, meaning its algorithms must either be overwritten or completely circumvented.

3.3 ExSAM

The ExSAM, an electrical machine specially designed by the supervisor of this thesis, Rasmus Andersson, is used to synchronize the countershaft in the gear shifting sequence. The machine has a peak torque of 250 Nm up to its base speed of 5 500 rpm and then a peak power of 140 kW up to 15 000 rpm as can be seen in figure 3.2.

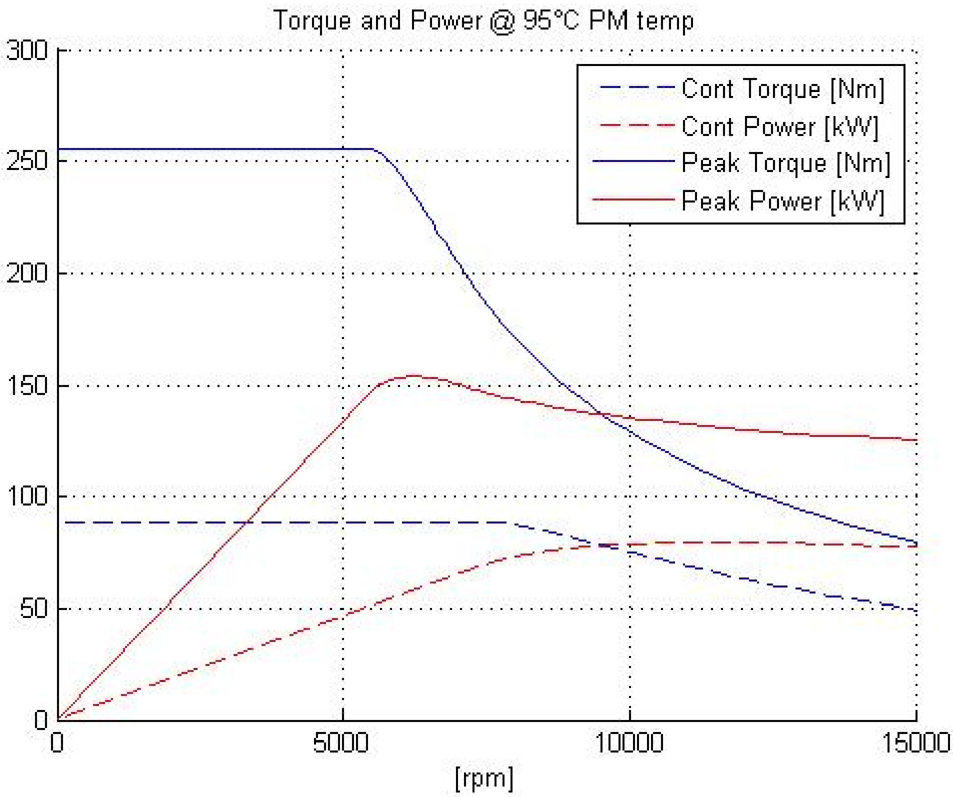

FIGURE 3.2: ExSAM characteristics

3.4 PTO-gear drive

A gear drive is custom made by SwePart Transmission AB to transfer the power from the electrical machine to the countershaft through the power take-off (PTO) on the AT2412D transmission. It consists of six cog wheels mounted on four shafts resulting in a transmission gear ratio of 5.3 : 1 between the ExSAM and the countershaft.

3.5 Diesel engine

The diesel engine is a Volvo TD71. It is controlled by changing the throttle between four discrete positions. This limits the engine to four discrete levels of fuel injection, which differs from the continuous control that a driver has while pushing down the gas pedal.

3.6 CompactRIO with extra chassis and modules

The CompactRIO Real-Time controller as well as a 4-slot expansion chassis and nine modules, all from National Instruments, comprise the main control setup. The system has an embedded controller that runs LabVIEW Real-Time as well as two separate FPGAs, one for each chassis. Below is a list of the components used:

- CompactRIO Real-Time controller (NI cRIO-9022)
- 4-slot Ethernet RIO expansion chassis (NI 9146)
- NI CompactRIO PROFIBUS Master/Slave Module
- 4-Channel, 100 kS/s, 16-Bit, 0 to 20 mA Analog Output Module (NI 9265)
- 4-Channel, 100 kS/s, 16-bit, 10 V Analog Output Module (NI 9263)
- 4-Channel, 100 kS/s/ch, 16-bit, 10 V Analog Input Module (NI 9215)
- 8-Channel, 5 V/TTL High-Speed Bidirectional Digital I/O Module (NI 9401)
- 2-Port, High-Speed CAN Module for NI CompactRIO (NI 9853)
- 8-Channel 5 to 30 V, 1 s, Sourcing Digital Output Module (NI 9474)
- 4-Channel, 100 RTD, 24-Bit Analog Input Module (NI 9217)
- 4-Channel LVTTL High-Speed Digital I/O Module (NI 9402)

Behind the CompactRIO is a chassis with circuits of relays that has been constructed by Getachew Darge at Lund University. The circuit serves as a means to step up the voltage of signals from the analog output module NI 9263 before they are sent to yet another relay control on the test rig. Both the chassis and the CompactRIO can be seen in figure 3.3.

3.6.1 NI CompactRIO PROFIBUS Master/Slave Module

As discussed in section 1.5 (Previous work) communication using the Profibus was originally bidirectional. However, in the scope of this thesis one-way only communication

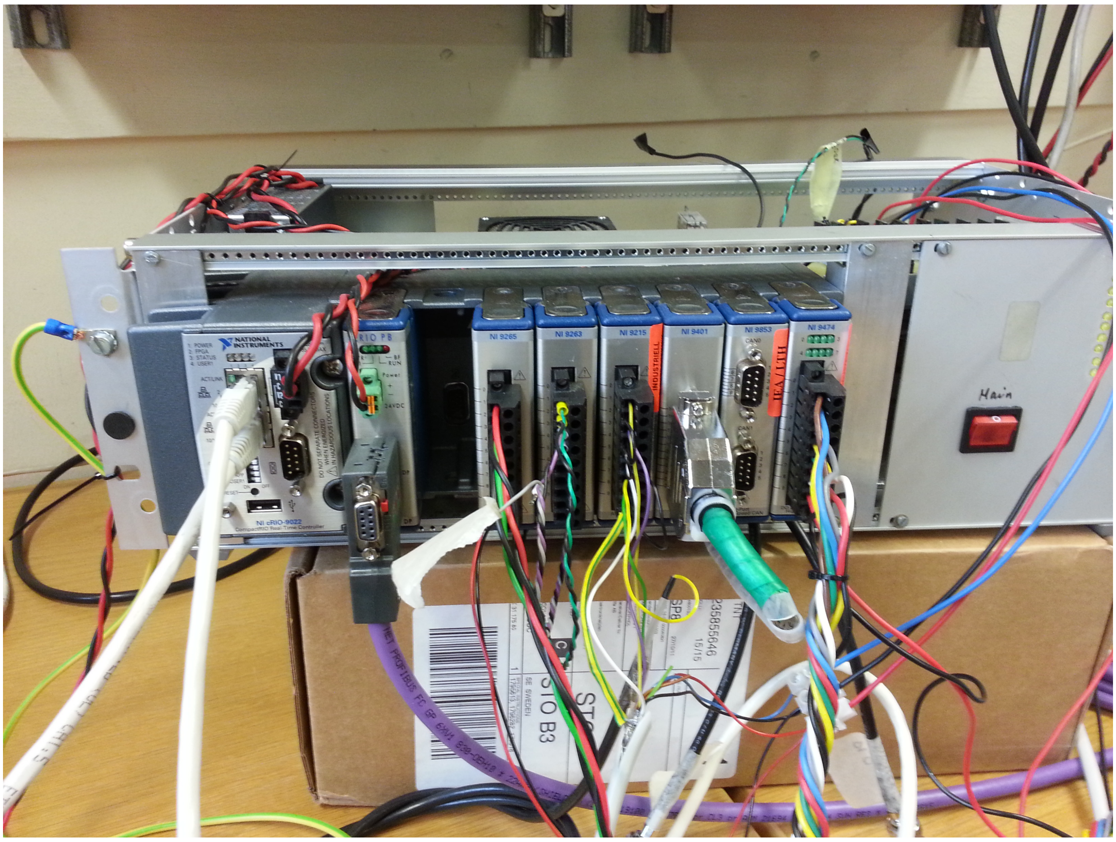

FIGURE 3.3: Frontview of CompactRIO

is implemented from the Vacon control unit to LabVIEW. Information that is less time critical such as U_{dc}, temperature information from the ExSAM, measured values for Is_x and Is_y, etc is sent via this link.

The CompactRIO PROFIBUS module requires 2.5 W of power, which is much higher than what the other modules require. One of the slots in the CompactRIO chassis is sacrificed in order to fulfill these power requirements. As a result the Profibus module is mounted in slot 1 and slot 2 is left empty.

3.6.2 4-Channel, 100 kS/s, 16-Bit, 0 to 20 mA Analog Output Module (NI 9265)

The NI 9265 is an analog output module used to send current references from a PID-controller in the LabVIEW program to the Vacon control unit. These current signals range from 0 to 20 mA and are interpreted by the Vacon control unit to determine the current injection to the ExSAM.

3.6.3 4-Channel, 100 kS/s, 16-bit, 10 V Analog Output Module (NI 9263)

The NI 9263 is an analog output module used for engaging and disengaging the clutch. This is achieved by applying 10 V to an air inlet or an air release valve for disengaging or engaging respectively. Although the module is in place, this setup is currently not used as dynamic engage and disengage of the clutch is not implemented. This module has previously been used to send voltage signals to the Vacon control unit, however this function has been replaced by the NI 9265 module. It can be argued that the current clutch control is more suited for digital signals as the discrete throttle levels are controlled with either 0 or 10 V. This has not been implemented due to the fact that the existing system, although not ideal, works and that no further effort is being made to realize a dynamic clutch control.

3.6.4 4-Channel, 100 kS/s/ch, 16-bit, 10 V Analog Input Module (NI 9215)

The NI 9215 receives the excitation signal and response signals directly from the resolver on the ExSAM. This information was originally communicated via the Profibus, however that method of speed feedback to the CompactRIO has been proven too slow prior to the commencement of this master thesis [3].

3.6.5 8 Channel, 5 V/TTL High-Speed Bidirectional Digital I/O Module (NI 9401)

The PTO-gear drive, diesel engine, retarder brake and flywheel brake are all switched on and off using digital signals from this output module via the adjoining relay circuit. The digital signals that control the discrete levels of fuel injection into the combustion chambers of the diesel engine are also sent from this module.

3.6.6 2-Port, High-Speed CAN Module for NI CompactRIO (NI 9853)

The NI 9853 module is used to receive CAN messages sent from the transmission control unit. The LabVIEW program receives every message sent via the CAN bus but only proceeds to decode the relevant messages pertaining to shaft speeds, oil temperature, shifting fork positions and the position of the clutch.

3.6.7 8-Channel 5 to 30 V, 1 s, Sourcing Digital Output Module (NI 9474)

This digital output module sends signals to the electronic control unit on the transmission to open and close the air valves that push the shifting forks between their engaged and neutral positions.

3.6.8 4-Channel, 100 RTD, 24-Bit Analog Input Module (NI 9217)

This module is used to read the analog temperature signals from four different locations in the ExSAM. It is situated in the extension chassis as this information is not deemed as time critical.

3.6.9 4-Channel LVTTL High-Speed Digital I/O Module (NI 9402)

The NI 9402 is used as an input only module and receives pulses generated by two separate encoders; one on the output shaft of the transmission and the other on the crank shaft of the diesel engine.

3.7 Vacon control unit

The power electronics that control the ExSAM is a VACON NXP standalone or NXC cabinet size-FR11 385 460 520 A rated at low loadability.

3.8 PC with LabVIEW and NC Drive

A PC with National Instruments' LabVIEW as well as Vacon's NC Drive are required to run the setup. These are described further in chapter 4 software.

Figure 3.4: The test rig

Chapter 4

Software

4.1 LabVIEW

The software used to program the FPGA and run the CompactRIO Real-time controller is National Instruments' LabVIEW. The program is a system design software with a graphical programming environment. Another benefit of LabVIEW is the parallel execution of code, which means that several different functions can run parallel to each other without waiting for allocated execution time. There are three different levels of execution that communicate with each other and allow for a comprehensive control environment:

- FPGA: FPGA stands for field-programmable gate array. It is able to complete complex digital computations with a clock frequency of 40 MHz. This is the fastest of the three execution levels as it operates on a microsecond level and is therefore used for all time critical functions.
- RT: The Real-Time program is slower than the FPGA but has room for extensive code and is also able handle floating-point calculations better. Input to the RT program comes via the FPGA.
- PC: The PC program also has a large code size capability. It has mainly been used to store and save data and display continuously updating graphs.[8]

4.2 NC Load and NC Application

NC Load and NC Application is software from Vacon for programming specific applications and downloading them to the Vacon control unit. PhD student Francisco Marquez of Lund University has created the code for the application that is used for the purpose of this thesis. The important mapping of LabVIEW reference currents to actual motor currents discussed further in section 5.3 is implemented in this application code.

4.3 NC Drive

The NC Drive programming tool is also from Vacon. A view of the motor parameters, an on/off control for the switching frequency and a monitoring graph can be seen in the three main windows in NC Drive: the *Parameter Window*, *Monitoring Window* and *Operating Window*.

Values for the parameters affecting the motor control such as maximum and minimum limits for the motor current and values for the motor's nominal current and voltage can be set in the *Parameter Window*. To achieve a stable motor control the speed control parameters in LabVIEW must be adjusted to match the range of Is_y set in NC Drive (see chapter 5). The parameters can be used to define the limits of the motor control. For example, allowing a higher maximum limit for the isy current injection alters how aggressive the control is. This is achieved by increasing the current limiting parameters: *P.2.1.5 current limit, P.3.8* Is_y *ref min and P.3.9* Is_y *ref max.*

The *Monitoring Window* shows selected parameters in a graph which is updated every 0.2 seconds. From here, an eye can be kept on key factors in order to avoid hazardous situations caused by, for example, pushing the limits for motor voltage or an unstable motor control. The graph can be paused and a zoom function is very useful when studying the details of the behavior of different parameters during a speed step.

The *Operating Window* controls the switching frequency of the converter with two buttons *start* and *stop*. This window proves very useful as an emergency stop function as it completely stops any current from being sent to the motor. In the case of an unstable motor control or other error this is the first line of defense for reacting to a rushing motor. However, stopping by completely shutting down all currents is not fault free as it can cause voltage surges and fuses to blow in the Vacon control unit.

Chapter 5

Method

5.1 Communication with the transmission

One of the first issues addressed is how to communicate between the transmission and LabVIEW. The original idea is to use the transmission's built in control unit (TECU) and overwrite its signals, a solution that requires no physical interference with the gearbox. Another benefit of using the TECU is the possibility to reuse and alter a pre-existing LabVIEW code from Volvo to implement the gear change. However, this idea must be abandoned due to technical and confidentiality matters. Instead an alternative solution where cables are connected directly to the air valves inside the gearbox is used. This allows direct control of the shifting forks from the CompactRIO. This only requires a minor alteration in the form of a hole on the top lid of the transmission and results in very fast control.

In order for the fork control to be meaningful it is necessary to be able to read the positions of the forks as well. This is achieved with the CAN protocol, J1939. A cable with proper terminations at the ends to avoid reflections together with a CANalyser device is used to help set up the CAN communication. Fork positions as well as shaft speeds and clutch position is read into LabVIEW via this link. It should also be possible to read the oil temperature in the transmission but since this is of no particular value for the project no extra effort has been spent on achieving this.

5.2 Communication with the ExSAM

The steering of the electric motor is done with power electronics from Vacon. A current reference signal is sent from LabVIEW through the NI-9265 module and is interpreted in the Vacon control unit, which in turn sets the current in the motor. The module can deliver a current between 0 mA and 20 mA which is mapped to a value between 0 and 10 000 in the control unit. This corresponds to the minimum negative motor current limit and maximum positive motor current limit respectively. For safety reasons explained in section 5.3.1 below all signals close to 0 are interpreted as 0 A instead of the negative motor current limit. The meaning of maximum and minimum current depends on the parameters set in NC Drive, for example *P.2.1.5* current limit and *P.3.8/P.3.9* Isy ref min/max. Because of several control levels in the Vacon program the current can be limited by many different parameters.

5.3 Vacon Programming

5.3.1 Is_y mapping

Since it is not possible to send out a negative current from a module in the CompactRIO a negative motor current must also be represented by a positive current reference from the module to the Vacon control unit. A natural way to do this to let 0 mA represent the maximum negative motor current and 20 mA represent the maximum positive motor current, creating a linear mapping from 0-20 mA (see figure 5.1). The problem with this solution is that if the module for some reason fails to deliver any current (due to a loose wire, power failure, etc.) the Vacon control unit interprets the absence of a reference signal as maximum negative current and the motor rushes in full reverse. To prevent this a safety zone is programmed in the Vacon control unit that tells it to interpret all signals below 1 000, the equivalent of 2 mA, to zero motor current. At 1 000 the control unit is set to interpret the signal as maximum negative current which then decreases down to zero motor current at 5 500 before it starts to increase as positive motor current up to 10 000 (see figure 5.2). That is, the linear mapping is set between 1 000 and 10 000 instead of 0 and 10 000. The lower limit is chosen arbitrarily to 1 000 since it gives an even loss of 10 % in resolution.

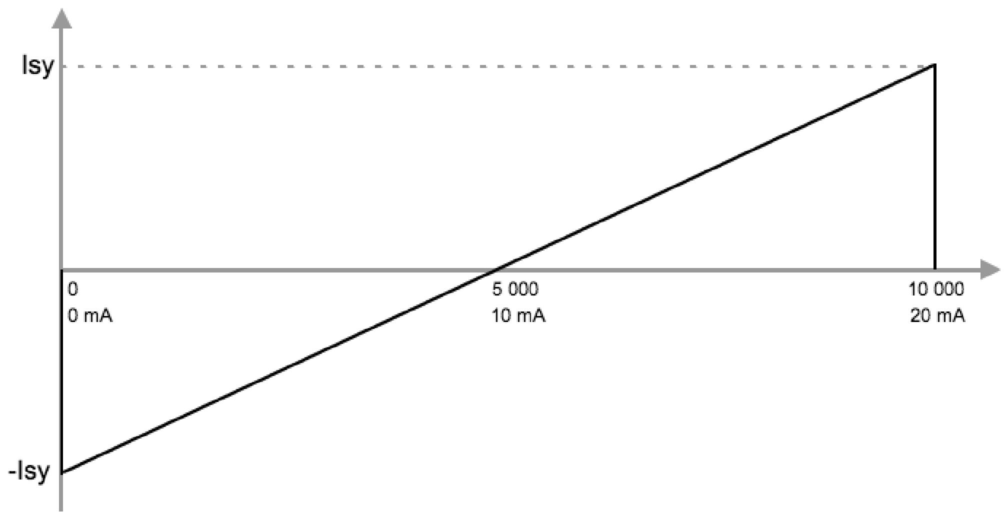

FIGURE 5.1: Simple Is_y current mapping

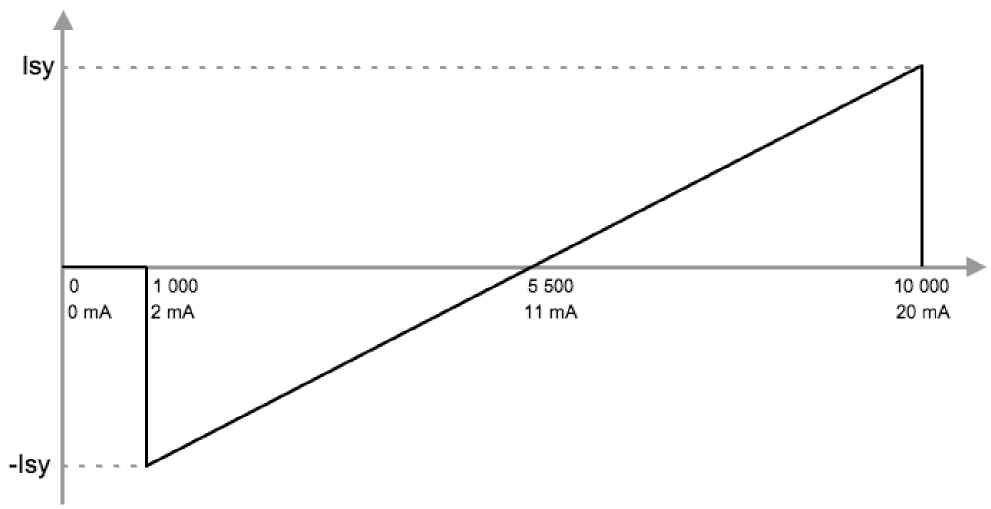

FIGURE 5.2: Is_y current mapping with safety zone

While testing the safety margin by setting a reference signal very close to 2 mA the whole test rig started to shake wildly as signal noise made the reference signal jump back and forth across the 2 mA boarder. This resulted in the motor switching between zero and maximum negative current. In order to solve this, the safety zone in the Vacon control unit is reduced further down to 700 the equivalent of 1.4 mA. By letting the reference signal from LabVIEW saturate at 2 mA a dead zone is established at minimum motor current between 700 and 1 000 in the Vacon control unit (see figure 5.3). This eliminates the risk of having a reference signal ending up right on the boarder of a discrete speed jump, with the added safety of zero motor current in the event of reference signal loss.

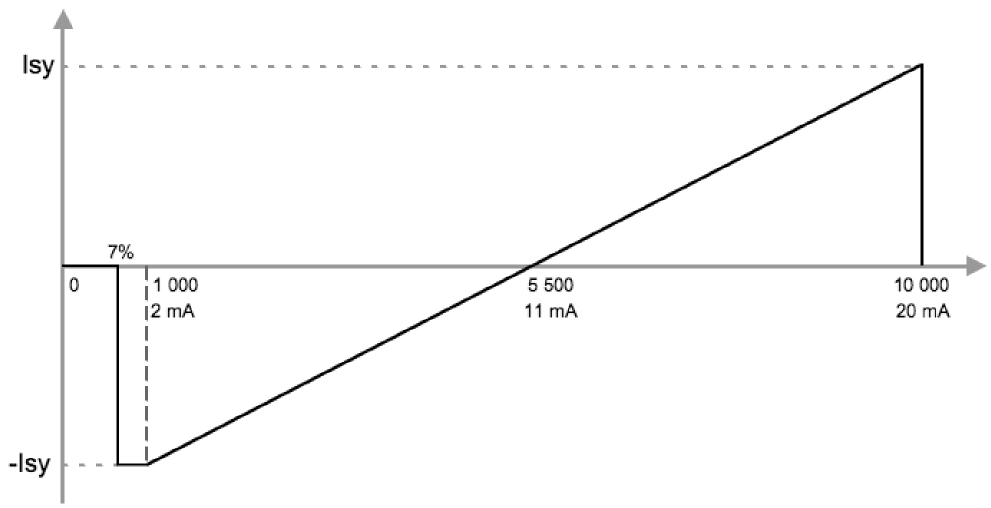

FIGURE 5.3: Is_y current mapping with dead zone

5.3.2 Is_x mapping

Unlike the Is_y current the Is_x current only needs to be negative since field weakening is always a negative current. This simplifies the mapping of the Is_x current especially since Vacon are also aware of this and therefore has implemented their control unit to interpret a positive signal as a negative current. The mapping is therefore simply a linear function from 0 mA to 20 mA (see figure 5.4)

5.3.3 Overspeed protection

In the early testing stage of the ExSAM at Sibbhultsverken a major fault occurred that caused a lot of damage to the equipment [3, p. 38]. The induced back emf voltage had become too high rendering the motor uncontrollable. Once control was lost the motor proceeded to spin even faster inducing an even higher back emf resulting in current going backwards into the power converter with catastrophic results. Since induced voltage is proportional to speed this type of fault can be avoided by introducing an over speed

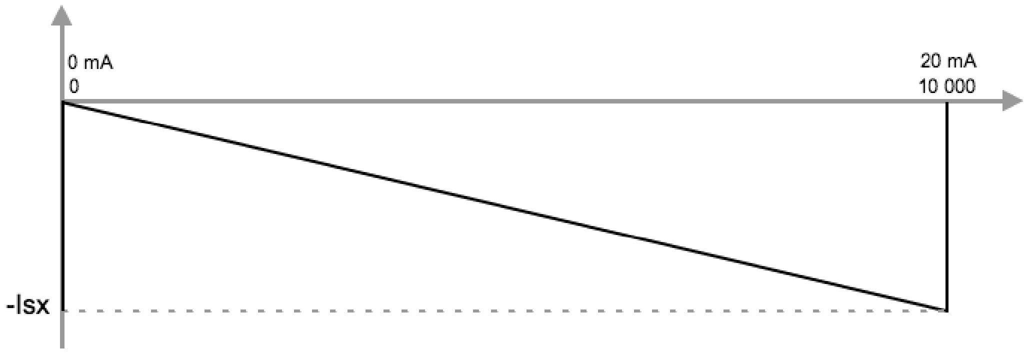

FIGURE 5.4: Is_x current mapping

protection. This is implemented in NC Drive with the help of PhD student Francisco Marquez, controllable by parameter setting *P.3.12*.

5.4 Gear sequence

The LabVIEW code seen in appendix A provides the core of this thesis as it is the implementation of the shift sequence described in section 2.1. It is here that the execution of a dynamic gear change takes place. As the main goal of the thesis is to reduce the time for a gear shift, it is crucial that everything in this code is executed as fast as possible and it is therefore placed on the FPGA level of LabVIEW. Two different approaches to this code have been studied. The first is to complete a gear change by using linear execution of every step in a sequence of code. However, this solution is far too cumbersome which paves way for the second coding approach which is to create a state machine that allows jumps between different states. The state machine is far superior as it opens for the possibility of immediate and specific error handling. The different states in the state machine can be seen in figure 5.5 and are explained below:

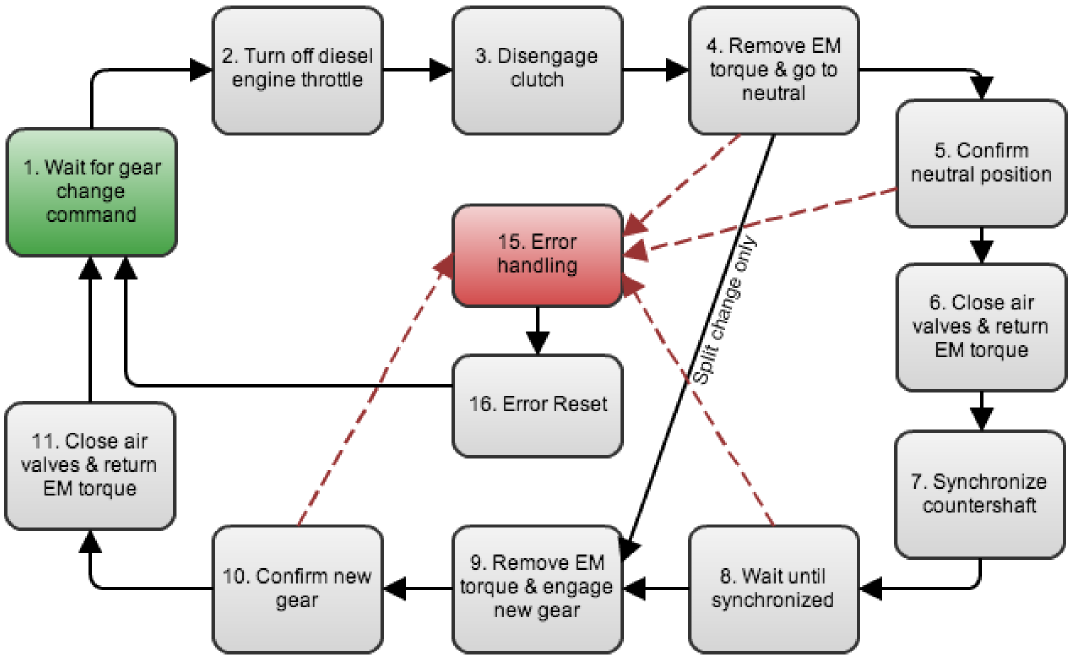

Figure 5.5: Flowchart of the state machine

1. This is the starting state; the state machine idles here until a new gear is selected and confirmed by the *Action* variable (see appendix C). In order for the gear change to move on to the next state a clearance is also needed that ensures that the requested gear will not lead to speeds that exceed the operating limits of either the diesel engine or the electric machine. The clearance signal also prohibits gear changes that would result in an unreasonably large speed step for the electric machine.

2. Turn off throttle; preparation state that turns off the throttle to the diesel engine once it is incorporated into the project. Currently not in use.

3. Disengage the clutch; sends a disengage signal to the clutch and waits until the clutch position exceeds the disengage limit or a timeout has been reached. In the latter case an error signal is generated, *Unable to disengage clutch*. Currently not in use.

4. Go to neutral; here the present gear and the requested gear are checked whereupon the proper action is taken to achieve a neutral state, including setting the torque to zero. In the case of a split change a jump is made to state 9 (since the split gear is synchronised).

5. Neutral control; checks that a neutral state has been achieved by both the main gears and the split gear. If not, a timeout generates an error for each set of gears, either *Main gears unable to go to neutral* or *Split unable to go to neutral*.

6. Close valves; closes the air valves and returns the torque control to the speed regulator.

7. New speed; sets a new speed demand based on the new gear ratio and an added overshoot.

8. New speed check; checks that the new speed is achieved or a timeout has been reached which generates an error, *Synch error*. If successful the motor torque is set to zero to allow a new gear to be engaged. As there is no torque the motor loses momentum faster than the flywheel on the output shaft. This is the reason why the set speed needs an extra overshoot from which it can fall through a speed window where a gear change is possible.

9. New gear; opens the valves to engage the new gear.

10. New gear acknowledgement; checks that the new gear is engaged or waits until a timeout has been reached which generates an error, *Unable to engage gear (timeout)*. Initially there was also an error, *Unable to engage gear (outside speed window)*, that aborted the change sequence if the speed sunk outside the allowed gear changing window. However due to the slow acknowledgement signal this error is sometimes erroneously generated and to avoid an abortion in such cases the error handling is altered to merely a warning indicator.

11. Final gear changing state; closes all the valves and returns control to the speed regulator which sets a new speed that matches the speed of the flywheel on the output shaft. If the gear change has been successful the state machine then returns to state 1 to await a new command.

15. Error state; this state handles any error that might have occurred in any previous state except the *Unable to engage gear (outside speed window)* error as explained in state 10. All errors are dealt with by shutting off the valves that move the gear shifting forks and setting the motor torque to zero. A control is also set that constantly resets the integral part of the speed controller to zero, this to avoid any unexpected build up of the integral part (if it were to be used). The only error

that entails specific handling is the *Unable to disengage clutch* that needs an extra control to turn off the disengage clutch signal. In order to leave this state and be able to take back control over the motor torque and speed controller again a button in the front panel labelled *Reset errors* has to be pushed which makes the state machine to move on to the next state.

16. Reset errors; turns off all the error indicators then moves on to state 1 again to wait for a new gear change command.

5.5 Motor control

An important aspect of achieving a fast dynamic gear change is a fast and stable motor speed control. It is this control that decides how fast the shaft synchronisation is, the phase of the gear change that is most crucial to this thesis. To avoid having a stationary error it is conventional to have both a proportional and integral part in the PID-controller of the countershaft speed.

The symmetric optimum method for calculating the proportional and integral constants is used. A sampling time of $100\mu s$ and the total inertia of the ExSAM together with the PTO-gear drive and the countershaft, $J = 0.0434kgm^2$, results in $K_p = 144.7$ and $K_i = 1.61$. The PID-controller in LabVIEW uses a fixed point notation that cannot be altered and has as a result a maximum limit just under 128 for K_p. A good rule of thumb for tuning a PID-controller is to start with parameters much smaller than the calculated values in order to avoid an unstable outcome. A heuristic method is applied to the tuning process, starting with low current limits. During the testing it has been found that the control is both faster and more stable without the integral part and therefore the original PID-controller is replaced by a pure P-controller. The downside of this improved speed and stability is a small but acceptable stationary error. With the motor control working satisfyingly the current limits are slowly increased up to 200 A. During this current increase the proportional parameter is reduced accordingly since the power electronics can deliver more current with higher limits. For instance, a doubled current limit means twice as much current with the same proportional gain, which in turn means that the gain has to be halved to keep the controller stable.

5.6 Testing at Lund University

To reduce the amount of time spent at the full scale test rig a basic test rig is set up at Lund University. It is comprised of power electronics, a small flywheel on the output shaft and an electric machine and transmission similar to those in the full scale rig.

This rig proved very valuable for the initial testing phase; setting up the communication, basic tuning of the motor control, mapping the currents and trying out the gear changing sequence up to the point of engaging the new gear. However the smaller flywheel does not have enough momentum to keep the speed constant during a gear change and hence an actual gear change cannot be performed with this set up. A larger flywheel is not an alternative due to safety reasons in the laboratory. The DC-link is only about 200 V and the power converter lacks cooling, which means that both the speed and the torque must to be kept relatively low when using this set up.

5.7 Dynamic tests at Sibbhultsverken

With the small scale testing sufficiently completed the next step is full scale testing at Sibbhultsverken. This includes disconnecting and transporting much of the equipment from Lund. Before commencing any tests all connections and safety features as well as motor control must thus be retested to make sure that they work as expected. Once the initial setup phase is completed the actual tests can commence. As mentioned earlier in the limitations section 1.4 the main focus of the testing is to find out the time it takes to perform gear changes of the main gears, more specifically between $1st$, $3rd$ and $5th$ gear. This could just as well have been $2nd$, $4th$ and $6th$ as it is just a matter of high or low split. The testing consists of four different parts:

1. Gear change with fixed output speed
2. Acceleration times
3. Simulated drive cycle with acceleration and deceleration
4. Fast gear change with low safety

5.7.1 Gear change with fixed output speed

Being the first of the dynamic tests the main focus is put towards simplicity and clear results. To make sure never to exceed the base speed level of the ExSAM the test always starts at maximum test speed with the transmission in $1st$ gear. From $1st$ gear a change to $3rd$ is made, resulting in a lower motor speed, and then another change to $5th$ gear. The whole sequence is then reversed back down to $1st$ gear again with increasing motor speed for every change. However, due to a small loss of momentum during every torque interruption the original speed is never exceeded. To make sure that the gear changing works as it would in a normal vehicle, tests are also made going directly from $1st$ gear to $5th$ and back down to $1st$ again. These tests are first performed with 150 Hz as the starting speed but this is eventually increased to 250 Hz. To go any higher would require the use of look up tables since 250 Hz equals 5 000 rpm and the base speed of the motor is 5 500 rpm.

5.7.2 Acceleration times

As expected the CAN communication proves to be a limiting factor. To get a theoretical guideline to the optimal speed of a gear change a series of tests are performed that just test the time of a speed steep in the ExSAM. This corresponds to the theoretical minimum time any gear change can take. The test consists of two parts; a 100 Hz step from 20 Hz to 120 Hz and then down again, and a sequence of steps that correspond to the speed changes required during the fixed output speed tests.

5.7.3 Simulated drive cycle with acceleration and deceleration

This test simulates an acceleration and deceleration of a truck more realistically than the fixed output speed test. Unlike the other tests that are soley programmed in the FPGA this test is programmed in the Real-Time environment of LabVIEW but uses the controls programmed in the FPGA. It starts with the transmission in $1st$ gear at a standstill then accelerates up to 250 Hz where it changes into $3rd$ gear. By shifting gear in the transmission the motor speed is reduced and the output speed is kept constant. Immediately after the gear change is completed the motor speed is accelerated up to 250 Hz again and a new gear change takes place to $5th$ gear. This last gear change is followed by a final acceleration before the sequence is reversed back down to 1*st* gear again, this time decelerating the electric machine down to 100 Hz between changes.

The results of this sequence as seen from the output shaft is a steady acceleration during the changes from 1*st* through to 5*th* gear followed by a steady deceleration during the changes from 5*th* back down to 1*st* gear. The optimal operating range for economical driving of a typical diesel engine used in Volvo trucks lies between approximately 1 000 and 1 400 rpm (see figure 5.6). During this test the electric machine range is between 250 Hz and 100 Hz. This range results in convenient steps for the electric machine and is determined keeping in mind the range for economical driving of the diesel engine. The implemented ExSAM speed limits 100 Hz to 250 Hz correspond to 540 rpm and 1 340 rpm for the diesel engine respectively. This is slightly lower than the economical driving limits. The fact that it is not possible to run the electric motor above its base speed without lookup tables does not limit this test. Higher speeds on the electric machine would mean operating outside the economical driving speed limits for the diesel engine. Reaching the maximum limit for the economical driving of the diesel engine, 1 400 rpm, would require the electric machine to run at 261 Hz. Without lookup tables it is still possible to run at 250 Hz meaning the tests are relevant despite the lack of lookup tables. Based on the range for economical driving the fastest speed the electric machine should have to run at is 337 Hz (6 740 rpm) which corresponds to 1 400 rpm on the diesel engine via the high split gear ratio. This would require lookup tables as it results in 1 240 rpm higher than the base speed.

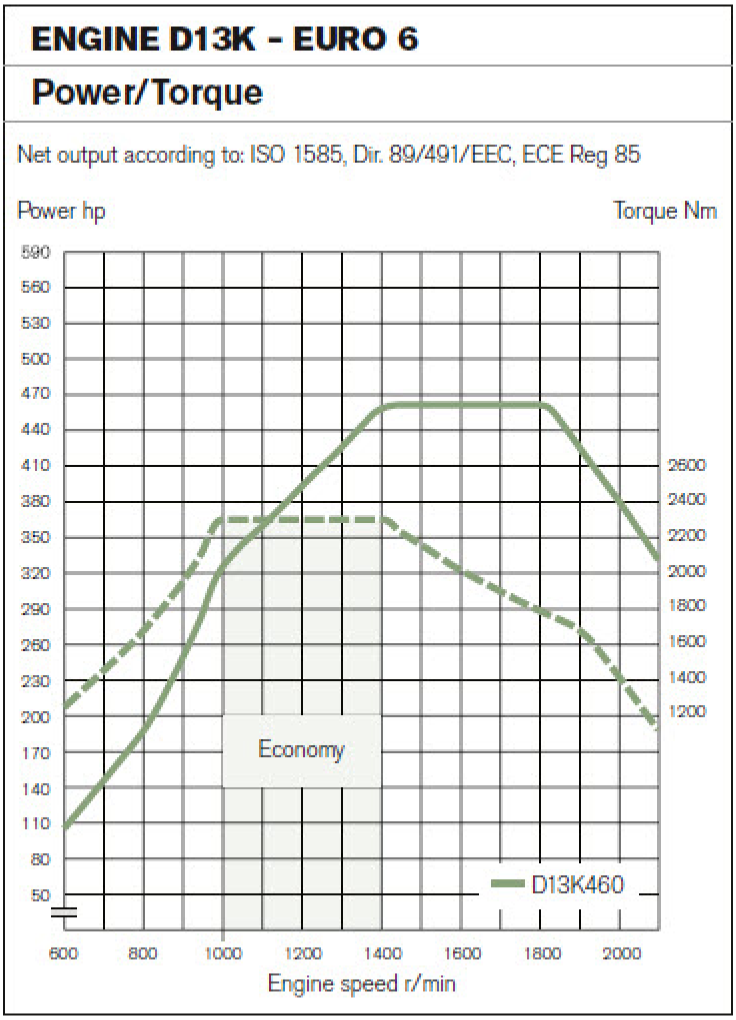

Figure 5.6: Power/Torque (Source: Volvotrucks [9])

5.7.4 Fast gear change with low safety

The final set of testing is purely experimental with the purpose to show the possible potential of the ExSAM. The main difference between this test and the previous ones is that the state machine is reprogrammed to move on after a 20 ms timeout when going into neutral and a 40 ms timeout when engaging a new gear. This is instead of waiting for acknowledgements from the CAN-bus. This means that the safety features checking for errors are disabled and it is therefore not recommended to run in this mode for extended periods of time.

Chapter 6

Results

When evaluating the results it is important to keep in mind that this thesis merely aspires to prove a concept. This has a major impact on the consistency level of the results. It is not necessary to prove a result repeatedly to confirm its validity. The fact that a result has been proven once is enough to show that it is possible accomplish and hence verifies the concept.

6.1 Measuring the results

It is important to be consistent when measuring the results. Values for the relevant variables are sampled at a rate of 10 000 samples/s and plotted via a script in Matlab. Since it is the torque interruption that is of relevance the *No torque* signal is used as the measuring mark, seen in figure 6.1 as magenta. The time of the torque interruption is measured from when the signal is turned on to when it is turned off the second time and the gear change is completed. The time in the middle where the signal is turned off is where the synchronisation takes place. This measurement does not measure the entire gear changing sequence described in section 5.4, only state four to eleven. The difference though is rather small as state one should not be included since it is the waiting state and state two and three are not in use. Figure 6.1 below is of a gear change from $3rd$ to $5th$ gear. The green graph is the speed reference signal sent to the PID-controller by the shifting sequence in LabVIEW, the blue graph is the actual speed of the ExSAM from the resolver and the red graph is the same resolver signal sent via the Vacon control unit and Profibus. The difference between the red Profibus signal and blue resolver signal shows the delay in communication discussed in section 1.5.

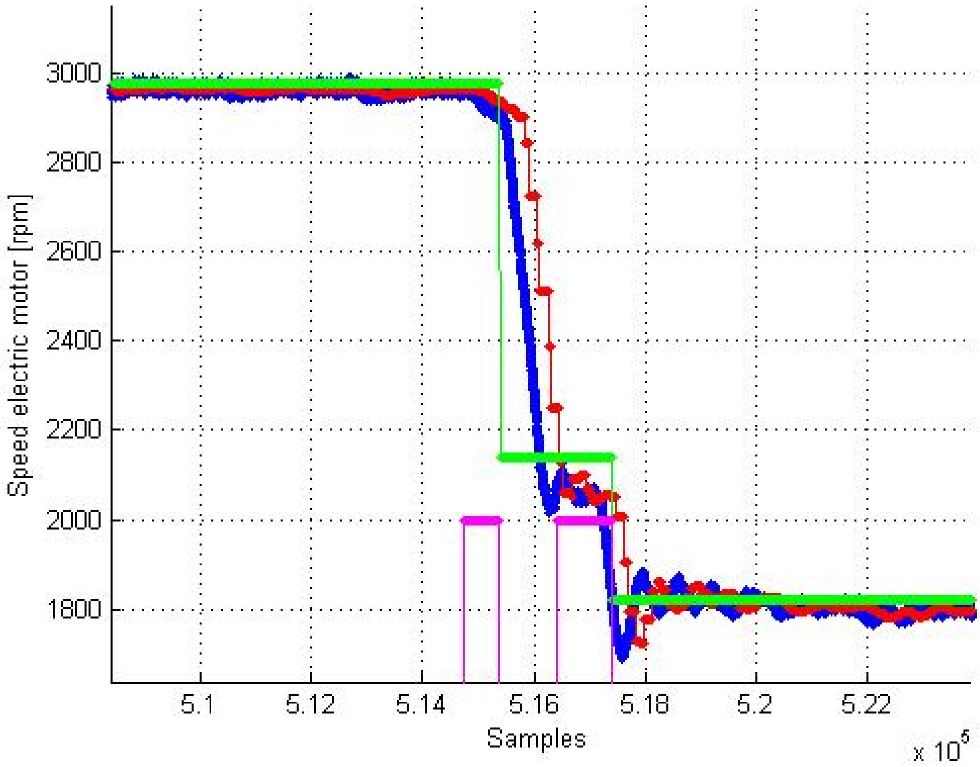

FIGURE 6.1: Gear change from 3rd to 5th with a fixed output speed starting at 250 Hz

6.2 Tests

6.2.1 Gear change with fixed output speed

Figure 6.2 shows a typical example of the fixed output speed test. It starts in $1st$ gear and the speed is first set to 1 000 rpm (50 Hz) to make sure that everything is working as intended before increasing to 5 000 rpm (250 Hz). Once at 5 000 rpm the gear changing begins by shifting to $3rd$ gear resulting in a motor speed of approximately 3 000 rpm (150 Hz). The following speed step to approximately 1 800 rpm (90 Hz) is the result of shifting from $3rd$ to $5th$. The shifting sequence is then reversed with an increasing motor speed in two steps. As the transmission gears down the overshoot explained in section 5.4 (State 8) can be seen as the spikes in the beginning of the step. The observant reader might notice that the title of this test is somewhat misleading since the finishing speed is slightly lower than the starting speed. This is due to the speed lost by the flywheel during the torque interruptions.

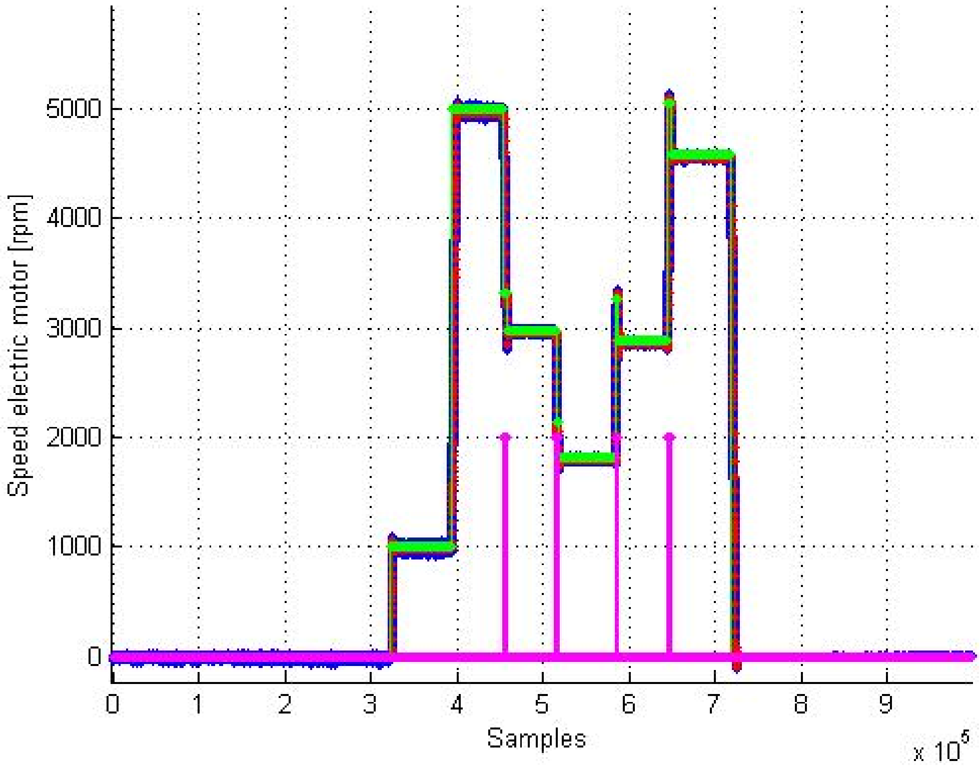

FIGURE 6.2: Gear changes with fixed output speed starting at 250 Hz with a sampling rate of 10 000 samples/second

Table 6.1 shows the time in milliseconds for gear changes with the same structure as the graph above. The column labelled *Starting speed* is the speed from which the first gear change (from $1st$ to $3rd$) takes place. The drive cycle then steps through each gear without adjusting the speed in between changes. It can be seen that the lower the starting frequency the faster the shifting times. This is because the relative speed step is determined by the actual speed at time of shifting, a higher speed means a larger speed step is required before the new gear can be engaged. Tests 12 and 13 show repetitions of the same shifting steps. This is to see the effects of varying time delays in the code execution and increase the chances of getting a fast result at least once.

If test 5 is compared with the results of test 12 it can be seen that the times are significantly better in test 5. The average of test 12 for the $1st$ to $3rd$ gear shift is 356 ms which is 50 ms higher than the result in test 5. Even more apparent is the shift from $3rd$ to $1st$ where the results differ by 100 ms between the tests. Both tests are done

Table 6.1: Results of gear change with fixed output speed

Test No.	$\lvert Is_y \rvert$ [A]	K_p	Starting speed [Hz]	1st-3rd [ms]	3rd-5th [ms]	5th-3rd [ms]	3rd-1st [ms]	1st-5th [ms]	5th-1st [ms]
1	180	17	150	260	210	240	310	–	–
2	60	40	150	200	180	280	350	–	–
3	120	25	150	204	200	220	250	–	–
4	120	25	150	248	240	230	250	–	–
5	120	25	250	305	194	250	350	–	–
6	180	17	150	264	240	270	270	260	280
7	180	17	100	–	–	–	–	–	–
8	200	15.3	250	400	270	360	380	–	–
9	200	15.3	250	390	270	320	410	–	–
10	200	15.3	150	–	–	–	–	370	460
11	200	15.3	250	370	–	–	472	–	–
12	200	15.3	250	360	–	–	476	–	–
				360	–	–	395	–	–
				380	–	–	539	–	–
				322	–	–	462	–	–
				318	–	–	470	–	–
				370	–	–	377	–	–
				361	–	–	382	–	–
				384	–	–	444	–	–
				336	–	–	448	–	–
13	200	15.3	250	–	367	387	–	–	–
				–	314	388	–	–	–
				–	336	357	–	–	–
				–	311	366	–	–	–
				–	402	373	–	–	–
				–	370	389	–	–	–
				–	463	392	–	–	–
				–	337	409	–	–	–
				–	307	407	–	–	–
				–	316	361	–	–	–

from the same starting frequency but test 5 has a lower maximum Is_y and subsequently a higher K_p for the control. The aim is to utilize as close to all of the nominal current of the ExSAM meaning that 200 A is the better option. However, as test 5 achieves better overall results despite running with 120 A it can be determined that the PID-controller is tuned slightly more aggressively in this case than in the tests using 200 A. A more aggressive PID-controller comes at the cost of a more unstable motor control but it remains uncertain whether or not optimal values for the PID-parameters have been chosen. Since the main purpose of the thesis is to prove the concept rather than optimizing the solution no more effort has been put on the control parameters.

As can be seen in table 6.1 the results vary significantly even between tests with the same conditions. Closer analysis reveals that the time between the *No torque* signals, that is during the synchronization of the countershaft is quite constant. To confirm this, tests are done focusing only on the time it takes for certain speed steps to be completed by the motor with the transmission in neutral as it would be in a real shifting sequence. These tests are explained further in the following subsections.

6.2.2 Acceleration times

This test simulates the speed steps during a gear changing sequence in the fixed output speed test described above. By comparing figure 6.2 and figure 6.3 it is clear that the speed steps in these tests are very similar. Noting only the acceleration time results in an interesting reference value that reveals how much of the time it takes to change gears is spent on synchronisation and how much is spent on shifting forks, waiting for acknowledgements etc. Since the *No torque* signal is not in use in this test the time measurement is done in an alternative way. The starting point is obviously chosen to be the change in the speed reference signal and the end point of the measurement is chosen to be when the speed signal first crosses the reference signal. The results from these tests are presented in table 6.2.

Table 6.2: Results of acceleration times tests

Test No.	$\lvert Is_y \rvert$ [A]	K_p	20-120Hz [ms]	120-20Hz [ms]	250-150Hz [ms]	150-90Hz [ms]	90-145Hz [ms]	145-230Hz [ms]
14	20	128	1140	600	530	350	700	1170
15	100	30	180	170	160	110	120	160
16	180	17	180	160	160	100	120	160
17	200	15.3	180	160	160	110	120	160

As can be seen in table 6.2 test 14 shows extremely slow values compared to the other tests. As mentioned in section 5.5 the PID-parameters are limited in LabVIEW. The upper boundary of the P-coefficient (128) is limiting the setup when the maximum value of Is_y is set as low as 20 A. If the tuning of the PID controller for 100 A is used to mathematically calculate the appropriate P-coefficient for 20 A then it seems it would need to be as large as 150 to achieve a similar control. As this is impossible the resulting control is much slower than the others. Tests 15 to 17 however support the findings in

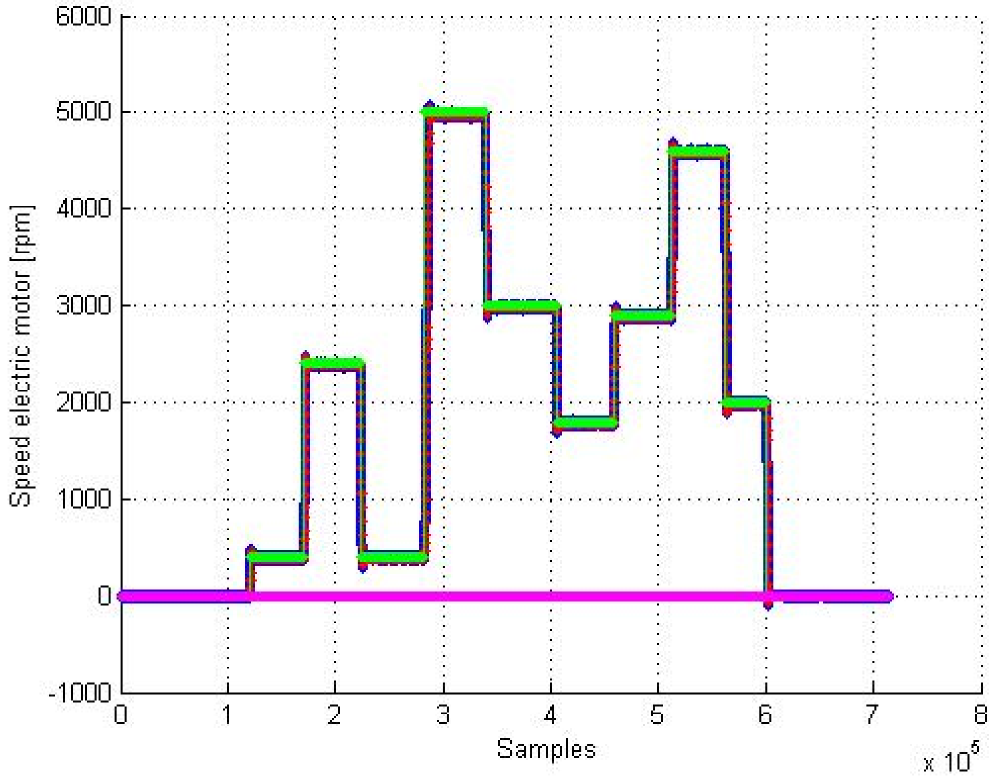

FIGURE 6.3: Acceleration sequence without gear change

the previous section that the acceleration times are fairly constant, showing times as fast as 100 ms in the case of shifting between $3rd$ and $5th$ gear.

Since the acceleration can be seen as fairly linear these values can be recalculated and extrapolated to rpm/s on the input shaft (see equations 6.1, 6.2, 6.3, 6.4). This gives an important reference value that can be compared to the one in the Objective section 1.3 in chapter 1. The time in test 17 for the speed step from 120 Hz to 20 Hz is used in the following calculations:

$$100\ Hz \cdot \frac{60\ s}{3\ pole\ pairs} = 2000\ rpm_{em} \tag{6.1}$$

$$\frac{2000\ rpm_{em}}{5.3\ (PTO\ gear\ ratio)} = 377.4\ rpm_{cs} \tag{6.2}$$

$$377.4\ rpm_{cs} \cdot \frac{44}{31} = 535.6\ rpm_{is} \tag{6.3}$$

$$\frac{535.6\ rpm_{is}}{0.160\ s} = 3347.5\ rpm/s \tag{6.4}$$

This gives a speed that is more than 1 000 rpm faster per second resulting in an improvment of:

$$\frac{3347.5\ rpm/s}{2000\ rpm/s} = 1.674 = 67.4\ \% \tag{6.5}$$

The main objective of the thesis is to improve the torque interruption time caused by long acceleration times. With the significant improvement of 67 % noted it is clear that synchronisation times are improved with ExSAM.

6.2.3 Simulated drive cycle with acceleration and deceleration

This test shows the acceleration up to 5 000 rpm in $1st$, $3rd$ and $5th$ gear, followed by a deceleration to 2 000 rpm in $5th$, $3rd$ and $1st$ gear. The test sequence can be seen in figure 6.4 and the time results are presented in table 6.3.

TABLE 6.3: Results of simulated drive cycle with acceleration and deceleration

Test No.	$\|Is_y\|$ [A]	K_p	Starting speed [Hz]	1st-3rd [ms]	3rd-5th [ms]	5th-3rd [ms]	3rd-1st [ms]
18	200	15.3	200-120 / 100-160	390	270	400	340
19	200	15.3	200-120 / 100-160	350	280	350	460
20	200	15.3	200-120 / 100-160	370	350	410	360

The times recorded during the drive cycle are consistent with the shifting times recorded during the fixed output speed test. The purpose of this test is mainly focused on providing a slightly more realistic demonstration of how and when the gearshifts would take place in an actual drive cycle. As the speed is accelerated up to 250 Hz between the changes it is reasonable to assume slightly longer gear changing times for the shifts between $3rd$ and $5th$ gear, which have preciously been done from 150 Hz.

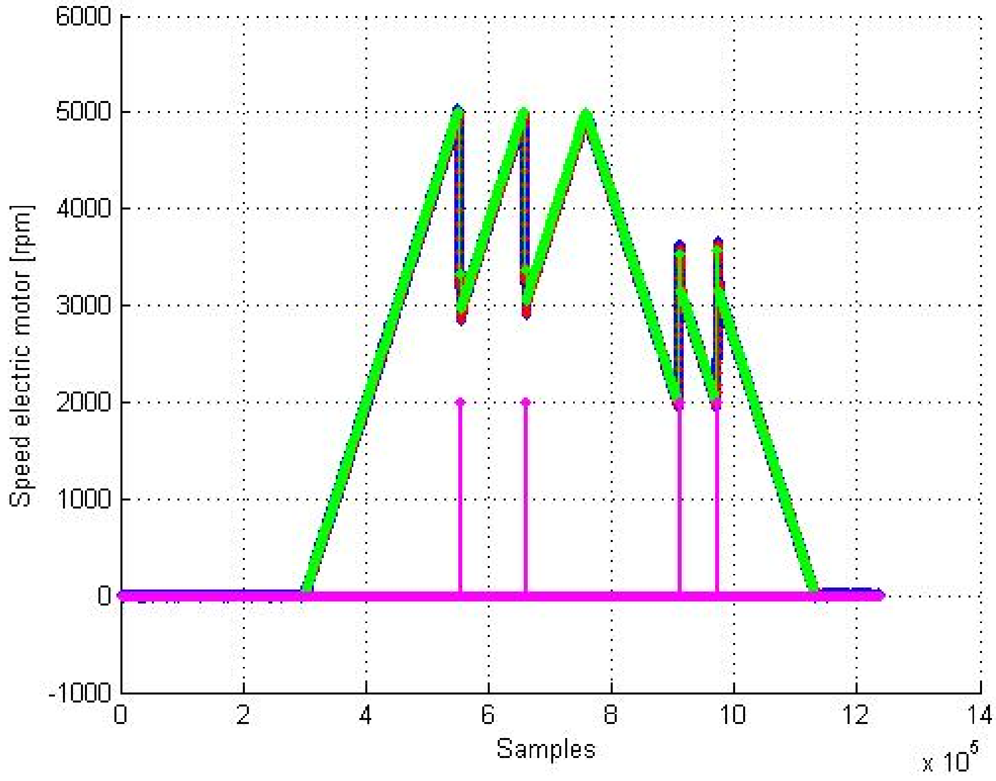

FIGURE 6.4: Gear change in simulated drive cycle

6.2.4 Fast gear change with low transmission safety

This test is essentially the same as the fixed output speed test but with manually shortened torque interruptions. Timeouts were originally set to 20 ms for each of the torque interruptions in the shifting sequence. The fast gear change is only repeated four times in order to not risk damaging the gearbox. As the first test sequence results in several unsuccessful gear shifts, the second torque interruption is increased from 20 to 40 ms. After increasing the timeout the tests immediately deliver vastly improved results. Repeating the previous tests it can be seen that even under the same circumstances the gear changing times vary significantly. By measuring the speed step only and finding very consistent results it can once again be settled that a large portion of the torque interruptions is spent waiting for position acknowledgements from CAN. Manually limiting the waiting time with a timeout control results in shifting times near half has long as with the original sequence. The fastest recorded shifting time from the starting frequency 250 Hz is 220 ms during a shift from $1st$ to $3rd$ gear. This is nearly 100 ms faster than the fastest result in test 12. The graph in figure 6.5 shows the results of test

23, which is the fixed output test using the low safety sequence. All measurements from the low safety test sequence are presented in table 6.4.

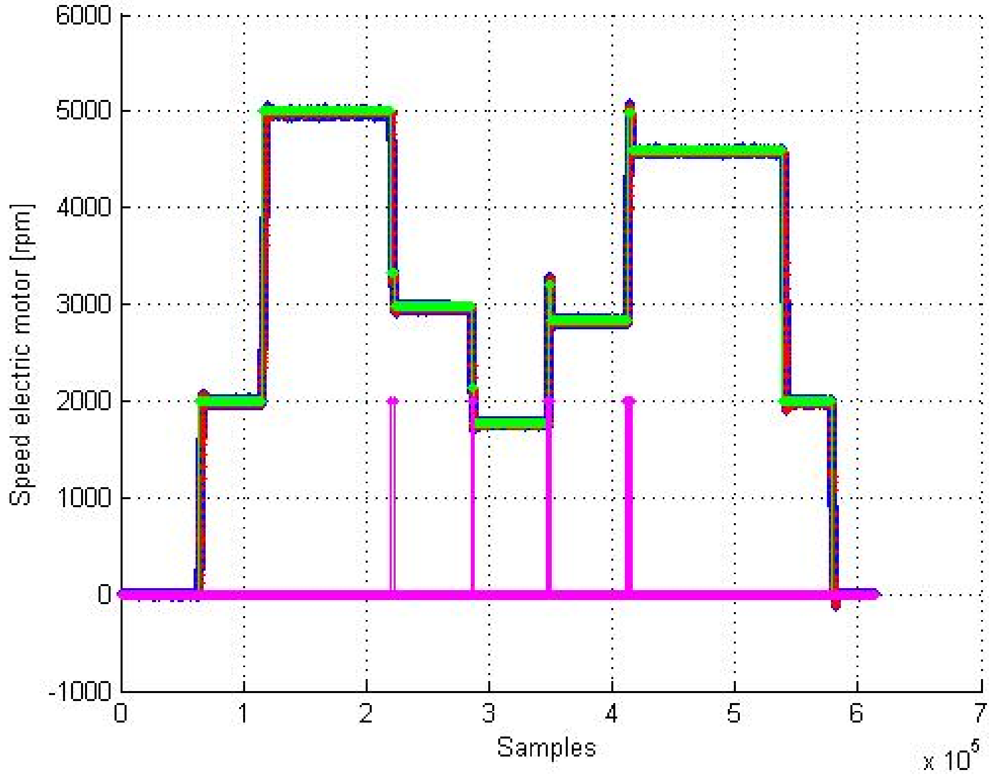

Figure 6.5: Fast gear changing sequence

Table 6.4: Results of fast gear change with low safety

Test No.	$\lvert Is_y \rvert$ [A]	K_p	Starting speed [Hz]	Timeout [ms]	1st-3rd [ms]	3rd-5th [ms]	5th-3rd [ms]	3rd-1st [ms]
21	50	50	150	20	192	150	250	330
22	200	15.3	250	20	250	–	–	–
23	200	15.3	250	40	270	160	220	330
24	200	15.3	250	40	220	220	280	–

To get an idea of the time is spent waiting for acknowledgement signals via the CAN-bus the longest time for the gearshift between $3rd$ and $5th$, 270 ms from test 9, can be compared to the fastest time of 160 ms, from test 23. The time of the actual speed step for this gear change can be seen in the acceleration times test as 100 ms, which is consistent with the result of 160 ms as the sum of the two timeouts add up to 60 ms; 20 ms + 100 ms + 40 ms = 160 ms. This means that the time it takes to actually move the shifting forks and engage a gear takes less than 40 ms. The exact amount of time it takes can however not be concluded from the tests performed. By subtracting the synchronization time and timeouts from 270 ms it can be seen that in test 9 up to 110 ms is wasted waiting for acknowledgement signals.

Chapter 7

Future work

Although the principle of using the ExSAM to improve synchronization times has been shown, there are some further actions that could be taken in order to present a more complete demonstration:

- Tuning and testing the lookup table speed control and integrating it into the shifting sequence
- Dynamic changing of the range gear
- Dynamic engaging and disengaging the clutch
- Using the retarder brake to simulate uphill gear changes
- Further investigation of delay in shift sequence
- Reading the encoders on the output shaft and diesel engine drive shaft

Furthermore, in order to bring this technology to a point where it can be used in actual vehicles, appropriate control algorithms must be implemented in the transmission control unit (TECU) and extensive testing is required in a more sophisticated rig and eventually in a real vehicle.

7.1 Lookup tables

At present, the speed control solution involving lookup tables is not implemented in the shifting sequence. Although the code is present in the LabVIEW program the need for thorough testing has not yet been fulfilled. Once tuning and rigorous testing of this speed control has been completed its integration into the shift sequence should be a relatively small step. The benefit of this is the possibility to run above the base speed of the ExSAM, which would enable the simulation of a more realistic drive cycle. Note that the flywheel on the output shaft would need to be balanced in order to not compromise safety when higher output speeds are in question.

7.2 Dynamic changing of range gear

To show a full acceleration and deceleration drive cycle the range gear must also be able to be changed dynamically in order to be able to shift between $5th$ and $7th$ gear. The range is a planetary gear set and is automatically synchronized when changed as long as the main gears are in neutral position. Therefore, adding the function to dynamically change the range gear does not present any major problems. The reason it is not yet implemented is a lack of time and space on the FPGA. Adding a further six gears means that the possible combinations of gear changes requested by the user increases from 36 to 144. Even if the possibilities where the gear change clearance would not be given due to unreasonably large speed steps the code would more than double in size. The matrix in figure 7.1 shows these possible combinations, where the blue field signifies the gears covered by the sequence as it is now; without the dynamic changing of the range gear incorporated.

7.3 Dynamic engaging and disengaging the clutch

Adding the possibility to dynamically engage and disengage the clutch is at this stage only advantageous from a demonstration point of view. The synchronization of the countershaft occurs once the transmission has been disengaged from the diesel engine, which means that proving the principle of improved synchronization times does not involve the diesel engine. Despite this, the overall presentation and authenticity of the

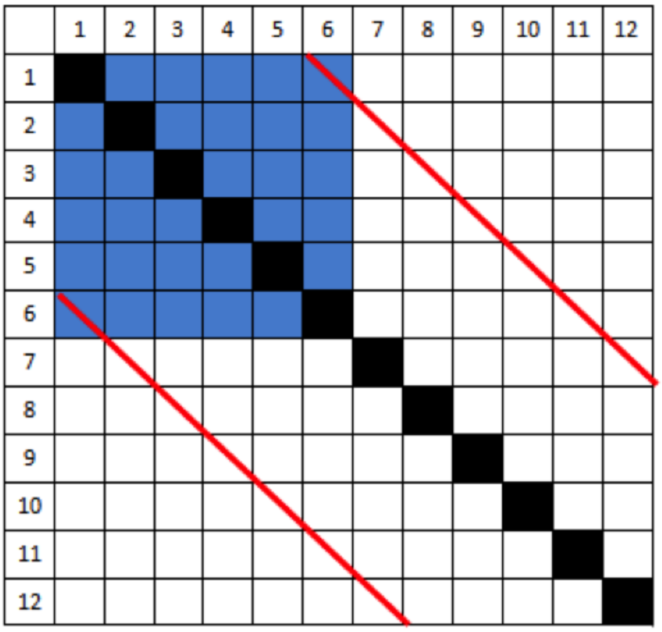

FIGURE 7.1: Shift combinations

simulated drive cycle would be improved if the clutch could be operated dynamically. However, there are some limitations associated with this. The diesel engine, in its present state, can only deliver five discrete throttle levels including no throttle, which to the greater extent can only result in four different speeds. In order for a shifting sequence to include the clutch the ExSAM would need to be used for synchronization a second time in the sequence but this time to synchronize the input shaft of the engaged gearbox to the driveshaft from the diesel engine.

To illustrate this with an example, suppose the diesel engine is operating at throttle level 4, the clutch is engaged and the transmission is in $3rd$ gear. A change to $5th$ gear is requested and the shifting sequence starts; the clutch is disengaged, gears are sent to neutral, the countershaft is synchronized, new gear is engaged and it is at this point that the second synchronization is needed. The speed of the transmission is set to match the speed of the flywheel on the output shaft before the new gear is engaged, this is to avoid changes in speed on the output shaft creating the feeling of a jerky gear change. However this speed might no longer match any of the discrete speed steps of the diesel engine. Most likely it will correspond to a level somewhere between the throttle levels. This means that before the clutch can be engaged the ExSAM must synchronize the entire transmission and flywheel to match one of the throttle levels, creating that jerky speed change on the output shaft that has thus far been avoided.

Another possible course of action for successfully incorporating the diesel engine would be to change the discrete throttle control to a continuous analog control, allowing better compatibility with the rest of the test rig.

7.4 Simulating uphill gear changes with retarder brake

A function that simulates uphill driving could be programmed in the rig control using the retarder brake to increase the load seen from the ExSAM. This would affect the gear shifting sequence, as the decrease in speed on the output shaft during the torque interruption would increase significantly. This could mean that the speed overshoot that is set in state seven of the shifting sequence would need to be increased. It is likely that some sort of simultaneous control is needed that adjusts the overshoot as a function of the load added by the retarder brake.

7.5 Further investigation of delay in shift sequence

The fast tests with low safety show a significant improvement in the shifting times so it is clear that waiting for acknowledgment signals is taking much too long. The CAN-bus is sending a stream of messages that loops every 20 ms, which suggests the delay should not be much longer than this. However our results show that it is. This suggests that it may be caused by the coded loop in LabVIEW used to read the data from the messages or perhaps even the loop that translates the data into gear positions. Either way this could be investigated further in order to improve the shift sequence.

7.6 Output shaft and diesel engine driveshaft encoders

Since Hammad Khan's thesis work there is an encoder situated at the base of the output shaft and another on the driveshaft of the diesel engine [3]. Some difficulty is currently experienced when attempting to read these encoders but due to a lack of time not much effort is given to solving this problem. At present, the output speed is calculated from the shaft speeds read via the CAN-bus. As the diesel engine is not yet involved in the shifting sequence being able to read this encoder is not yet necessary.

Appendix A

LabVIEW code - Gear shifting sequence

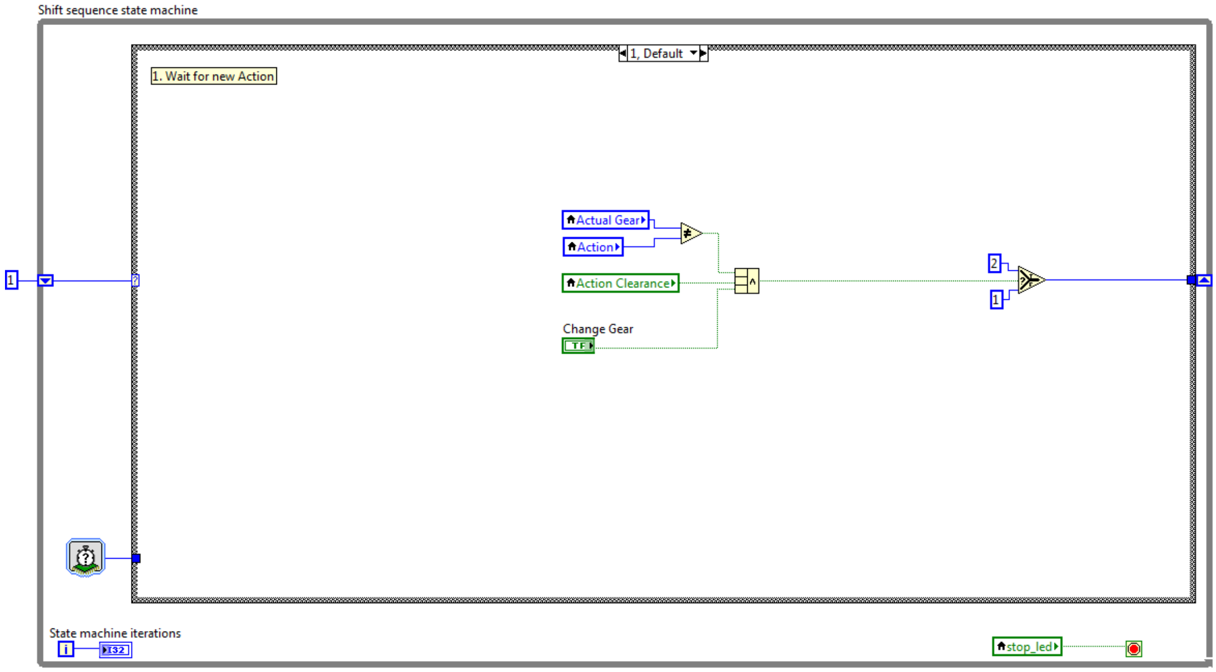

FIGURE A.1: State 1: Wait for gear change command

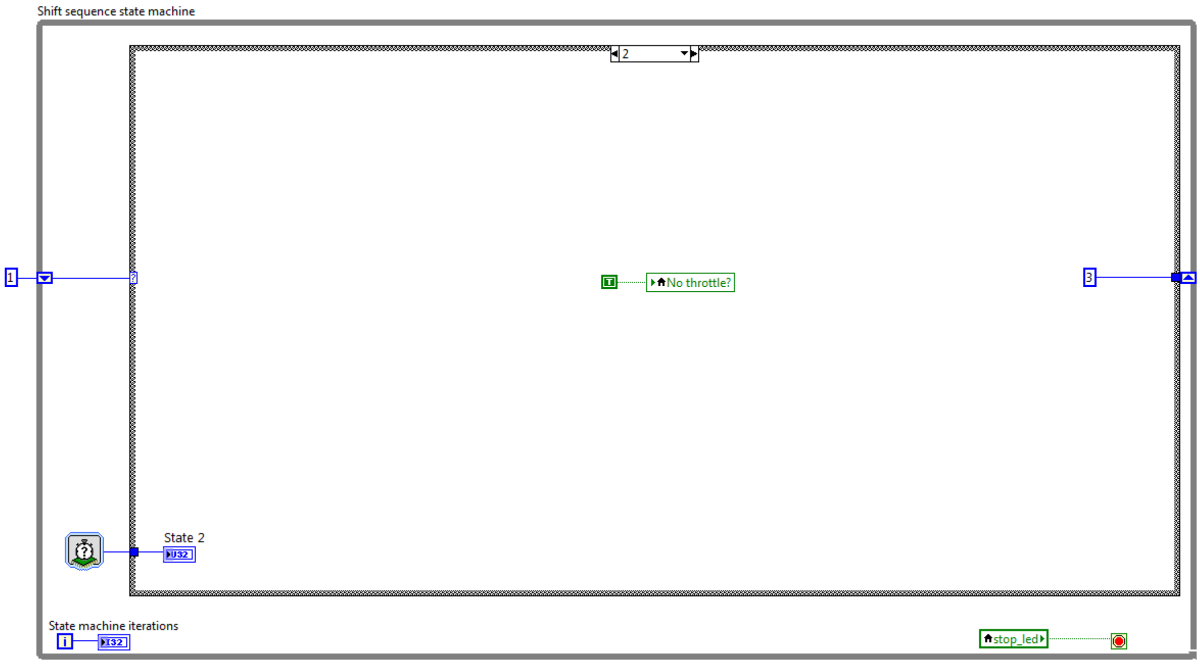

FIGURE A.2: State 2: Turn off diesel throttle

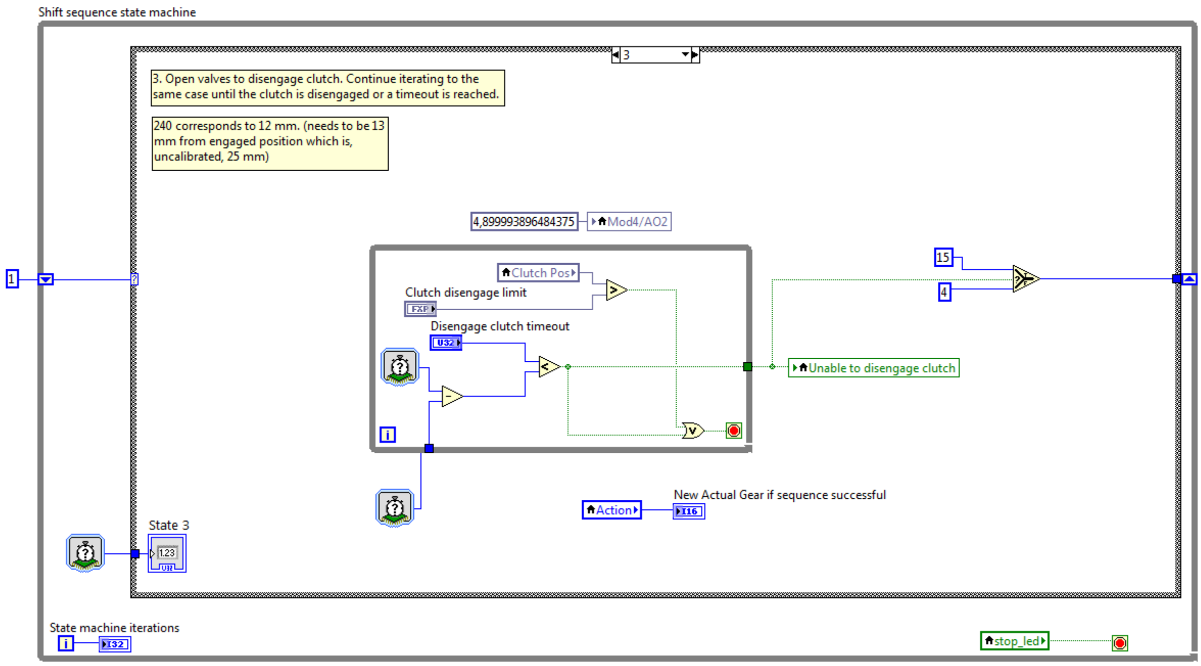

FIGURE A.3: State 3: Disengage clutch

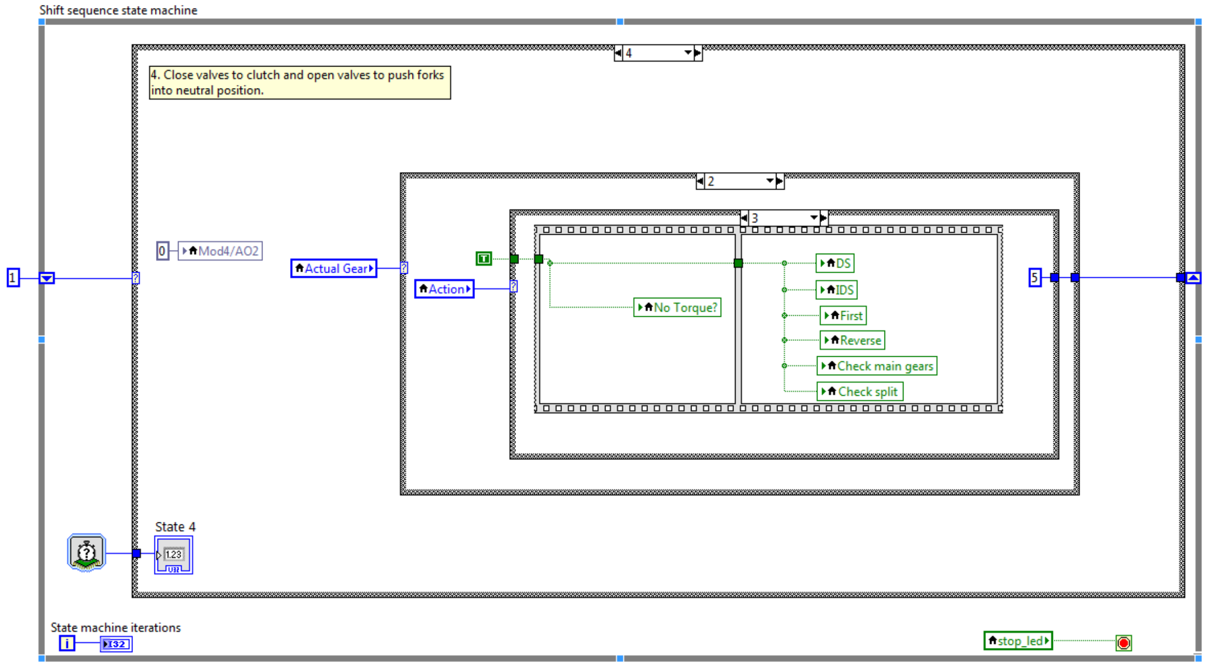

FIGURE A.4: State 4: Remove EM torque and go to neutral

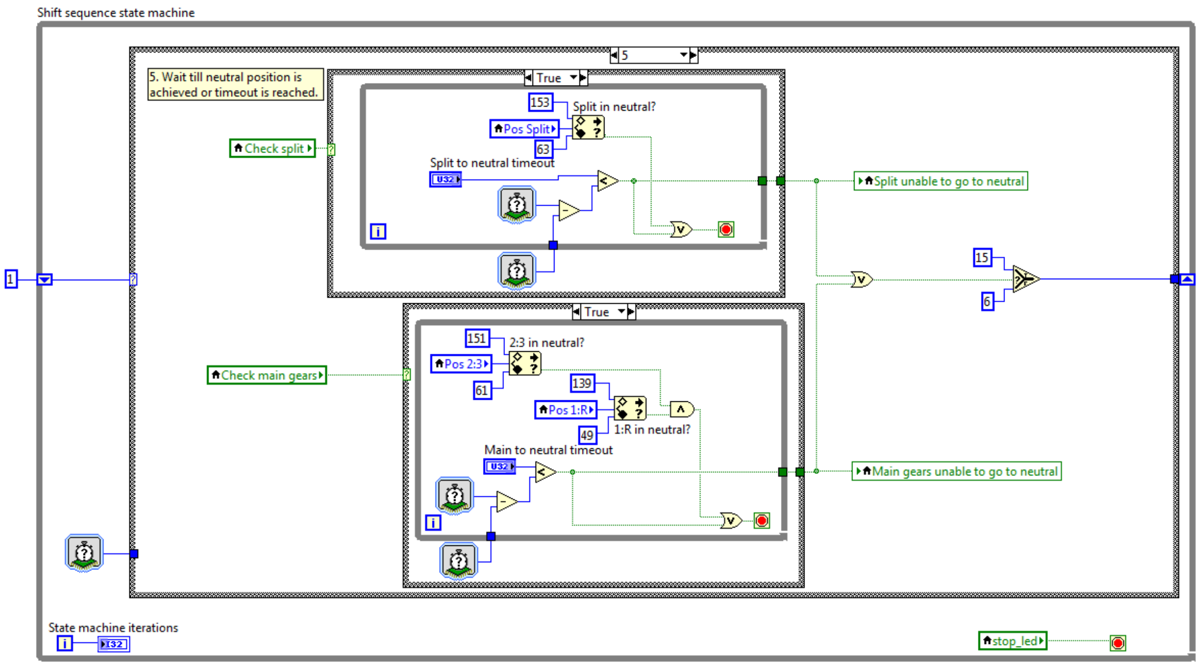

FIGURE A.5: State 5: Confirm neutral

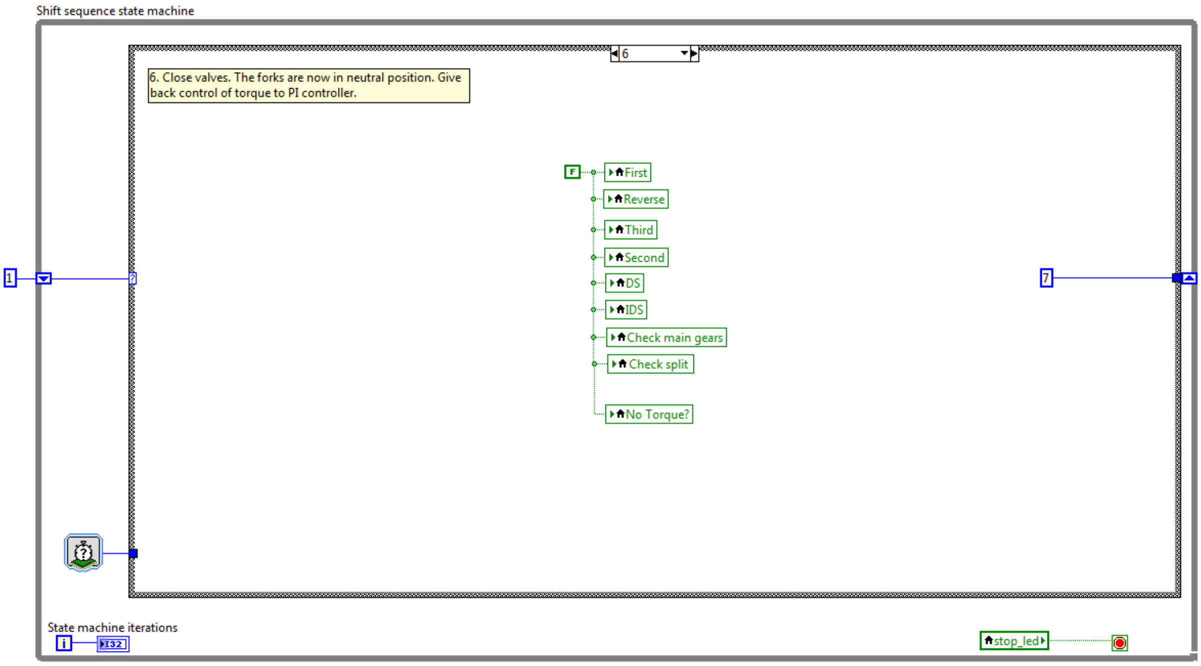

FIGURE A.6: State 6: Close air valves and return EM torque

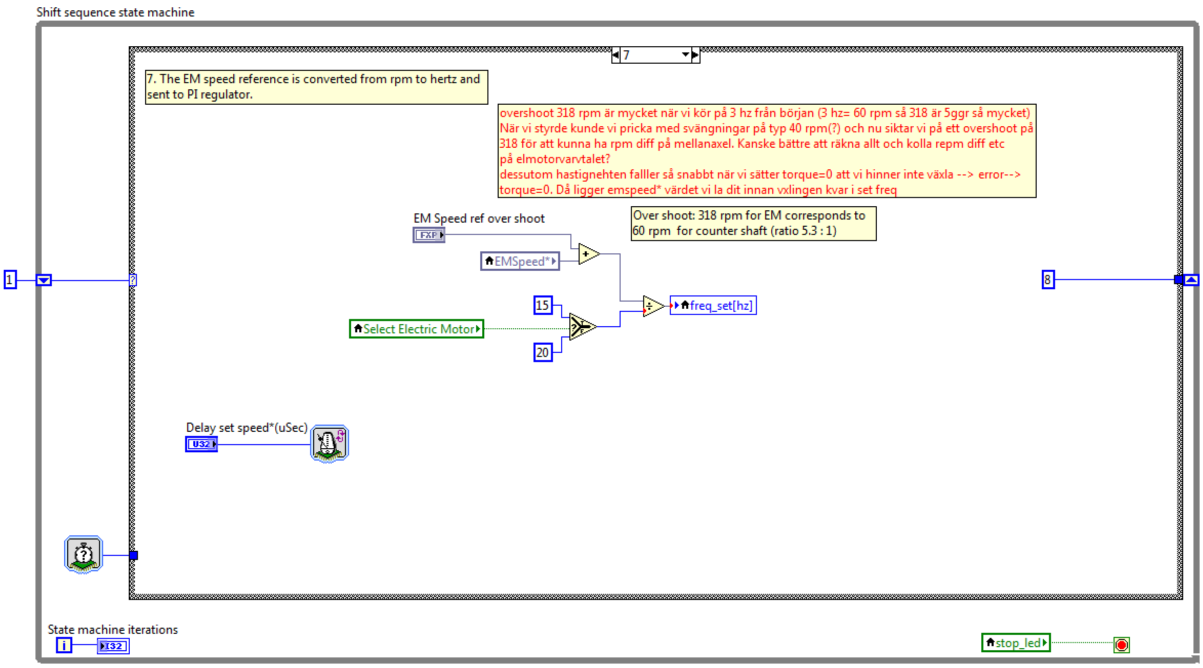

FIGURE A.7: State 7: Synchronize countershaft

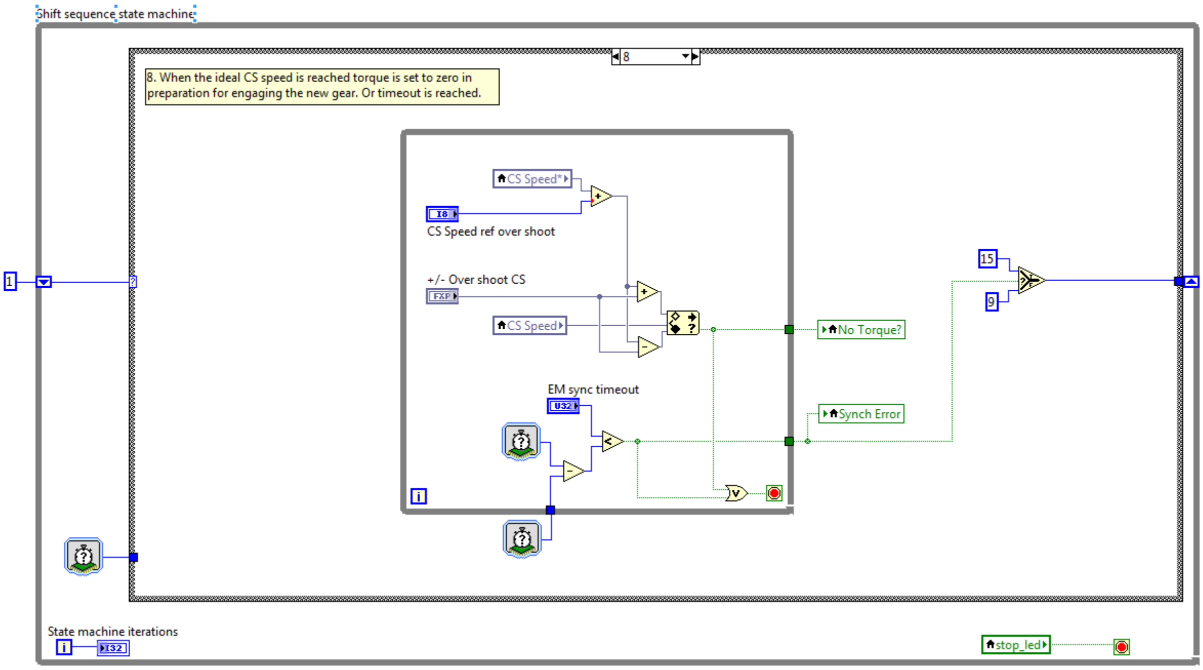

Figure A.8: State 8: Wait until synchronized

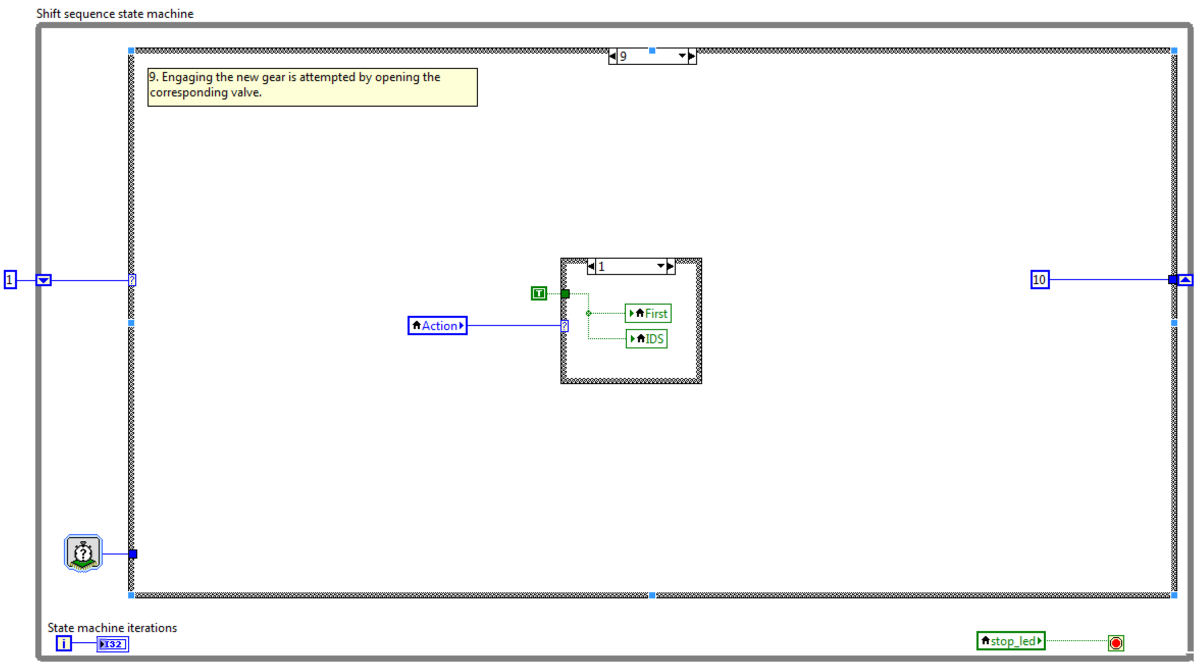

Figure A.9: State 9: Remove EM torque and engage new gear

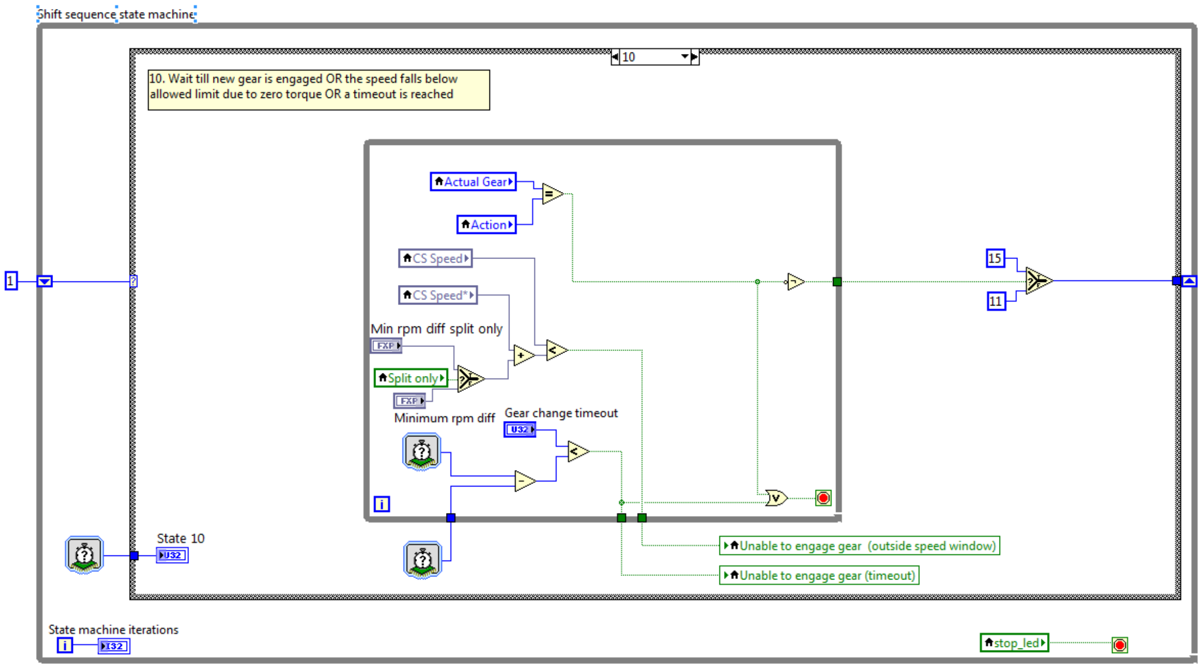

FIGURE A.10: State 10: Confirm new gear

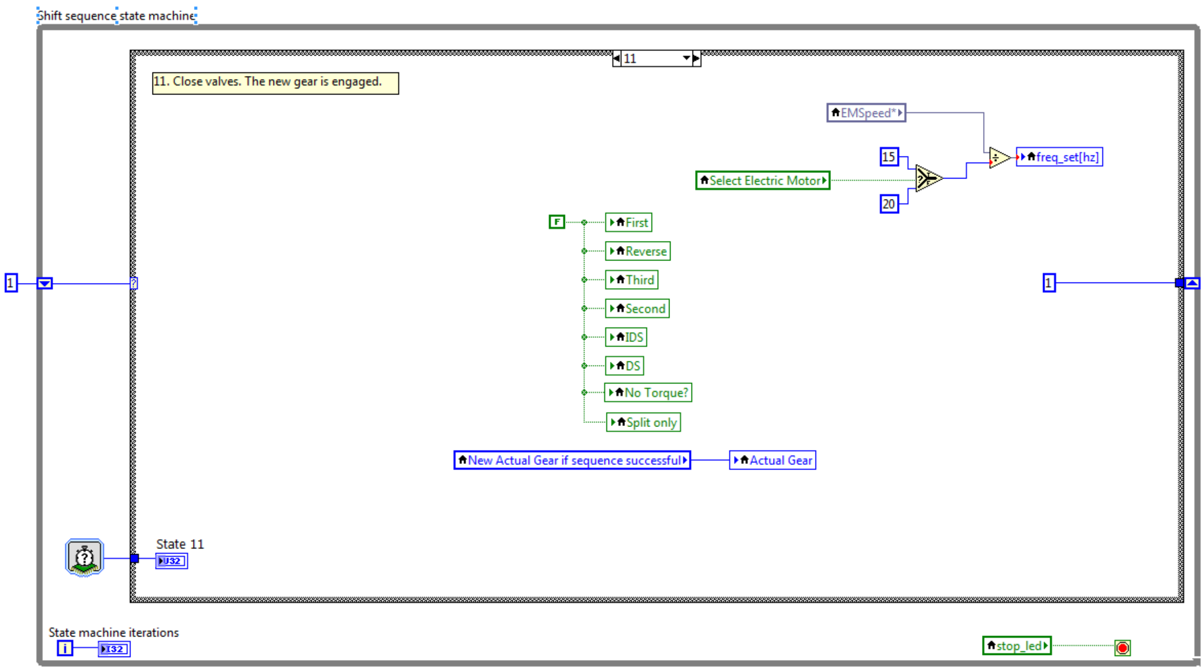

FIGURE A.11: State 11: Close air valves and return EM torque

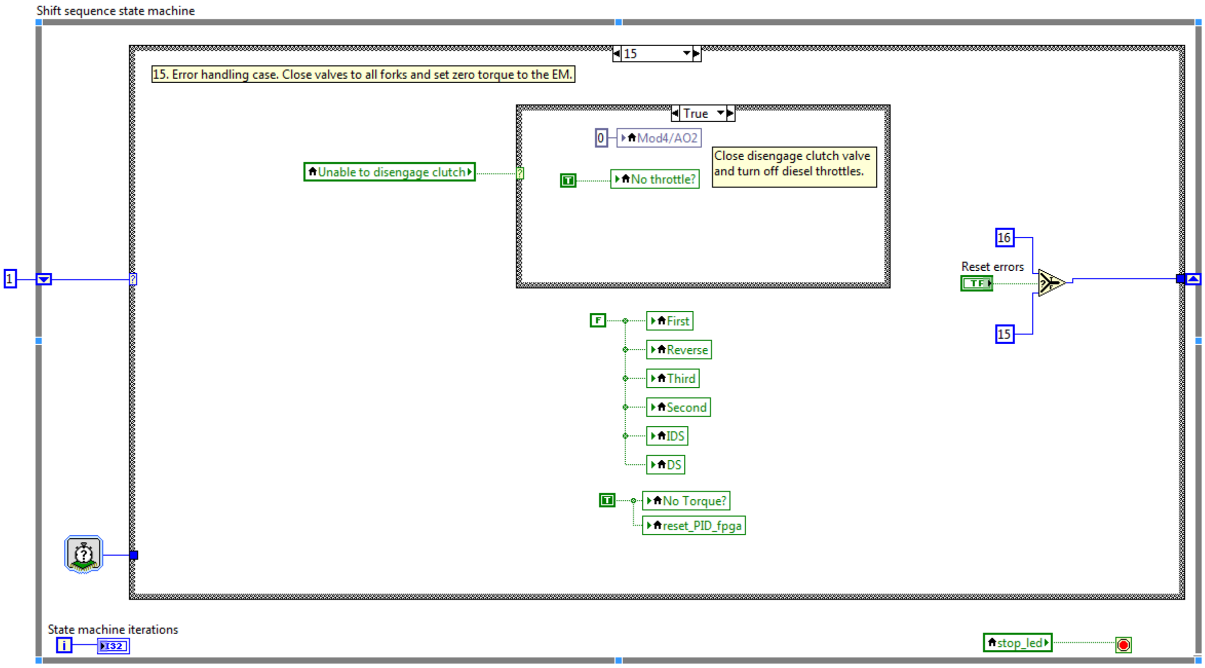

FIGURE A.12: State 15: Error handling

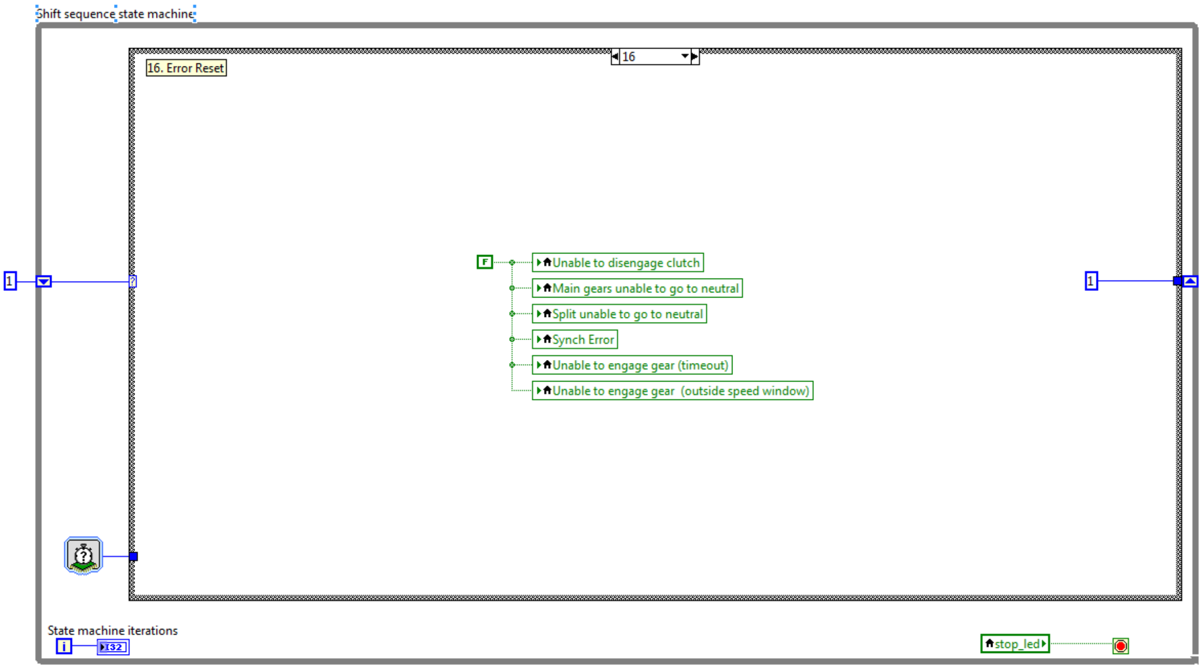

FIGURE A.13: State 16: Error reset

Appendix B

Shift sequence variable list

B.1 *Actual gear:*

An integer indicator that says which gear the the gearbox is in (excluding the range gear). It can assume values 0 through to 6. Variable determined in ForkPos.vi.

TABLE B.1: *Actual gear*

Actual gear	Low Range	High Range
0	Error: at least one of 1:R, 2:3 or Split forks is in neutral position.	Error: at least one of 1:R, 2:3 or Split forks is in neutral position.
1	$1st$ gear	$7th$ gear
2	$2nd$ gear	$8th$ gear
3	$3rd$ gear	$9th$ gear
4	$4th$ gear	$10th$ gear
5	$5th$ gear	$11th$ gear
6	$6th$ gear	$12th$ gear

B.2 *Main gear state:*

An integer indicator that says which gear the gearbox is in with respect to the main gears (this excludes the split and range gears). It can have values 0, 1, 2 or 3 corresponding to neutral, $1st$, $2nd$, and $3rd$ gear. Variable determined in ForkPos.vi.

B.3 *Action:*

An integer control set by the user or drive cycle to specify the next gear. It can be allocated values 1 through to 6. *Action* must be different to *Actual Gear* for the shift sequence to commence.

Table B.2: *Action*

Actual gear	Low Range	High Range
1	$1st$ gear	$7th$ gear
2	$2nd$ gear	$8th$ gear
3	$3rd$ gear	$9th$ gear
4	$4th$ gear	$10th$ gear
5	$5th$ gear	$11th$ gear
6	$6th$ gear	$12th$ gear

B.4 *Action Clearance:*

A boolean indicator that must be TRUE if a shift sequence is to commence. Variable set in Emspeed_ref.vi. It is set to TRUE if the following criteria would be satisfied after changing to the requested new gear:

1. Input shaft speed (diesel engine speed) $<$ 2500 rpm
2. EM speed $<$ 12000 rpm
3. $|$ EM speed step $|$ $<$ 3000 rpm

B.5 *Change Gear:*

A boolean control that must be set to TRUE for the shift sequence to commence.

B.6 *Reverse, First:*

Boolean controls that open the pneumatic valves that move the 1:R fork. By sending TRUE to both of them the fork position can be set to neutral. Set automatically in shift sequence.

B.7 *Second, Third:*

Boolean controls that open the pneumatic valves that move the 2:3 fork. By sending TRUE to both of them the fork position can be set to neutral. Set automatically in shift sequence.

B.8 *DS, IDS:*

Boolean controls that open the pneumatic valves that move the split fork from High Split (DS) to Low split (IDS) respectively. By sending TRUE to both of them the fork position can be set to neutral. Set automatically in shift sequence.

B.9 *LR, HR:*

Boolean controls that open the pneumatic valves that set Low Range (LR) and High Range (HR) respectively. The range gear should not be changed while the shafts are in motion.

B.10 *No Torque?:*

A boolean control used to set the torque in the electrical machine to zero by setting the Is_y reference to zero. TRUE: Is_y reference is zero.

B.11 *Unable to disengage clutch:*

A boolean indicator that shows if an error occurred while attempting to disengage the clutch. It is set to TRUE if a timeout limit is reached before an acknowledgment that the disengage was successful is received.

B.12 *Main gears unable to go to neutral:*

A boolean indicator that shows if an error occurred while attempting to put 1:R and/or 2:3 forks into neutral position. It is set to TRUE if a timeout limit is reached before an acknowledgment that the attempt was successful is received.

B.13 *Split unable to go to neutral:*

A boolean indicator that shows if an error occurred while attempting to put Split fork into neutral position. It is set to TRUE if a timeout limit is reached before an acknowledgment that the attempt was successful is received.

B.14 *Sync Error:*

A boolean indicator that shows if an error occurred while using the ExSAM to synchronize the countershaft. It is set to TRUE if a timeout limit is reached before the countershaft reaches the ideal speed.

B.15 *Unable to engage gear:*

A boolean indicator that shows if an error occurred while attempting to engage the new gear. It is set to TRUE if a timeout limit is reached before an acknowledgment that the attempt was successful is received.

B.16 *RESET ERRORS:*

A boolean control that sets all the error indicators to FALSE. If an error occurred in the previous gear change sequence this must be done before a new gear change is attempted.

B.17 *Pos 1:R:*

An integer indicator that shows the value sent from CAN that represents the position of the 1:R fork. Divide by five to get the position in millimetres.

B.18 *Pos 2:3:*

An integer indicator that shows the value sent from CAN that represents the position of the 2:3 fork. Divide by five to get the position in millimetres.

B.19 *Pos Split:*

An integer indicator that shows the value sent from CAN that represents the position of the split fork. Divide by five to get the position in millimetres.

B.20 *Pos Range:*

An integer indicator that shows the value sent from CAN that represents the position of the range gear. Divide by five to get the position in millimetres.

B.21 *Clutch Pos:*

A fixed point indicator that shows the value sent from CAN that represents the position of the clutch. Divide by 20 to get the position in millimetres.

B.22 *Oil Temp:*

An integer indicator read from CAN that shows the oil temperature in the gearbox. Currently not in use.

B.23 *Select Electric Motor:*

A boolean control that must be set to show which electric motor the program is to control: Haldex with four pole pairs or ExSam with three pole pairs. This variable affects how the rpm speed of the electric motor is calculated and is crucial for the synchronization step in the shift sequence. TRUE: Haldex, FALSE: ExSam.

B.24 *Mod7/CAN0:*

Is a cluster indicator that shows the elements of each received CAN-message. Because the messages are received at a such a fast rate, this indicator is mainly used to confirm that messages are in fact being received.

B.25 *Fork pos:*

An integer array of indicators that shows the data from the CAN-message that describes the position of the forks in the gearbox.

B.26 *Clutch position:*

An integer array of indicators that shows the data from the CAN-message that describes the position of the clutch.

B.27 *Oil temp:*

An integer array of indicators that shows the data from the CAN-message that describes the oil temperature in the gearbox.

B.28 ***Speeds:***

An integer array of indicators that shows the data from the CAN-message that describes shaft speeds in the gearbox.

B.29 ***CANCEL initial gear state reading:***

A boolean control used if for some reason no CAN-messages are being received. This is to avoid getting trapped in an initial while loop that is attempting to determine the gear state of the gearbox using position information sent from CAN.

B.30 ***ERROR HANDLING***

If any of the error variables are set to TRUE the program will automatically close the valves to the forks in the gearbox and set the torque in the electrical machine to zero.

In the case of the Unable to disengage clutch error the program will also automatically close the disengage clutch valve and turn off the diesel throttles.

Appendix C

Matlab code

Inertia

```
%%%%% J = 0.5*pi*rho*l*(R^4-r^4)
rho=7850;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
%%%%% J for each separate part in PTO-transmission

%%% 10298 "ingående axel"
l1=220.5e-3-126e-3;
l2=126e-3-45e-3;
l3=45e-3;
R1=15e-3;
R2=25e-3;
R3=68.75e-3/2;

J1=0.5*pi*rho*l1*R1^4;
J2=0.5*pi*rho*l2*R2^4;
J3=0.5*pi*rho*l3*R3^4;

J10298=J1+J2+J3;

%%% 10299 "mellanaxel"
l1=158.5e-3-108e-3;
l2=108e-3-56e-3;
l3=5e-3;
l4=51e-3;
R1=32.5e-3;
R2=36e-3;
```

```
R3=39e-3;
R4=49.05e-3;
r=4.5e-3;

J1=0.5*pi*rho*l1*(R1^4-r^4);
J2=0.5*pi*rho*l2*(R2^4-r^4);
J3=0.5*pi*rho*l3*(R3^4-r^4);
J4=0.5*pi*rho*l4*(R4^4-r^4);

J10299=J1+J2+J3+J4;

%%% 10300 "mellanaxel 2"
l1=158.5e-3-114e-3;
l2=114e-3-50e-3;
l3=8e-3;
l4=42e-3;

R1=25e-3;
R2=29e-3;
R3=32e-3;
R4=83.51e-3/2;

r=4.5e-3;

J1=0.5*pi*rho*l1*(R1^4-r^4);
J2=0.5*pi*rho*l2*(R2^4-r^4);
J3=0.5*pi*rho*l3*(R3^4-r^4);
J4=0.5*pi*rho*l4*(R4^4-r^4);

J10300=J1+J2+J3+J4;

%%% 10301 "utgående axel"
```

```
l1=196.8e-3-126e-3;
l2=33.5e-3;
l3=(126-50-16+17.5)*1e-3;
l4=15e-3;
l5=35e-3;

R1=17.5e-3;
R2=21e-3;
R3=27.5e-3;
R4=135.3e-3/2;
R5=135.3e-3/2;

r=4.5e-3;
r5=44e-3;

J1=0.5*pi*rho*l1*(R1^4-r^4);
J2=0.5*pi*rho*l2*(R2^4-r^4);
J3=0.5*pi*rho*l3*(R3^4-r^4);
J4=0.5*pi*rho*l4*(R4^4-r^4);
J5=0.5*pi*rho*l5*(R5^4-r5^4);

J10301=J1+J2+J3+J4+J5;

%%% 10302 "kugghjul mellan"
l1=40e-3;
R1=158e-3/2;
r=36e-3;

J10302=0.5*pi*rho*l1*(R1^4-r^4);

%%% 10303 "kugghjul mellan 2"
l1=43e-3;
```

```
R1=49e-3;
r=29e-3;

J10303=0.5*pi*rho*l1*(R1^4-r^4);

%%% 10297 "hylsa kilförband"
l1=64e-3;
R1=25e-3;
r=15e-3;

J10297=0.5*pi*rho*l1*(R1^4-r^4);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
Jem=0.032;
Jload=0;

J1=J10298;
J2=J10303;
J3=J10300;
J4=J10302;
J5=J10299;
J6=J10301;

%%% Number of teeth on gears
N1=21;
N2=38;
N3=20;
N4=41;
N5=21;
N6=30;
```

```
%%% J total PTO disengaged from counter shaft
Jtot=Jem+J1+(N1/N2)^2*(J2+J3+(N3/N4)^2*(J4+J5+(N5/N6)^
2*(J6+Jload)))

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%
%%%% J for total power transmission

%%% J input shaft
JclutchDisc=0.15;
JinputShaft=0.0218;
J_LP=0.00968;
J_HP=0.017;
JinputShaftTot=JclutchDisc+JinputShaft+J_LP+J_HP;

%%% J counter shaft
J_cs=0.1818;

%%% J load with LP
N_LP=31;
N_cs=44;
Jload_LP=J_cs+(N_cs/N_LP)^2*JinputShaftTot;

%%% J load with HP
N_HP=35;
N_cs=39;
Jload_HP=J_cs+(N_cs/N_HP)^2*JinputShaftTot;

%%% J load with split disengaged
Jload_cs=J_cs;

%%% Jtot with split in LP,HP and neutral
```

```
Jtot_LP=Jem+J1+(N1/N2)^2*(J2+J3+(N3/N4)^2*(J4+J5+(N5/N
6)^2*(J6+Jload_LP)))
Jtot_HP=Jem+J1+(N1/N2)^2*(J2+J3+(N3/N4)^2*(J4+J5+(N5/N
6)^2*(J6+Jload_HP)))
Jtot_neutral=Jem+J1+(N1/N2)^2*(J2+J3+(N3/N4)^2*(J4+J5+
(N5/N6)^2*(J6+Jload_cs)))

>> Inertia
Jtot =
    0.0370
Jtot_LP =
    0.0577
Jtot_HP =
    0.0522
Jtot_neutral =
    0.0434
```

Time Ref

```
J=[0.0577 0.0522 0.0434];% [LP,HP,neutral]

T=255; % Torque is constant from 0 to 5500 rpm
t_1000rpm=(J/T)*2*pi*(1000/60)
t_2000rpm=(J/T)*2*pi*(2000/60)
t_3000rpm=(J/T)*2*pi*(3000/60)
t_4000rpm=(J/T)*2*pi*(4000/60)

T2=104; % Estimation from 12000 rpm (max).
        % Does not take into consideration that T
decreases in this rpm section.
t2_1000rpm=(J/T2)*2*pi*(1000/60)
t2_2000rpm=(J/T2)*2*pi*(2000/60)
t2_3000rpm=(J/T2)*2*pi*(3000/60)
t2_4000rpm=(J/T2)*2*pi*(4000/60)

>> TimeRef
t_1000rpm =
    0.0237    0.0214    0.0178
t_2000rpm =
    0.0474    0.0429    0.0356
t_3000rpm =
    0.0711    0.0643    0.0535
t_4000rpm =
    0.0948    0.0857    0.0713
t2_1000rpm =
    0.0581    0.0526    0.0437
t2_2000rpm =
    0.1162    0.1051    0.0874
```

```
t2_3000rpm =
    0.1743    0.1577    0.1311
t2_4000rpm =
    0.2324    0.2102    0.1748
```

Bibliography

[1] AB Volvo (publ), "Facts about hybrids and Volvo Group I-SAM parallel hybrid system", http://www.volvobuses.com/bus/uk/en-gb/newsmedia/pressreleases/cwp2moss.news.documents/1385 (accessed: 07/05/2013).

[2] Conversation with Anders Göransson, Tech Center at Sibbhultsverken, 27/03/2013.

[3] Hammad Khan, "Dynamic Control of PMSM via Profibus Communication Between NI-CRio and Vacon Frequency Converter", *Master Thesis, Department of Industrial Electrical Engineering and Automation, Lund Institute of Technology*, Lund, Sweden, 2012.

[4] http://images.volvotrucks.com (accessed: 02/07/2013).

[5] http://www.beam-wiki.org/wiki/Image:Epicyclical_%28Planetary%29_Gear_Train.GIF (accessed: 07/05/2013)

[6] E-mail conversation with PhD student Francisco Marquez, 16/04/2013.

[7] Mats Alaküla and Per Karlsson, "Power Electronics - Devices, Converters, Controls and Applications", *Department of Industrial Electrical Engineering and Automation, Lund Institute of Technology*, Lund, Sweden, pages: 240–243.

[8] Dan Strassberg, "Choosing a Real-Time Platform: FPGAs vs. Real-Time Operating Systems", April 2004, http://www.edn.com/contents/images/edn0404sNIFIN.pdf (accessed: 10/04/2013).

[9] Volvo, "Volvo är redo för Euro 6. Här är D13K460", 2012, http://www.volvotrucks.com/trucks/sweden-market/sv-se/trucks/environment/Pages/Euro6.aspx (accessed: 07/05/2013).