Position Acquisition and Control for Linear Direct Drives with Passive Vehicles

Vom Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

Dipl.-Ing. Marius Alexandru Mihalachi

Geboren am 29.08.1981 in Câmpina, Rumänien

Referent:	Prof. Dr.-Ing. Peter Mutschler
Korreferent:	Prof. Dr.-Ing. Wolf-Rüdiger Canders

Tag der Einreichung:	1.7.2010
Tag der mündlichen Prüfung:	23.11.2010

D17
Darmstadt 2011

Abstract

For combined processing and transportation of materials in industrial production lines, a long primary, linear synchronous drive with passive, lightweight vehicles, is being designed and experimentally tested. This thesis concentrates on position acquisition and motion control of the proposed system.

In order to allow a high degree of independency in the movement of the vehicles, the stator (primary) of the linear machine is divided into many segments. Each segment of the track is fed by a dedicated power stack, and control information is exchanged between all power stacks and all vehicle controllers via an Inverter Bus.

A number of processing stations are spread along the track of the linear drive, being connected by transport sections.

Inside the processing stations, high quality speed and position control of the vehicles is required. For this, precise and fast position measurement is necessary, so position sensors must be used. The passive vehicles impose additional challenges for the position acquisition system, as neither energy nor information must be transmitted to the moving parts.

The evaluation of two position acquisition systems, which comply with this requirement, is presented in this thesis.

The first system is based on a high-resolution optical encoder. For this application, the scale of the optical sensor is mounted at the vehicle and several active read-heads are installed along the track, such that at each position the scale covers at least one read-head. When the scale is passing from one read-head to the next one, the position information from both read-heads must be evaluated simultaneously and synchronised, so that a continuous position signal will result for the entire measuring length.

The second position acquisition system uses a comparatively lower resolution capacitive sensor, and is intended as a simpler and cost effective alternative to the optical system. The principle of operation of a capacitive sensor is first analysed, and a model is determined. Then, based on this model, two methods of extracting the position information are presented: one uses instantaneous (sampling-based) demodulation, while the other is based on phase measurement.

In the transport sections of the linear drive the requirements concerning the accuracy and dynamic of the position measurement are less demanding than in the processing stations. In this sections sensorless control, based on the evaluation of the electromotive force (EMF) is implemented. The distinctive parameters of the different stator segments are taken into consideration.

Due to mechanical constraints, there are gaps in the winding arrangement between consecutive segments of the machine, which means that the EMF vectors of two consecutive segments can have an arbitrary phase difference, providing additional challenges, especially for the sensorless control.

At the transition between processing stations and transport sections, a synchronisation procedure between the measured position and the estimated one is described and experimentally evaluated.

Kurzfassung

Ein Langstator- Linearsynchronantrieb mit leichten, passiven Fahrzeugen, welcher sowohl für die Bearbeitung als auch für den Transport von Materialien in industriellen Produktionsanlagen dienen kann, wird entworfen und experimentell geprüft. Diese Arbeit konzentriert sich auf die Positionserfassung und Bewegungssteuerung des vorgeschlagenen Systems.

Um eine hohe Unabhängigkeit der Bewegung mehrerer Fahrzeuge zu ermöglichen, wird der Ständer (Primärteil) der Linearmaschine in zahlreiche Speiseabschnitte (Segmente) unterteilt. Jedes Segment wird von einem zugeordneten Wechselrichter gespeist. Die Kontrollinformation wird zwischen allen Wechselrichter und allen Fahrzeugkontroller durch einen Wechselrichterbus ausgetauscht.

Mehrere Bearbeitungsstationen werden entlang des Fahrwegs verteilt und durch Transportabschnitte verbunden.

Innerhalb der Bearbeitungsstationen ist eine hochgenaue Geschwindigkeits- und Positionsregelung der Fahrzeuge erforderlich. Eine präzise und schnelle Positionsmessung ist dazu unerlässlich, weshalb Positionssensoren eingesetzt werden müssen. Passive Fahrzeuge stellen zusätzliche Herausforderungen für das Positionserfassungssystem dar, da weder Energie noch Informationen an die beweglichen Teile übertragen werden können.

Die Auswertung zweier Positionserfassungssysteme, die die o.g. Anforderungen erfüllen, wird in dieser Arbeit dargestellt.

Das erste System basiert auf einem optischen Encoder mit hoher Auflösung. Für die gegebene Anwendung wird die Maßverkörperung des Sensors am Fahrzeug angebracht und mehrere aktive Abtastköpfe entlang der Fahrbahn montiert, so dass in allen Positionen die Maßverkörperung mindestens einen Abtastkopf überdeckt. Beim Übergang der Maßverkörperung von einem Abtastkopf auf den nächsten müssen die Signale von beiden Abtastköpfe gleichzeitig abgetastet und synchronisiert werden, damit ein kontinuierliches Positionssignal über die gesamte Messlänge erzeugt wird.

Das zweite Positionserfassungssystem verwendet einen kapazitiven Sensor mit einer relativ niedrigeren Auflösung, welches eine einfachere und kostengünstigere Alternative zu dem optischen System darstellt. Die Arbeitsweise des kapazitiven Sensors wurde zunächst analysiert und ein Model ermittelt. Basierend auf diesem Model werden zwei Auswertungsmethoden dargestellt: eine verwendet unverzügliche (abtastungsbasierte) Demodulation und die andere basiert auf Phasenmessung.

Innerhalb der Transportabschnitte des Linearantriebs sind die Anforderungen an Genauigkeit und Dynamik der Positionserfassung weniger anspruchsvoll als innerhalb der Bearbeitungsstationen. In diesen Abschnitten wird sensorlose Regelung implementiert, basierend auf der Auswertung der Elektromotorischen Kraft (EMK). Die unterschiedlichen Parameter aller Statorsegmente werden dabei berücksichtigt.

Aufgrund mechanischer Beschränkungen sind Lücken in den Ständerwindungen zwischen aufeinanderfolgenden Speiseabschnitten vorhanden. Dies bedeutet, dass die EMK-Vektoren zweier aufeinanderfolgender Statorsegmente einen beliebigen Phasenunterschied haben können, wodurch sich insbesondere für die sensorlose Regelung zusätzliche Herausforderungen ergeben.

Beim Übergang zwischen Bearbeitungsstationen und Transportabschnitten wird ein Synchronisierungsverfahren, zwischen der gemessenen und beobachteten Position, dargestellt und experimentell ausgewertet.

Contents

Abbreviations

3D	Three-Dimensional
A/D	Analog to Digital
ADC	Analog to Digital Converter
ADDR	Address
APM	Advanced Power Management
ASIC	Application-Specific Integrated Circuit
atan2	Four-quadrant Arctangent Function
BIOS	Basic Input / Output System
CHSM	Current Head State Machine
CNTR	Period Counter
COMP	Comparator
CPLD	Complex Programmable Logic Device
DC	Direct Current
DMA	Direct Memory Access
DRV	Driver
DSP	Digital Signal Processor
EMF	Electromotive Force
EMI	Electromagnetic Interference
EMK	Elektromotorische Kraft
ENA	Enable
EPROM	Electrically Programmable, Read-Only Memory
FEM	Finite Elements Method
FIFO	First In First Out
FPGA	Field Programmable Gate Array
FPU	Floating Point Unit
FR4	Flame Retardant, woven glass reinforced epoxy resin
HAL	Hardware Abstraction Layer
IGBT	Insulated Gate Bipolar Transistor
IRQ	Interrupt Request
ISA	Industry Standard Architecture
JTAG	Joint Test Action Group
MOSFET	Metal-Oxide-Semiconductor Field Effect Transistor
MSB	Most Significant Bit

MUX	Multiplexer
PC	Personal Computer
PCB	Printed Circuit Board
PCI	Peripheral Component Interconnect
PI	Proportional-Integrative (Controller)
PLL	Phase Locked Loop
PMS(L)M	Permanent Magnet Synchronous (Linear) Machine
PS	Power Stack
PSI	Power Stack Interface
PWM	Pulse Width Modulation
RAM	Random Access Memory
RCV	Receiver
r.m.s.	Root-Mean-Square
RTAI	Real-Time Application Interface
RTM	Real-Time Module
RTW	Real-Time Workshop
S2D	Sensor to Digital (Board)
SB	Sensor Bus
SMC	Soft Magnetic Compound
SS	Stator Segment
TTL	Transistor-Transistor Logic
UI	User Interface
USB	Universal Serial Bus
VC	Vehicle Controller
VCI	Vehicle Controller Interface

Symbols

Optical Sensors System (Section 2.2)

/BS	Analog to Digital Converter Busy Signal (active low)
/CS	Analog to Digital Converter Chip Select Signal (active low)
/RD	Analog to Digital Converter Read Signal (active low)
ΔN	Number of periods covered by an optical read-head
$C_1, ... C_5$	Cosine signals from optical read-heads 1, ... 5
C_A, C_B	Cosine signals routed through Multiplexers A, B
$D_0 ... D_7$	Data Frames 0, ... 7 (Sensor Bus Communication)
$H_1, ... H_5$	Optical read-heads 1, ... 5
H_A, H_B	Optical read-heads whose signals are routed through Multiplexers A, B
LD_A, LD_B	Load signals of Period Counters A, B
LDI	Local Data Input (PCI Interface Firmware)
LDO	Local Data Output (PCI Interface Firmware)
MC_A, MC_B	Control Codes of Multiplexers A, B
N_A, N_B	Values of the period counters A, B
P	Period (pitch) of the optical sensors
R-....	Register ... (Sensor to Digital Firmware)
R_1, R_5	Reference signals from read-heads 1 (leftmost) and 5 (rightmost)
RB-...	Sensor Bus Register ... (Sensor to Digital Firmware)
R-TMP-...	Temporary Register ... (Sensor to Digital Firmware)
$S_1, ... S_5$	Sine signals from optical read-heads 1, ... 5
S_A, S_B	Sine signals routed through Multiplexers A, B
SB-ACK	Sensor Bus Request Acknowledged
SB-IN	Sensor Bus Data Input
SB-OUT	Sensor Bus Data Output
SB-RD	Sensor Bus Read Enable
SB-REQ	Sensor Bus Data Request
SB-WR	Sensor Bus Write Enable
T_D	Delay Period (Sensor Bus Communication)
T_P	Pulse Period (Sensor Bus Communication)
T_T	Total Transmission Time (Sensor Bus Communication)
T_Z	High Impedance State Duration (Sensor Bus Communication)
USE_A, USE_B	Specify if the information from read-heads A, B is to be used in the calculation of the position

x	Position given by the optical sensors system
x_0	Position offset of the current read-head
x_A, x_B	Positions calculated based on the information provided by optical read-heads H_A, H_B
$x_{inc,A}$ $x_{inc,B}$	Incremental positions calculated based on the information provided by optical read-heads H_A, H_B

Capacitive Sensor (Section 2.3)

A	Height of the upper and lower sinusoidal copper patterns on the slider
A_h	Amplitude of the sinusoidal excitation voltages
C_a, ... C_d	Position dependent capacitances
$C_{a,0}$, ... $C_{d,0}$	Position independent capacitances between transmitting and receiving electrodes
$C_{a,in}$, ... $C_{d,in}$	Input capacitances
C_{avg}	Average value of the position dependent capacitances
C_k	Coupling capacitance (between modulating and receiving electrodes)
C_{out}	Output capacitance
C_R	Charge amplifier feedback capacitance
C_{var}	Amplitude of the variation of the position dependent capacitances
d_1	Spacing between transmitting and modulating plates
d_2	Spacing between modulating and receiving plates
d_3	Thickness of each of the three plates of the capacitive sensor
F_a	Position dependent overlapping area between the transmitting electrodes "a" and the modulating electrode
f_h	Frequency of the sinusoidal excitation voltages
G_a	Position dependent overlapping area between the transmitting electrodes "a" and the coupling electrode
h	Height of middle (rectangular) copper pattern on the slider
N_{mp}	Number of sensors' periods covered by the slider (modulating plate)
N_p	Total number of periods of the capacitive sensor
P	Period (pitch) of the capacitive sensor
Q	Quality factor
R_R	Charge amplifier feedback resistance
s	Laplace operator
U_a, ... U_d	Excitation voltages of the capacitive sensor
U_{ac}, U_{db}	Line-to-line excitation voltages
U_{DC}	DC-Link Voltage
U_{out}	Output of the charge amplifier

$U_{\Delta C}$, $U_{\Delta S}$	Outputs of the two MOSFET H-Bridges used to generate the sinusoidal excitation voltages
$U_{\Delta C,F}$, $U_{\Delta S,F}$	Filtered outputs of the two MOSFET H-Bridges
U_0	Sum of the four excitation voltages
w	Width of the transmitting electrodes
x	Position given by the capacitive sensor
x_P	Normalised position of the capacitive sensor
ε_0	Absolute permittivity of vacuum
ε_3	Relative permittivity of epoxy (FR4)
φ_C	Phase shift of the normalised position
ω_h	Pulsation of the sinusoidal excitation voltages

Linear Drive for Material Handling (Section 3)

A	State Matrix (State-Space Representation)
B	Input Matrix (State-Space Representation)
B_v	Friction coefficient of the vehicle
C	Output Matrix (State-Space Representation)
e_α, e_β	Components of the EMF vector of a stator segment in the stationary (α–β) reference frame
$\hat{e}_\alpha$, $\hat{e}_\beta$	Components of the estimated EMF vector in the stationary (α–β) reference frame
$\tilde{e}_\alpha$, $\tilde{e}_\beta$	Components of the EMF estimation error vector in the stationary (α–β) reference frame
F(s)	Transfer function associated to the the estimation error dynamics
F_E	Electrical force
F_E^*	Reference value of the electrical force
F_L	Load force
$\hat{F}_L$	Estimated load force
$\tilde{F}_L$	Load force estimation error
G_F, G_v, G_x	Gains of the mechanical observer
G_Ψ, G_e	Gains of the EMF observer
$\mathbf{I}_3$, $\mathbf{I}_4$	Identity matrices of 3rd, respectively 4th order
i_α, i_β	Components of the current vector of a stator segment in the stationary (α–β) reference frame

i_d, i_q	Components of the current vector of a stator segment in the rotor oriented (d–q) reference frame
i_q^*	Reference value of the quadrature (q) current
K_E	EMF constant
K_F	Current-Force coefficient
K_P	Proportional gain (PI Controller)
L	Inductance matrix
L_0	Mean value of the self-inductance of a stator segment
L_1	Amplitude of the inductance variation of a stator segment
L_2	Mean value of the mutual inductance of a stator segment
M_0,... M_7	States of the Monitoring State Machine
M_v	Mass of the vehicle
p_1, p_2, p_3	Poles of a characteristic polynomial
R	Ramp function
s	Laplace Operator
T_i	Integral time constant (PI Controller)
T_S	Sampling time
u_α, u_β	Components of the voltage vector of a stator segment in the stationary (α–β) reference frame
u_α^*, u_β^*	Components of the reference voltage vector of a stator segment in the stationary (α–β) reference frame
v	Speed of the vehicle
v*	Reference value of the vehicle's speed
$\hat{v}$	Estimated speed
$\tilde{v}$	Speed estimation error
v_C	Speed value used as feedback in the control algorithm
v_S	Speed provided by the optical sensors system
x	Position of the vehicle
x*	Reference value of the vehicle's position
$\hat{x}$	Estimated position
$\tilde{x}$	Position estimation error
x_B	Begin position of a stator segment
x_C	Position value used as feedback in the control algorithm
x_E	End position of a stator segment
x_S	Position provided by the optical sensors system
Γ	EMF observer gains ratio

Δx_C	Difference between the position given by the optical sensors and the estimated position
Δx_V	Length of the vehicle
ε_x	Correction term of the mechanical observer
$\varepsilon_{\Psi,\alpha}$, $\varepsilon_{\Psi,\beta}$	Difference between the measured and the estimated current dependent flux vectors components in the stationary (α–β) reference frame (correction terms of the EMF observer)
θ	Electrical angle of the vehicle
$\hat{\theta}$	Estimated electrical angle of the vehicle
τ_P	Pole pitch of the linear machine
Ψ_α, Ψ_β	Components of the total flux linkage vector of a stator segment in the stationary (α–β) reference frame
$\Psi_{L,\alpha}$, $\Psi_{L,\beta}$	Components of the current dependent flux vector of a stator segment in the stationary (α–β) reference frame
$\hat{\Psi}_{L,\alpha}$, $\hat{\Psi}_{L,\beta}$	Components of the estimated, current dependent flux vector of a stator segment in the stationary (α–β) reference frame
$\tilde{\Psi}_{L,\alpha}$, $\tilde{\Psi}_{L,\beta}$	Components of the estimation error of the current dependent flux vector of a stator segment in the stationary (α–β) reference frame
$\Psi_{PM,\alpha}$, $\Psi_{PM,\beta}$	Components of the flux linkage generated by the permanent magnets in the stationary (α–β) reference frame
ω	Electrical angular speed

1 Introduction

Linear electrical motors are presently gaining an increasingly widespread use in application fields like transportation [1], [2], and industry (machining, actuators, etc.). Linear motors have the capability to produce thrust directly, thus avoiding the use of rotative-to-linear transmission elements like belts and pulleys, racks and pinions or screw systems, which are associated with the rotative motors whenever they must achieve linear motion.

At present, linear drives are used in industry preferably for precise motion with no backlash, high accuracy of positioning and high dynamics (acceleration), on a straight track, e.g. in advanced machine tools.

In industrial production plants, materials must be worked upon in processing stations (e.g. for milling, drilling, grinding etc.) and transported between these stations. Using just one system of linear drives for processing, as well as for transportation, will result in benefits like high precision, high dynamics, high productivity, and reduced wear and maintenance (due to the lack of mechanical transmission elements) [3], [4], [5].

Figure 1.1 shows a simple example of combined transportation and processing of materials with linear drives. In such applications, the following properties are necessary for the linear drive system:

- The carriageway (track) must allow for horizontal and vertical curves, as well as for closed paths and, possibly, switches (S).
- A number of processing stations (P1 … P3) are spread along the track.
- On the carriageway several vehicles (work-piece carriers, V1 … V4) must be able to travel simultaneously, with a high degree of independency.
- Each vehicle has to be controlled very precisely (with an accuracy of a few micrometers) when it operates within a processing station. This requires the use of position sensors.
- While the vehicles are moving outside of the processing stations (i.e. in the transport sections of the track), usually a lower precision in positioning is sufficient,
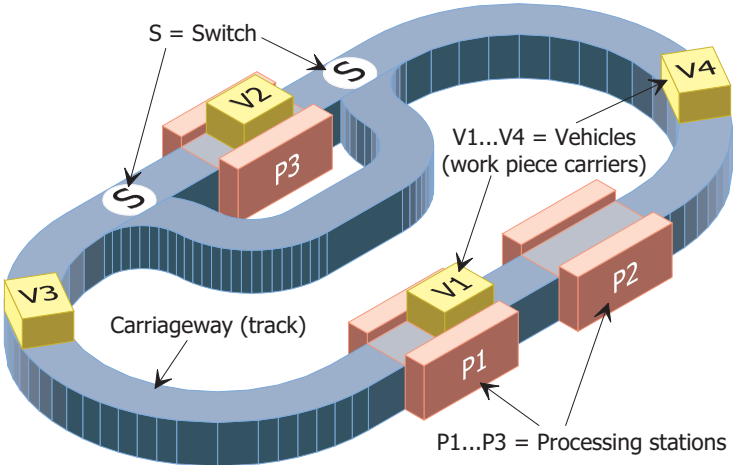

Figure 1.1: Simple example of combined transportation and processing of materials, using a linear drives system

so that a sensorless motion control should be used, avoiding the complexity and costs introduced by position sensors.

The aim is to create a modular system, which allows for different properties at different sections of the track:

- Straight sections
- Curved sections (left/right curves or up/downhill)
- Sections with high force for acceleration
- Sections for high precision positioning (equipped with position sensors), etc.

Two university departments cooperate in this project. The *Institute for Electrical Machines, Traction and Drives* from *Technische Universität Braunschweig* is responsible for the design of a new type of electrical machine (briefly presented in Section 3.1.1) and all the mechanical constructions (track, vehicles). Our department is responsible for hard- and software of power electronics and control of the proposed system.

Due to its high efficiency, high power density and because it allows a higher air gap, the Permanent Magnet Synchronous Machine (PMSM) is a good choice for this application. As high acceleration is mandatory, lightweight, passive vehicles, using an active track, i.e. the long primary configuration of the PMSM (moving magnets), is the best suited. With passive vehicles, there is no energy or information transfer from the stationary side to the vehicle.

In order to allow for individual motion control of several vehicles, the active track is separated into many segments [5], each segment being fed by the power stack of a dedicated inverter. This approach has also the benefit of reduced losses, by turning off the supply of the stator segments where there is momentarily no vehicle. Due to the modular construction of the machine, there are gaps in the stator winding between consecutive stator segments, which must be taken into consideration by the control algorithm.

As previously stated, position sensors are required within the processing stations. Currently, there exists a multitude of linear position sensors, based on various physical principles (optical, inductive, capacitive, Hall effect, magnetoresistive, magnetostrictive, etc.) [6], [7]. However, not all of them are suitable for our application. The following requirements must be fulfilled by the position sensing system:

1. The vehicles are fully passive, i.e. there is no auxiliary energy source available on board of the vehicles;
2. Measuring accuracy of a few micrometers;
3. Measuring length of several meters;
4. Measuring interval of 100 µs or less;
5. The delay introduced in the position and speed control loops by the measuring system has to be as low as possible, in order to allow for a high dynamic load stiffness [8].

A detailed study [9], concerning different position sensing principles and commercially available position sensors was conducted at our department. Based on this study, two types of position sensors, suitable for linear drives with passive vehicles, were selected for evaluation: optical sensors and capacitive sensors.

In the transport sections of the linear drive for material handling described in this work there are no requirements for high dynamics or high positioning precision. In these sections, the complexity and costs introduced by position sensors will be avoided, by using sensorless motion control.

There are several sensorless methods for PMSM that are found in literature. They can be divided in two main classes:

- Methods based on the evaluation of the Electromotive Force (EMF) or of the mover's flux [10], [11], [12]; these methods are relatively robust, but have the disadvantage that they lose performance at low speed and do not work at standstill (i.e. they can be used for sensorless travelling at speeds higher than a minimal value, but they do not allow sensorless positioning).

- Methods based on the evaluation of the machine's anisotropies [13], [14], [15], usually by means of injecting a test signal. These methods are suited for standstill and low speed range.

When compared to the rotating PMSM, the segmented, long primary, linear ones present additional challenges with regard to the sensorless acquisition of the mechanical quantities:

- The translator (vehicle) of a long primary linear machine covers only partially the stator segments, which makes the extraction of the mechanical quantities from the measurable electrical ones, more difficult.

- The mechanical quantities must also be acquired at the transitions between stator segments.

All sensorless methods work incrementally, i.e. the absolute position cannot be determined without the knowledge of an absolute, start position. In the particular case of the discussed linear drives system, this start position can be determined when the vehicle passes through a sensor-equipped section of the track.

A topology of the power electronics and the control system is proposed in [16] for a similar linear drive. It consists on using two inverters, connected through static switches to the different segments of the track. This configuration, however, allows driving only one vehicle on the carriageway. Topologies that allow for multiple vehicles on the same track are discussed in [5].

Methods to deal with the vehicle transition from one segment to other are addressed in [17], [18] for the case of no gaps between segments. If there are gaps between segments proposals can be found in [19], [20].

In the previous proposals, the position is assumed to be a known variable. The usual approach for measuring the position is using a linear encoder with the read head in the vehicle and a scale on the track. The position signal is then transmitted to the stationary controller by a wireless modem, e.g. [18], [2]. This can be source of high delay in the control loop, and low reliability. In addition, this arrangement needs an auxiliary energy supply on board of the vehicle.

Using a magnetostrictive position sensor, as it is proposed in [4], the vehicle can be totally passive. However, the position information has a large delay for longer distances, which will not allow high dynamics in the control of movement (see also Section 2.1).

In this dissertation, a system and control method is proposed for a multi-segment linear drive with totally passive vehicles, which allows for gaps between consecutive segments, sensor based operation in some sections and sensorless based operation in others. The main contributions are:

- A novel method for measuring the vehicle position with high accuracy and high dynamics, using optical encoders, which method is suitable for use with passive vehicles.

- Two new and improved evaluation methods for a commercially available capacitive sensor that has a relatively lower resolution. This capacitive sensor could be used, in certain applications, as a less complex, and more cost effective alternative to the optical sensors.

- A method for handling the gaps and distinctive parameters among stator segments.

- A procedure that allows the smooth handover (transition) of the vehicles between the sensor-equipped, and the sensorless sections of the track.

This thesis is composed of two main sections: In Chapter 2, the two position sensing systems for passive vehicles, based on optical and capacitive sensors, respectively, are presented, and their performance is evaluated. In Chapter 3, the position acquisition and the control of the linear drive for material handling currently under development at our department is discussed, including the overall system architecture, the sensor-based position acquisition inside the processing stations, the sensorless position estimation in the transport sections of the track, and the synchronisation between the position given by sensors and the estimated one.

Finally, in Chapter 4, the conclusions are drawn, and future work, regarding the expansion of the functionality of the linear drive for material handling presented in this work, is outlined.

2 Position Sensing Systems for Passive Vehicles

2.1 Overview

The linear drive for material handling presented in this work consists of processing stations, connected by transport sections. Within the processing stations, highly precise positioning is necessary, thus the position of the vehicles must be acquired using position sensors. The measured position is used as feedback value of a high precision position control loop, while its approximate (numerical) derivative is used by the subordinated speed control loop.

Optical sensors are the industry standard for applications where high precision is required. In typical applications (e.g. machining), the passive scale of the sensors is mounted on the stationary side (track), while the active read-head is on the vehicle (see Figure 2.1). For passive vehicles, the passive scale must be attached to the vehicle and several active read-heads will be mounted along the track, as in Figure 2.2.

Incremental optical scales are commonly used with a narrow grading, in the micrometers – up to tens of micrometers range. Fine interpolation of sine/cosine signals usually provided by the optical read-heads will further increase the position resolution
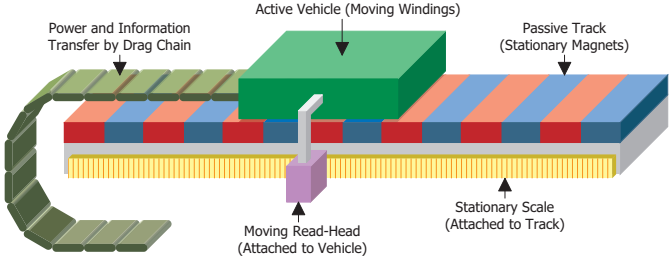

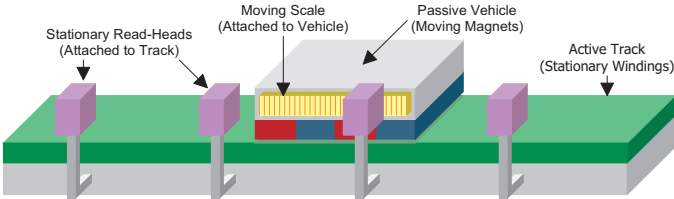**Figure 2.1: Position sensing with optical encoder and <u>active</u> vehicle**

Figure 2.2: Position sensing with optical encoder and <u>passive</u> vehicle
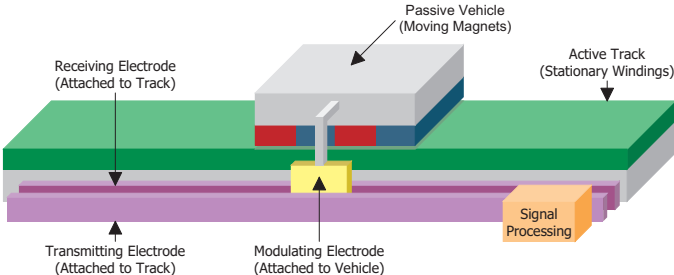

Figure 2.3: Position sensing with capacitive encoder and passive vehicle

(nanometers range).

The capacitive sensor schematically depicted in Figure 2.3 is a low-cost, low-complexity alternative to the optical system from Figure 2.2. The passive slider, attached to the vehicle, modulates (position-dependent) an electrical field produced between the stationary transmitting and receiving electrode; the position can then be extracted by demodulation.

Besides optical and capacitive sensors, there exist some more physical sensing principles, which may be interesting for the discussed application, e.g. [21] where a sonic strain pulse is induced in a specially designed magnetostrictive waveguide by the momentary interaction of two magnetic fields. One field comes from a movable permanent magnet fixed at the vehicle, as it passes along the outside of the sensor tube; the other field comes from a current pulse (interrogation pulse) applied along the waveguide. The interaction of the two magnetic fields produces a strain pulse, which travels at sonic speed along the waveguide until it is detected at the head of the sensor. The position of the magnet (= vehicle) is determined by measuring the elapsed time between the application of the interrogation pulse and the arrival of the resulting strain pulse [22].

These sensors have many interesting properties such as absolute measurement, possibility to simultaneously measure the position of several vehicles and to measure also along curved tracks. But, due to the travelling time of the sonic stain pulse, the above-mentioned requirements 4 and 5 are not fulfilled. The measuring cycle is 500 µs for a 1,2 m measuring distance and 1000 µs for 2,4 m [21], [23]. Therefore, these types of sensors are interesting for other applications, where such a delay in the control loop can be accepted.

The rest of this chapter is organised as follows:

In Section 2.2 a position sensing system for passive vehicles, based on optical sensors (as shown in Figure 2.2) will be described. The hardware, firmware and software developed for this position acquisition system will be presented, as well as experimental results regarding the performance of the optical system.

Section 2.3 presents the evaluation of the capacitive sensor (Figure 2.3). Firstly, an electrical model of the sensor will be derived. Then, two evaluation methods developed for the capacitive sensor (based on rectangular and, respectively, on sinusoidal excitations) will be presented. Experimental results for both excitation methods will also be shown and compared.

2.2 Optical Sensors System

2.2.1 General Description of the System

In this section, the implementation of an optical sensors system for use in linear drives with passive vehicles will be described. Optical sensors are the industry standard for position measuring when high precision positioning is required. The used sensor has an incremental scale with a grading (pitch) of 40 µm, and in order to further increase the resolution, the read-heads generate two analogue signals (sine and cosine) with a cycle length equal to the pitch. Fine interpolation, using arctangent function, will provide a position resolution (but not accuracy) in the nanometers range. The high resolution is necessary in order to generate a smooth speed signal (for the speed control loop) as the numeric derivative of position within a short sampling interval and without delay by filtering. The sensors also provide a reference signal, used for the zero-position detection. For a more detailed description of the sensors, see section 2.2.2.

Normally, the passive scale of the sensors has the same length as the maximal travelling distance of the linear machine and is mounted on the stationary side, while the active read-head is on the moving part. In the case of a passive mover, the mounting must be reversed, which means that the length of the sensor's scale is limited to the length of the mover. If a travelling distance longer than the mobile part is required, more than one read-head will be necessary.

In the proposed system, a number of five read-heads are spread along the carriageway ca. 210 mm apart from each other, so that the 250 mm long scale covers at any given position at least one read-head. Because it is not feasible to mount the heads at distances that are exact multiples of the 40 µm pitch, one must calculate and compensate for the difference between the phases of two neighbouring heads (as determined with the arctangent function). When the scale passes from one read-head to the next one, the signals from both heads will be simultaneously evaluated once, and the initial position of the incoming read-head can be calculated. Future position calculations are based on this initial value. Thus, a monotone variation of the calculated position along the entire measuring length can be ensured.

Figure 2.4 shows the block diagram of the proposed position acquisition system. The signals from the read-heads are brought to a pre-processing unit (Sensor to Digital
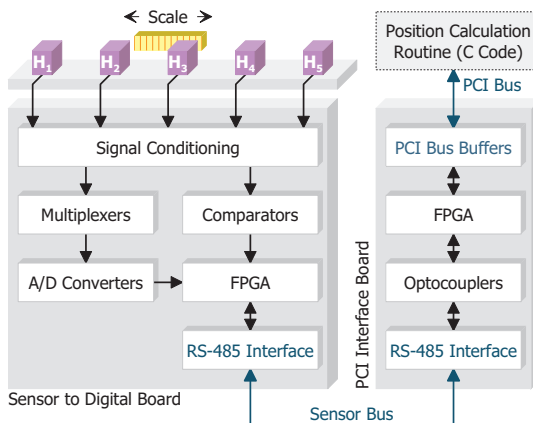

Figure 2.4: Block diagram of the optical position acquisition system

Board). The analogue sine and cosine signals from two neighbouring heads are fed, via multiplexers, into four analog-to-digital converters, where they are digitised, and sent further to an Field Programmable Gate Array (FPGA). The FPGA calculates the coarse position based on the sign bits of sine and cosine. It also keeps track of the current read-head and decides (based on the coarse position of the current read-head) when to start evaluating the signals from the next read-head. The reference signals from the first and last read-heads are also necessary, to detect when the vehicle enters the sensors covered region. They are fed, through comparators, to the Sensor to Digital Board's FPGA.

The pre-processing unit is connected through the Sensor Bus to a PCI Interface Board. The Sensor Bus is a 16 bit parallel bus, implementing the RS-485 differential signalling standard [24]. As its name suggests, the main purpose of PCI Interface Board is to provide an interface between the Sensor Bus and the Peripheral Component Interconnect (PCI) Bus [25] of a PC, where the control code (implemented in C language) is running. The second role of the PCI Interface Board is to galvanically isolate the PC from the rest of the position acquisition system, through optocouplers.

The position calculation routine is integrated in the control program and will be called in every control cycle (100 µs). When a call occurs, it will send a request, via the PCI Interface Board, to the pre-processing unit, which will send back the updated position information. Based on this information, the routine calculates a new position value and passes it to the control algorithm.

The detailed function of the component parts of the position acquisition system will be presented in the following sections.

2.2.2 Sensors Used

For the proposed system, the linear optical encoder LIDA 181 [26] from Heidenhain company (depicted in Figure 2.5) will be used. The encoder is based on imaging scanning principle (for details, see Appendix 5.1) and has a grating period (pitch) of 40 µm. The analogue sine and cosine signals (with a cycle length equal with the pitch) delivered by the sensor are differential, in order to reduce the influence of noise and have a magnitude of 1 V peak-to-peak. It must be noted that in order to function as specified, the sensors require a rather small gap between scale and read head (0.75 mm), with very small tolerance (±150 µm) and also small angular tolerances. This can be quite challenging with a linear machine where the guiding is not very stiff (see section 3.2). The most important datasheet specifications of the used optical sensors are summarised in Table 2.1.

Figure 2.6 shows the definitions for the positive moving direction and for the zero position. When the sensor moves in positive direction, the cosine signal has a phase of +90° with respect to the sine (according to the trigonometric definition); when the sensor moves in negative direction the cosine signal lags the sine (-90° phase). The periods are counted based on the zero crossings of the sine when the cosine is positive: crossing from 4th to 1st quadrant increases the period counter, while crossing from 1st to 4th quadrant decreases it. The reference pulses are centred in the middle of the 1st quadrant and can have a width between 180° and 540°. This means there is exactly one period count (up or down, depending on the moving direction) during a reference pulse. A read-head

Figure 2.5: LIDA 181 from Heidenhain. Source: [26]

| Specification	LIDA 181
Measuring principle	Imaging scanning
Measuring standard	Steel tape with AURODUR graduation
Gap between scale / read-head	0.75 ± 0.15 mm
Grating period	40 µm
Thermal expansion coefficient	10 ppm/K
Accuracy grade	± 5 µm
Measuring length	220 mm
Reference marks	Selectable by magnet every 50 mm
Maximal traversing speed	480 m/min
Vibration (55 to 2000 Hz)	≤ 200 m/s² (IEC 60 068-2-6)
Shock (11 ms)	≤ 500 m/s² (IEC 60 068-2-27)
Operating temperature	0 to 50°C
Power supply	5 V ± 5% / < 150 mA
Incremental signals	$\sim$ 1 V_{PP} / 40 µm

Table 2.1: Manufacturer specifications for LDIA 181. Source: [26]

generates two reference pulses, one close to each end of the scale.

The zero-position of the entire system is defined as the zero-position of the leftmost read-head (assuming left-to-right positive direction). The zero-position of the leftmost read-head is defined as the position where a zero crossing of the sine signal occurs, whilst the cosine signal is positive and a reference pulse is generated by the rightmost (sic!) part of the scale.

2.2.3 Pre-processing Unit (Sensor to Digital)

The block diagram of the pre-processing unit is shown in Figure 2.7. The differential signals from the five read-heads are first converted to single-ended ones. Analogue sine and cosine signals (S_1, C_1, ... S_5, C_5) are routed through two multiplexers (MUX_A and MUX_B) to the inputs of four 12-bit analog-to-digital converters (ADC-S_A, ADC-C_A, ADC-S_B, ADC-C_B). The outputs of the analog-to-digital converters are connected through the ADC-Bus to the board's FPGA, which also generates the control signals for the analog-to-digital
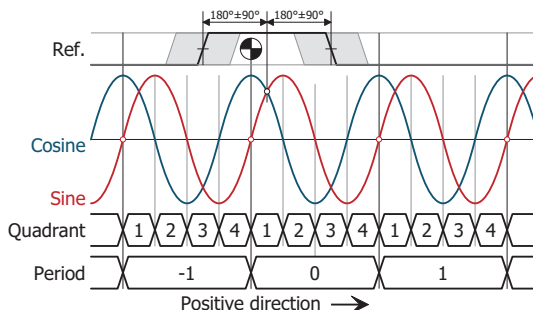
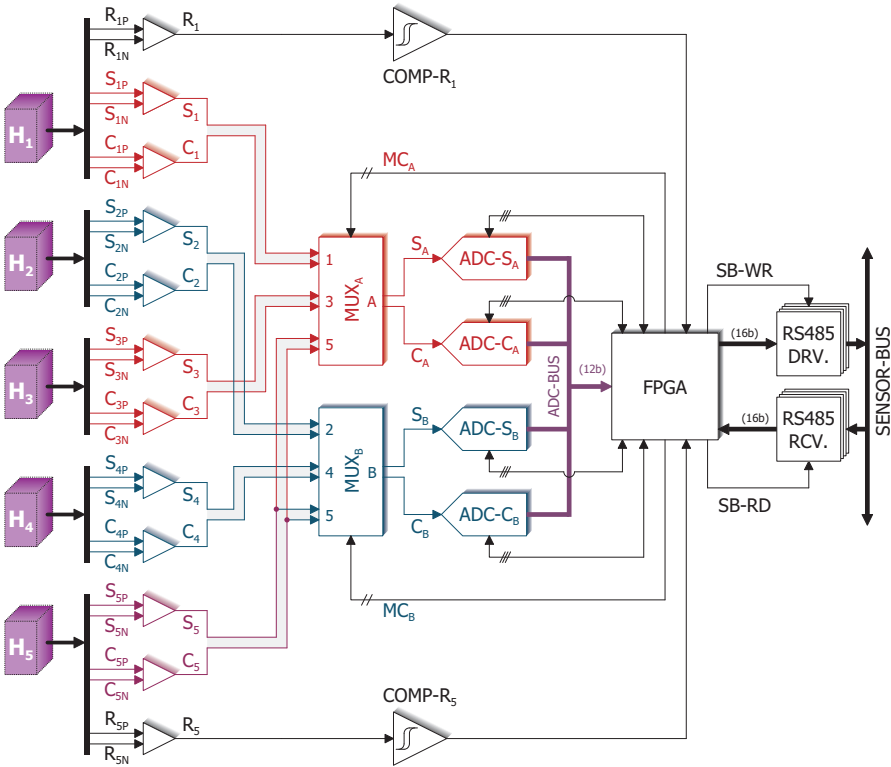**Figure 2.6: Definition of the positive direction and of the zero position**

Figure 2.7: Block diagram of the pre-processing unit

converters and multiplexers.

The essential function of the signal processing that occurs in the FPGA is to ensure that when the scale passes from one read-head to the next one no position information loss occurs. To achieve this, the four analogue signals of the neighbouring heads (S_A, C_A, S_B, C_B) will be sampled simultaneously, in order to correctly calculate their relative phase difference. The sampling rate of the analog-to-digital converters is set to 2 µs. This ensures that up to a maximal speed of 10 m/s there are at least 2 samples of the sine/cosine signals in each 40 µm grating period of the sensor, and the position acquisition algorithm can function correctly.

Reference signals from the first and from the last read-heads (R_1 and R_5) are also needed, in order to detect when the vehicle is entering the sensors' region. They are brought to the FPGA, through two comparators (COMP-R_1 and COMP-R_5).

At the beginning of every 100 µs control cycle, the control algorithm requests updated position information through the 16-bit Sensor Bus. This bus is connected to the board's FPGA through RS-485 drivers and receivers. Detailed description of the communication protocol on Sensor Bus is given in section 2.2.4.

The vehicle can enter the sensors' region from left or from right, so the sine and cosine signals from read-heads 1 and 5 must be simultaneously available. This is achieved when the signals from head 1 come through MUX$_A$ (MC$_A$ = 1), and the ones from head 5, through MUX$_B$ (MC$_B$ = 5). On the other hand, the signals from read-head 4 are

routed through MUX$_B$, so when the vehicle is travelling in positive direction, at the transition from head 4 to head 5, the sine and cosine signals of the last must be available through MUX$_A$ (otherwise it would not be possible to sample the four signals simultaneously). This is why S$_5$ and C$_5$ are routed through both MUX$_A$ and MUX$_B$.

The distance covered by read-head 5 will be split in two: in the first half, the signals from MUX$_A$ will be used, and in the second half, the signals from MUX$_B$. So, to the physical read-head 5 correspond two logical read-heads "5A" and "5B", each covering half the distance covered by head 5. At the transition between the two logical heads the position phase difference is zero. In the following, heads 5A and 5B will be treated in the signal processing as two different read-heads.

The realisation of the pre-processing unit (Sensor to Digital Board) is shown in Figure 2.8. The five optical sensors are connected to the board through connectors (1) … (5). In the signal conditioning part (6), the sensors signals are converted to single-ended, their gain is adjusted, and the high frequency noise is filtered. The sine and cosine signals are than routed, through the multiplexers (7) to the four analog-to-digital converters (8), whilst the analogue reference signals from heads 1 and 5 are digitalized using the two comparators (9). The threshold values of the comparators are adjustable, so that the reference pulses can be kept in the limits defined by Figure 2.6.

The outputs of the comparators, as well as the ones from the analog-to-digital converters are brought to the FPGA (10). EPROM (11) is used to store the FPGA firmware when the power is off, and to automatically program the FPGA at power-up. The FPGA can also be programmed using the connectors (13) or (14). An 80 MHz oscillator (12) generates the clock signal. Auxiliary input/output port (15) can be used for debugging. In the area (16) are the RS-485 drivers and receivers for the Sensor Bus, and (17) is the connector for the Sensor Bus cable.

The board requires +5 V and –5 V power supplies for the analogue circuitry, and a separate 5 V supply for the digital one. These are supplied to the board using the connector (18). Additionally, the FPGA also needs 3.3 V and 1.5 V supplies, which are
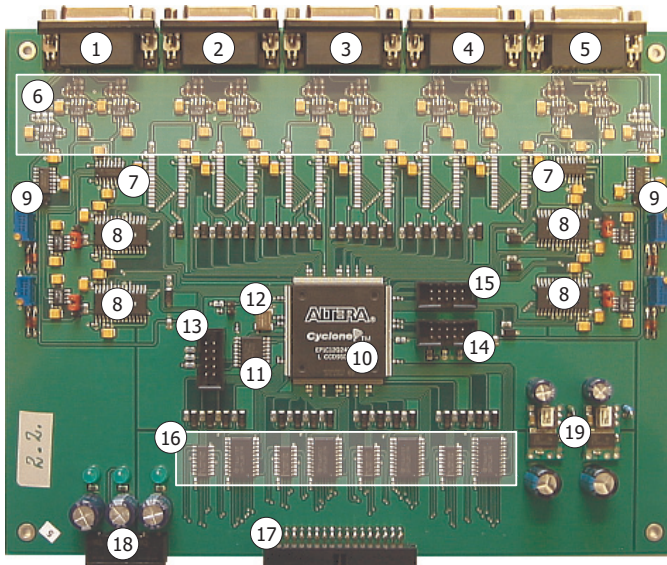

Figure 2.8: Realisation of the preprocessing unit (Sensor to Digital Board)

generated on the board, using the DC/DC converters (19).

The distances covered by each read-head are shown in Figure 2.9. The scale has a length of 250 mm, and the read-heads are mounted along the track at approximately every 210 mm. Considering the grating period (Pitch) $P = 40$ µm of the sensor, each of the heads 1 … 4 will cover a number $\Delta N = 5250$ periods. Head 5A covers $\Delta N/2 = 2625$ periods, whilst head 5B covers ΔN_{5B} periods. If the heads would be mounted exactly ΔN periods apart from each other, the relation $\Delta N_{5B} = \Delta N/2$ would hold true, but in reality a small difference exists (on the experimental setup a difference of 5 periods was identified). ΔN_{5B} can be determined during a test-run in the positive direction, as the value of the "B" period counter on the falling edge of the second pulse of the reference signal R_5 (see below). If instead of the real value ΔN_{5B}, the ideal value $\Delta N/2$ is used, a hysteresis of the determined position occurs between runs in positive and negative directions.
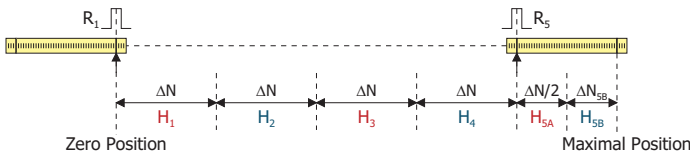

Figure 2.9: Distances between read-heads

The values ΔN, $\Delta N/2$, and ΔN_{5B} will be used by the FPGA to decide when the transition between one read-head and the following one must be made.

The firmware in the Sensor to Digital FPGA (a simplified diagram of which is shown in Figure 2.10) can be divided in seven major functional blocks:

1. **ADC Control** – controls the four analog-to-digital converters.
2. **Sin/Cos Registers** – store the conversions' results.
3. **Counter A Control** – controls a 14-bit counter (CNTRA) which counts the periods (coarse position) of the read-head currently routed through MUXA, based on the sign bits of SA and CA.
4. **Counter B Control** – identical with Counter A Control, CNTRB counts based on the sign bits of SB and CB.
5. **Current Head Control** – controls the Current Head State Machine (CHSM), which keeps track of the read-head(s), currently covered by the scale. Additionally, this block generates the control signals for the two multiplexers (MCA and MCB).
6. **Sensor Bus Control** – implements the interface with the Sensor Bus
7. **Synchronisation** – generates the timing and the synchronisation signals for all previous blocks.

It is assumed that initially the vehicle is outside the sensors' region. The Current Head State Machine is in state H_0 (signifying that no read-head is covered by the scale). The CHSM uses two 3-bit buses, $LD_A(3)$ and $LD_B(3)$ to control the behaviour of the two period counters $CNTR_A(14)$ and $CNTR_B(14)$. The possible values for LD_A and their meaning are given in Table 2.2. Those of LD_B are similar, with the only difference that for $LD_B = 100_b$ the value ΔN_{5B} instead of $\Delta N/2$ is loaded in $CNTR_B$. In the initial state, $LD_A = 001_b$ and $LD_B = 100_b$, meaning that the two counters will be blocked, holding the values $N_A = -1$ and $N_B = \Delta N_{5B}$, respectively.

Multiplexer code register $R\text{-}MC_A(2)$ holds the value $MC_A = 1$, so that $S_A = S_1$, $C_A = C_1$, and $R\text{-}MC_B(2)$ – the value $MC_B = 5$ ($S_B = S_5$, $C_B = C_5$). Registers $R\text{-}USE_A$ and $R\text{-}USE_B$ indicate whether position information from heads "A" or "B", respectively, is valid and should be used for position calculation. Initially they are both reset – neither of the active heads (1 and 5B) is providing valid signals. The Conversion Timer (in the Synchronisation
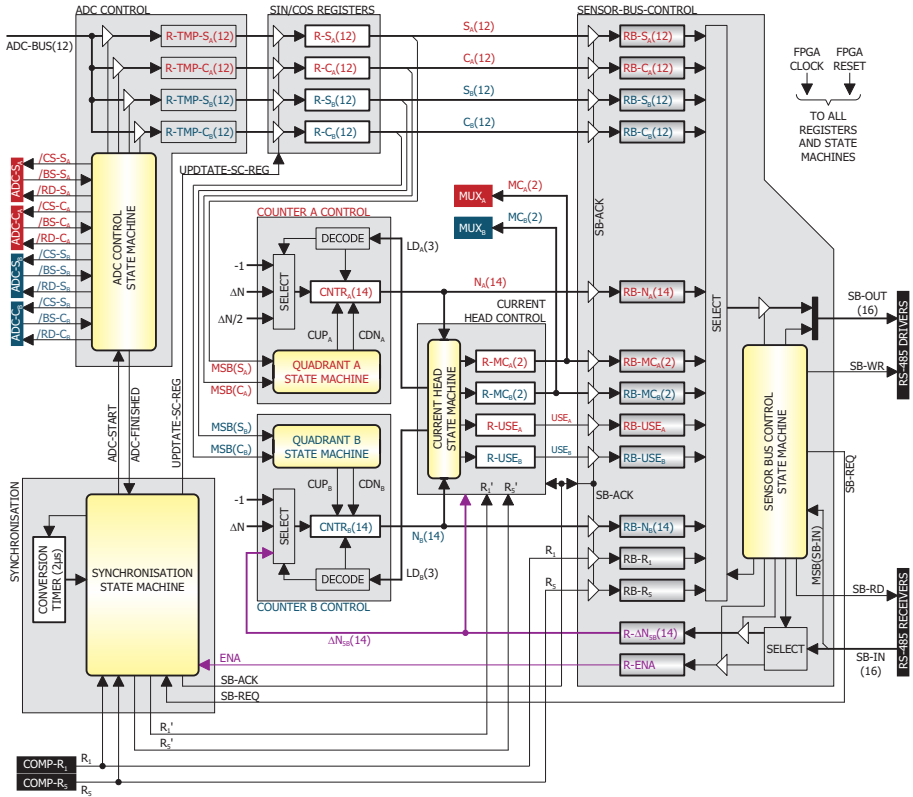
Figure 2.10: Simplified diagram of the pre-processing unit FPGA code

Block) is stopped, and no analog-to-digital conversions take place. The system remains in this state until one of the comparators' outputs (R_1 or R_5) is asserted, signalling the entering of the vehicle into the sensors' region (from left or from right).

Let us assume that the vehicle comes from left, thus travelling in the positive direction. The synchronisation logic detects the rising edge of R_1 and asserts the signal ADC-START and simultaneously starts the 2 µs Conversion Timer. It then waits for the first conversion to end.

When the ADC-START signal is asserted, the ADC Control State Machine immediately asserts the four conversion start signals (/CS-S_A, /CS-C_A, /CS-S_B, /CS-C_B). The conversions are finished after a maximum time of 750 ns [27], which is indicated by

$LD_A(3)$	Signification	
000_b	$CNTR_A$ is enabled and counts based on $MSB(S_A)$ and $MSB(C_A)$	
001_b	Value −1 is loaded in register $CNTR_A$	Changes in $MSB(S_A)$ and $MSB(C_A)$ have no influence on the counter's value
010_b	Value ΔN is loaded in register $CNTR_A$	
100_b	Value $\Delta N/2$ is loaded in register $CNTR_A$	
Other	Illegal	

Table 2.2: Possible values of LD_A

the de-asserting of the four busy signals (/BS-S_A, ... /BS-C_B). At this point, the four read signals (/RD-S_A, ... /RD-C_B) are used to sequentially bring the conversions' results into four temporary registers R-TMP-S_A(12), ... R-TMP-C_B(12). This sequential reading through the ADC-BUS(12) takes 200 ns. After that, the ADC Control State Machine asserts the ADC-FINISHED signal, determining the Synchronisation State Machine to assert the UPDATE-SC-REG signal, thus simultaneously updating the content of registers R-S_A(12), R-C_A(12), R-S_B(12), and R-C_B(12).

Based on the new values of S_A and C_A (their sign bits) the Quadrant A State Machine will be set in the correct state. The first conversion was made soon after the rising edge of R_1. This means that the quadrant determined by S_A and C_A can be 2, 3 or 4, depending on the width of R_1 (see Figure 2.6). As previously stated, $CNTR_A$ is blocked at the value –1 by CHSM. Because the scale does not cover head 5B, the values of S_B and C_B are at this point irrelevant.

Two clocks after the UPDATE-SC-REG signal was asserted (leaving time for the sine/cosine registers and for quadrant state machines to stabilize), the synchronization state machine asserts R_1', leading to the transition of the CHSM from state H_0 into state $H_{1A(1)}$. The complete Current Head State Machine is shown in Figure 2.11. In state $H_{1A(1)}$, $LD_A = 000_b$, thus $CNTR_A$ is enabled and the value "1" is stored in R-USE$_A$, indicating valid position information from head "A" (in this case 1). The multiplexers' control signals remain unchanged.

At this point the content of all the position information registers (sine/cosine registers, period counters, multiplexer code registers and "use" registers) is consistent, indicating the new state of the system: the vehicle is at the beginning of the region covered by of head 1, and its position can be correctly calculated based on the "A" values. The "B" position information will be ignored until the scale covers a read-head whose signals are routed through the "B" multiplexer.

The reference signal R_1 (or R_5, if entering from right) is used only to trigger the first conversion cycle and, when this first conversion is finished, the first transition in CHSM. All the subsequent reference pulses are ignored, until the vehicle is again outside the sensors' region.

When the conversion timer (started at the same time with the first conversion) hits 2 µs a new conversion is started and, after it is finished, new values of S_A, C_A, S_B, and C_B will be available, leading to the update of the quadrant state machines and of the period counters and, based on the updated counters' values to the update of the state of CHSM. During this update time, the position information can be inconsistent. The redundant temporary sine/cosine registers in the ADC Control block where introduced to minimize this inconsistency time. Instead of 200 ns (as it takes to sequentially update the four conversion results through the ADC Bus), the time for updating of the sine/cosine registers (R-S_A, ... R-C_B) is reduced to one FPGA clock (12.5 ns). Adding to this time the time necessary to update the quadrant state machines, the period counters, and the CHSM, an inconsistency time of 50 ns results. Any data request coming from Sensor Bus during this time will be delayed until the position information registers are consistent again, and the delay must be short enough not to impair the timing requirements of the Sensor Bus protocol.

When the Sensor Bus Control State Machine receives a position information request, it asserts the SB-REQ signal. The Synchronisation State Machine responds by asserting the acknowledge signal SB-ACK, with a delay no longer than 50 ns, if necessary, as discussed above.
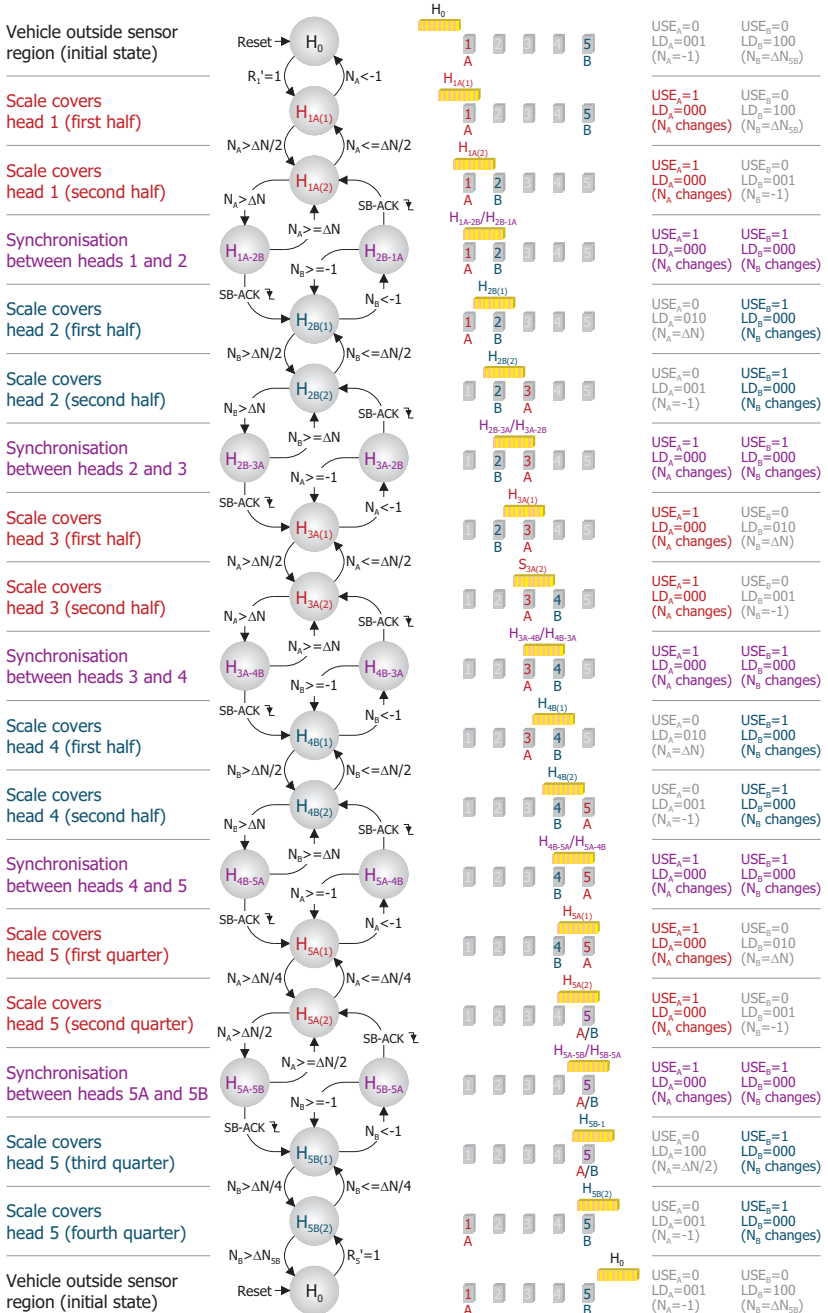
Figure 2.11: Current Head State Machine (CHSM)

On the rising edge of SB-ACK, the content of the position information registers is transferred into a set of corresponding bus registers:

- Sine/cosine registers   →   RB-SA(12), RB-CA(12), RB-SB(12), RB-CB(12);
- Period counters   →   RB-NA(14), RB-NB(14);
- Multiplexers' codes   →   RB-MCA(2), RB-MCB(2);
- Used (valid) head   →   RB-USEA, RB-USEB;
- Reference signals   →   RB-R1, RB-R5.

From these registers the position information will be sent further on the Sensor Bus. The position information is updated in the Sensor to Digital FPGA every 2 μs. The content of the bus registers listed above is a consistent snapshot of this information, taken every 100 μs, when it is required by the control.

The data stored in registers R-ENA and R-ΔN_{5B}(14) is coming from the control program. Register R-ΔN_{5B}(14) contains the real number of periods covered by head 5B (as determined during a test-run). Its contents will be written only once, when the control program starts. A zero written in R-ENA will determine the Synchronisation State Machine to ignore the reference signals, thus effectively disabling all the position acquisition related firmware in the Sensor to Digital FPGA. This behaviour is useful for the initialisation phase, when the control must ensure that the vehicle is outside sensors' region before enabling the Sensor to Digital algorithm.

State H_0 corresponds in the CHSM to the vehicle being outside sensor's region. For each of the six logical read-heads (1 … 4, 5A, 5B) there are two states. The only difference between the two states is in the multiplexers' control signals. For example, in state $H_{1A(1)}$, the signals from head 1, coming through MUX$_A$, are used to determine the position. In this state MC$_B$ = 5. When the middle of the scale passes the head (N$_A$ > ΔN/2), a transition to state $H_{1A(2)}$ occurs, where MC$_B$ = 2. CNTR$_B$ continues to remain blocked, but LD$_B$ changes to 001$_b$, so the value –1 will be now loaded into the counter. Thus, the signals from head 2 are now coming through MUX$_B$ to the analog-to-digital converters and are ready for evaluation. The multiplexers' codes corresponding to each state of CHSM are depicted on the diagrams on the right of the state machine in Figure 2.11. The values of USE$_A$, USE$_B$, LD$_A$ and LD$_B$ are also listed for each state.

When in state $H_{1A(2)}$ and N$_A$ > ΔN, there is no direct transition to state $H_{2B(1)}$. Because the control program must determine the offset of head 2, the position information of both heads, from the same 2 μs sample, must be sent through the Sensor Bus. So, when N$_A$ > ΔN, the transition to state H_{1A-2B} occurs. In this state LD$_B$ is set to 000$_b$ (so CNTR$_B$ is also enabled), and the position information from both heads is now valid (indicated by USE$_A$ and USE$_B$ being both set). This state is maintained until after a read request from the Sensor Bus is acknowledged, i.e. until the falling edge of SB-ACK comes. When this happens, the CHSM goes into state $H_{2B(1)}$, where only the "B" position information will be used. The transition between $H_{2B(1)}$ and $H_{1A(2)}$ (vehicle travelling in the negative direction) happens similarly, passing in this case through state H_{2B-1A}.

Some examples of transitions of the CHSM states will be illustrated in the last part of this section, based on the CHSM representation in Figure 2.11, and on the simplified signal diagrams in Figure 2.12:

a) Entering sensor's region, moving in positive direction

Initially, CHSM is in state H_0 and CNTR$_A$ is blocked to –1. When the rising edge of R$_1$ comes (time t$_1$), the sampling of S$_A$ = S$_1$ and C$_A$ = C$_1$ begins. At time t$_2$ (when the first conversion is finished), CHSM changes state to $H_{1A(1)}$ and CNTR$_A$ is enabled. At time t$_3$ (after passing through the zero position of the sensors' region) CNTR$_A$ counts up, yielding N$_A$ = 0, and continues counting based on the sign bits of SA and CA.
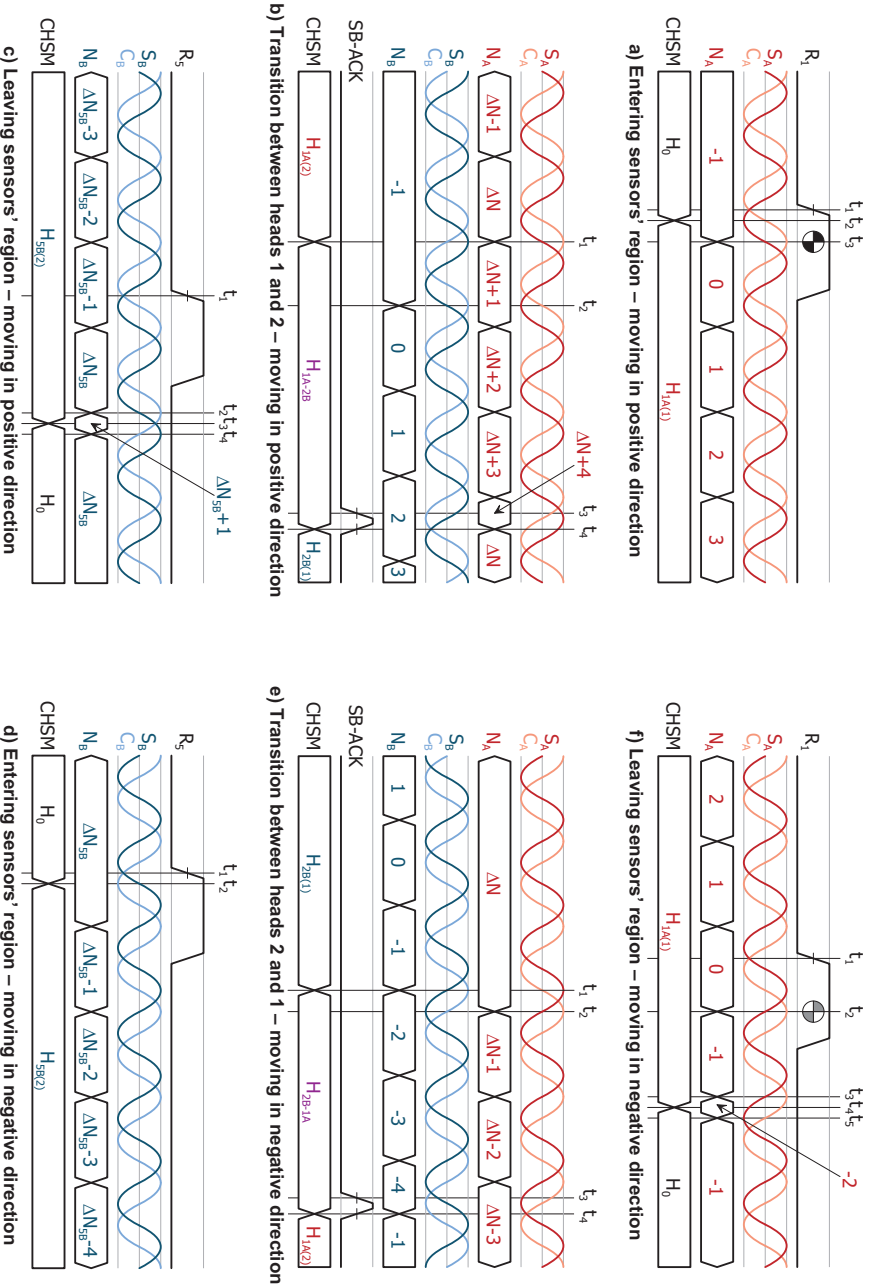
Figure 2.12: Examples of state transitions in Current Head State Machine

a) Entering sensors' region – moving in positive direction

b) Transition between heads 1 and 2 – moving in positive direction

c) Leaving sensors' region – moving in positive direction

d) Entering sensors' region – moving in negative direction

e) Transition between heads 2 and 1 – moving in negative direction

f) Leaving sensors' region – moving in negative direction

Any reading of the position information before t_2 will result in $USE_A = 0$ and $USE_B = 0$ (vehicle outside sensor's region). If the position information is read between t_2 and t_3, the control program will calculate a negative value of the position.

b) <u>Transition between heads H_1 and H_2, moving in positive direction</u>

The vehicle is travelling the region covered by head 1. When it approaches the end of this region, the scale will also cover head 2. CHSM is in state $H_{1A(2)}$, and $CNTR_A$ is counting. The sine/cosine signals from head 2 are coming through MUX_B, but $CNTR_B$ is blocked at −1 (not counting). When $N_A > \Delta N$ (at time t_1), CHSM switches to state H_{1A-2B} and $CNTR_B$ is enabled. At time t_2, the quadrant defined by S_B and C_B changes from 4 to 1 and $CNTR_B$ counts up, yielding $N_B = 0$. The two counters are now both counting in parallel, until t_3, when the rising edge of SB-ACK arrives. The position information is saved into the bus registers, and on the falling edge of SB-ACK (time t_4) CHSM changes state to $H_{2B(1)}$. $CNTR_B$ continues counting, but $CNTR_A$ is blocked, holding $N_A = \Delta N$. State H_{1A-2B} can last up to 100 µs (the interval between two successive readings on the Sensor Bus), which translates in up to 25 position periods (1 mm) at the maximum speed of 10 m/s.

c) <u>Leaving sensors' region, moving in positive direction</u>

The scale covers head 5B, CHSM is in state $H_{5B(2)}$ and $CNTR_B$ is counting. At time t_1 the reference signal R_5 comes, but this has no influence on the Synchronisation block. The position acquisition algorithm in the Sensor to Digital FPGA will ensure that the transition to state H_0 will occur after the falling edge of R_5 based solely on the value of $CNTR_B$. If the reference signal respects the tolerances defined in Figure 2.6, the falling edge of R_5 will come at the latest in the middle of quadrant 4 of period ΔN_{5B}, so the state transition in CHSM will be triggered by $N_B > \Delta N_{5B}$. At time t_3, $CNTR_B$ counts up and N_B becomes $\Delta N_{5B}+1$. One FPGA clock later (time t_4) CHSM switches to state H_0, setting $LD_B = 100_b$ and another clock later (time t_5) $CNTR_B$ is blocked, outputting $N_B = \Delta N_{5B}$.

d) <u>Entering sensor's region, moving in negative direction</u>

The vehicle is outside of the sensors' region, moving in negative direction, and approaching read-head 5 from the right (i.e. logical head 5B). CHSM is in state H_0. At time t_1 the rising edge of R_5 comes and all four sine/cosine signals are sampled. In this case $S_B = S_5$ and $C_B = C_5$ are relevant. After first conversion, CHSM changes to state $H_{5B(2)}$, at time t_2. $CNTR_B$ is enabled (before it was blocked to ΔN_{5B}) and position information "B" is now valid.

e) <u>Transition between heads H_2 and H_1, moving in negative direction</u>

The scale covers heads 2 and 1, with the vehicle moving in negative direction. CHSM is in state $H_{2B(1)}$, $CNTR_B$ is counting down, and $CNTR_A$ is blocked at $N_A = \Delta N$. When $N_B < -1$ (time t_1), the transition to state H_{2B-1A} occurs and $CNTR_A$ is enabled ($LD_A = 000_b$). At t_2 quadrant A changes from 1 to 4, and $CNTR_A$ counts down ($N_A = \Delta N-1$). The two counters run in parallel and the position information from both heads is ready for evaluation until t_3, when, on the rising edge of SB-ACK, it is copied to the bus registers. On the falling edge of SB-ACK (t_4) the transition to $H_{1A(2)}$ occurs, $CNTR_B$ is disabled and loaded with $N_B = -1$, the "B" position information is invalidated ($USE_B = 0$), and the position evaluation will continue based only on the "A" values.

f) <u>Leaving sensors' region, moving in negative direction</u>

The vehicle is close to the left limit of the region covered by head 1, moving in negative direction. CHSM is in state $H_{1A(1)}$ and $CNTR_A$ counts down. At t_1 the rising edge of R_1 occurs, but it has no effect. At time t_2 the zero position of the sensors system is crossed towards negative values. At t_1, N_A becomes less than −1, so at t_4 (one FPGA clock later) CHSM changes state to H_0, and another clock later −1 is stored in $CNTR_A$. The system is now again in its initial state, waiting for one of the two reference signals.

2.2.4 Sensor Bus Communication

The Sensor Bus connects the Sensor to Digital Board with the PCI Interface Board. It is a 16-bit parallel bus, implementing the RS-485 differential signalling standard. Line 15 (the most significant bit) of the bus is used as control signal, whilst the other 15 lines (14 … 0) are used as data lines. So, in one transfer (frame) 15 information bits can be sent simultaneously through the bus. At one given time, one of the two communication parties drives all the lines of the bus (control and data). This signalling technique was adopted because it allows for a frame length that is not dependent on the delays on the bus (delays caused by the RS-485 drivers and receivers, the propagation delay on the bus cable, and especially delays caused by the optocouplers on the PCI Interface Board).

Figure 2.13 shows the transfer protocol between the Sensor to Digital and the PCI Interface Board. In the upper part of the figure the control and data signals are sketched as "seen" from the PCI Interface FPGA, while in the lower part as "seen" from Sensor to Digital FPGA. Initially, the PCI Interface drives the bus, holding all the lines at logical 1. When required from the control program (time t_0) it starts sending a data request to the Sensor to Digital Board. Data frame D_0 (sent from control program) accompanies this data request. The content of D_0 is listed in Table 2.3: bit 14 (ENA) will be saved in the R-ENA register of the Sensor to Digital FPGA and will enable or disable the position acquisition algorithm. Bits 13 … 0 contain the real number of periods covered by head 5B. The sign bit of ΔN_{5B} is not sent on the bus (it is always 0), but instead the SET_{5B} bit will be sent. This bit is used to save ΔN_{5B} in R-ΔN_{5B} only once, at the beginning of the control program.

Frame D_0 is held on the bus until t_3 (400 ns). In this time the control line is held low for 200 ns and then back high for the remaining 200 ns. This line is acting as bus clock signal: on its falling edges the communication party driving the bus lines writes data on the bus, whilst on its rising edges the other party reads the data. The low and high pulses of
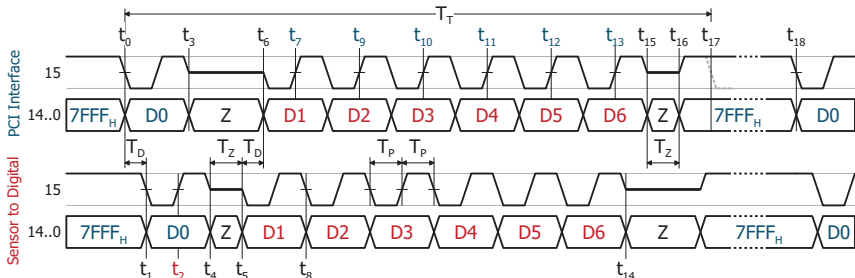

Figure 2.13: Sensor Bus transfer protocol

Frame / Bit	D_0	D_1	D_2	D_3	D_4	D_5	D_6
14	ENA	USE_A	0	0	USE_B	0	0
13	SET_{5B}	R_1	MC_A (2)	N_A (14)	R_5	MC_B (2)	N_B (14)
12	ΔN_{5B} (13)	0			0		
11 … 0		S_A (12)	C_A (12)		S_B (12)	C_B (12)	

Table 2.3: Sensor Bus data frames

the control line are 200 ns (T_P) long, each.

After the time t_3, the PCI Interface stops driving all the bus lines, setting the bus drivers in high impedance state (Z). The Sensor to Digital FPGA detects the falling edge of the control signal at time t_1, after the delay time T_D. Measurements have determined that on the experimental setup this delay does not exceed 130 ns. The bus delay has no influence on the pulse period T_P, but must be known, for the Sensor Bus protocol to function correctly. The data will be saved in the Sensor to Digital FPGA registers at time t_2, on the rising edge of the control signal. Simultaneously, a data request is signalled, by asserting the SB-REQ signal. Between t_4 and t_5 the bus lines are not driven by any of the two parties. The high impedance state lasts for $T_Z = 200$ ns. Between t_2 and t_5 the data request must be acknowledged (by the Synchronisation block in the Sensor to Digital FPGA), and the position information saved in the bus registers. The contents of the bus registers will be sent on the Sensor Bus in six data frames $D_1 \ldots D_6$, which are also listed in Table 2.3.

At time t_6, after another delay period T_D, data frame D_1 will be "seen" by the PCI Interface FPGA and it will be saved at time t_7. The remaining data frames will also be sent and saved in the PCI Interface FPGA registers at t_9, t_{10}, $\ldots$ t_{13}. At t_{14}, the Sensor to Digital has sent all the position information and stops driving the bus lines. After receiving the last data frame (t_{15}), the PCI Interface does not immediately start to drive the bus; a second high impedance state occurs between t_{15} and t_{16} and afterwards the PCI Interface drives all the bus lines at logic 1.

It is possible to initiate a new transfer as soon as t_{17}. The total transfer time on the Sensor Bus amounts to $T_T = 2 \cdot T_P \cdot (1+6) + 2 \cdot T_Z + 2 \cdot T_D$, which is about 3.5 µs. The bus will remain in the idle state (all lines driven to "1" by PCI Interface) until a new position request comes from the control program (the time interval $t_0 \ldots t_{18}$ in Figure 2.13 is the 100 µs control cycle).

2.2.5 PCI Interface Board

The PCI Interface Board assures the data transfer between the Sensor Bus and the PCI Bus of the control PC. It does not alter in any way the position requests from control program, nor the position information sent by the Sensor to Digital Board, acting solely as a data transfer relay. It also provides galvanic separation between the control PC and the position acquisition electronics on the Sensor to Digital Board. The realisation of the board is shown in Figure 2.14:

  (1) PCI Interface FPGA;
  (2) EPROM, storing the FPGA firmware;
  (3) 80 MHz oscillator;
  (4) PCI Bus buffers;
  (5) PCI connector;
  (6) JTAG programming connector;
  (7) Active Serial programming connector;
  (8) Auxiliary inputs/outputs, used for debugging;
  (9) 5 V and 3.3 V power supplies, connected to those coming through PCI connector;
  (10) DC/DC converter, for the 1.5 V supply required by the FPGA;
  (11) Optocouplers, separating the potentials of PC and Sensor to Digital Board;
  (12) Sensor Bus drivers and receivers;
  (13) Sensor Bus connector;
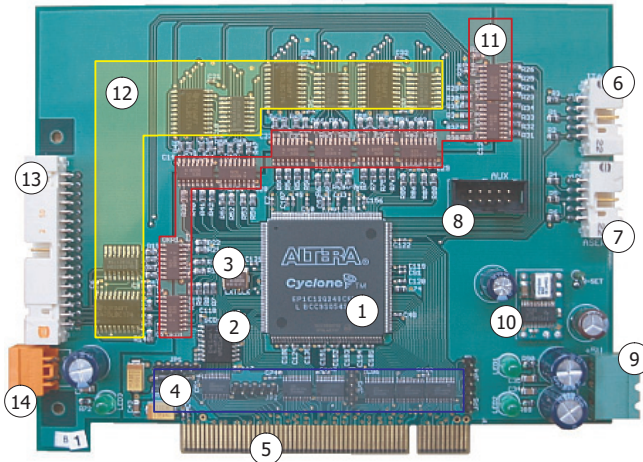 (14) Separate 5 V supply for the Sensor Bus drivers and receivers.

Figure 2.14: PCI Interface Board

Figure 2.15 depicts a simplified block diagram of the PCI Interface FPGA firmware. The „PCI Interface Function" block, provided by the FPGA's manufacturer, handles the PCI bus protocol. Data coming from the PCI bus is available on the 32-bit Local Data Input bus (LDI), and data can be sent on the PCI bus using the 32-bit Local Data Output bus (LDO). The Data Transfer Control State Machine is responsible for decoding the PCI bus addresses, received from the „PCI Interface Function" block through ADDR(32). A number of five addresses are recognized:

- **ADDR-D_0**: The control program writes at this address the data frame D_0, as defined in Table 2.3.
- **ADDR-D_1D_2**, **ADDR-D_3D_4**, and **ADDR-D_5D_6**: The control program reads from these addresses the position information, as sent from the Sensor to Digital Board in the six data frames $D_1 \ldots D_6$. Two 15-bit Sensor Bus data frames can be read in a single 32-bit PCI transfer.
- **ADDR-STAT**: This address gives to the control program read access to register R-
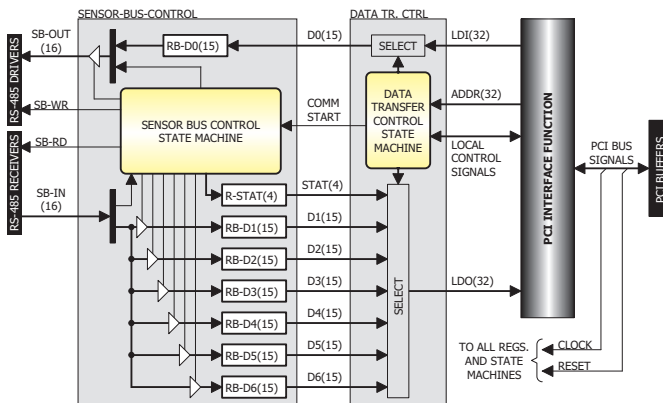

Figure 2.15: Simplified diagram of PCI Interface FPGA Firmware

STAT(4), which contains the state of the Sensor Bus Control State Machine (from the PCI Interface Firmware), and implicitly the Sensor Bus transfer status.

At the beginning of the control cycle, the control program writes on the PCI bus the data frame D_0, at its address ADDR-D_0. The Data Transfer Control State Machine decodes the address and consequently the data is saved in RB-D_0(15), from where it will be sent on the Sensor Bus. Concomitantly, it asserts the communication start signal (COMM-START), signalling to the Sensor Bus Control State Machine to initiate a new transfer on the Sensor Bus. The Sensor Bus Control State Machine handles the communication protocol on the Sensor Bus and saves the position information coming from the Sensor to Digital board in registers RB-D_1(15) … RB-D_6(15).

The control program can check at any time the status of the transfer, by reading the contents of R-STAT(4). When the Sensor Bus Control State Machine is again in idle state (meaning that the position transfer is completed), the control program can read the position information in three 32-bit PCI accesses and, based on this information, it will determine a new value of the position.

2.2.6 Position Reconstruction in C Code

The position of the vehicle is reconstructed from the information sent through the Sensor Bus in the position calculation function of the control program. This function is called every 100 µs, when new values of USE_A, MC_A, S_A, C_A, N_A and of USE_B, MC_B, S_B, C_B, N_B are available. A software-implemented state machine (Figure 2.16) uses USE_A, USE_B, MC_A and MC_B to keep track of the current head, in parallel with the Current Head State Machine (CHSM) from the Sensor to Digital firmware. One state of the software state machine is allocated for each of the six logical read-heads, plus the state H'$_0$, when the vehicle is outside the sensors' region and no position information is available. Figure 2.17 depicts the principle of the state changes in the position reconstruction state machine, and of the calculation of the position "x".

Initially, the vehicle is outside the sensors' region. This is signalled from Sensor to Digital by $USE_A = 0$ and $USE_B = 0$. The reconstruction state machine is in state H'$_0$, and the position x cannot be calculated. At time t_0 (Figure 2.17), the Sensor to Digital board sends for the first time $USE_A = 1$. This means that valid position information from head 1 (in this case A) is available. Firstly, based on S_A, C_A and N_A, the local position given by head A is calculated:

$$x_A(t_0) = P \cdot N_A(t_0) + x_{inc,A}(t_0),$$

<div align="right">Eq. 2.1</div>

where P is the pitch of the sensor and $x_{inc,A}(t_0)$ is the incremental position of head 1, as calculated based on $S_A(t_0)$ and $C_A(t_0)$ using the arctangent function:

$$x_{inc,A}(t_0) = \frac{P}{2\pi} \cdot atan2(S_A(t_0), C_A(t_0)).$$

<div align="right">Eq. 2.2</div>

The atan2 function returns a value between $-\pi$ and π. In 3rd and 4th quadrants ($S_A < 0$) its result must be corrected by adding 2π (or, equivalently, by adding P at the incremental position):

$$if(S_A(t_0) < 0)$$
$$x_{inc,A}(t_0) = x_{inc,A}(t_0) + P$$

<div align="right">Eq. 2.3</div>

The variable x_0 will be used to store the offset of the current read-head with respect to the zero position of the system. Because zero position of the system was defined as the zero position of the head 1, its offset will be zero:
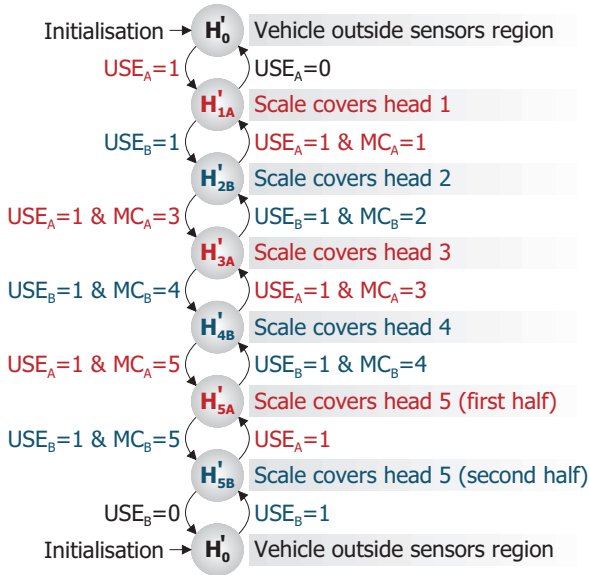
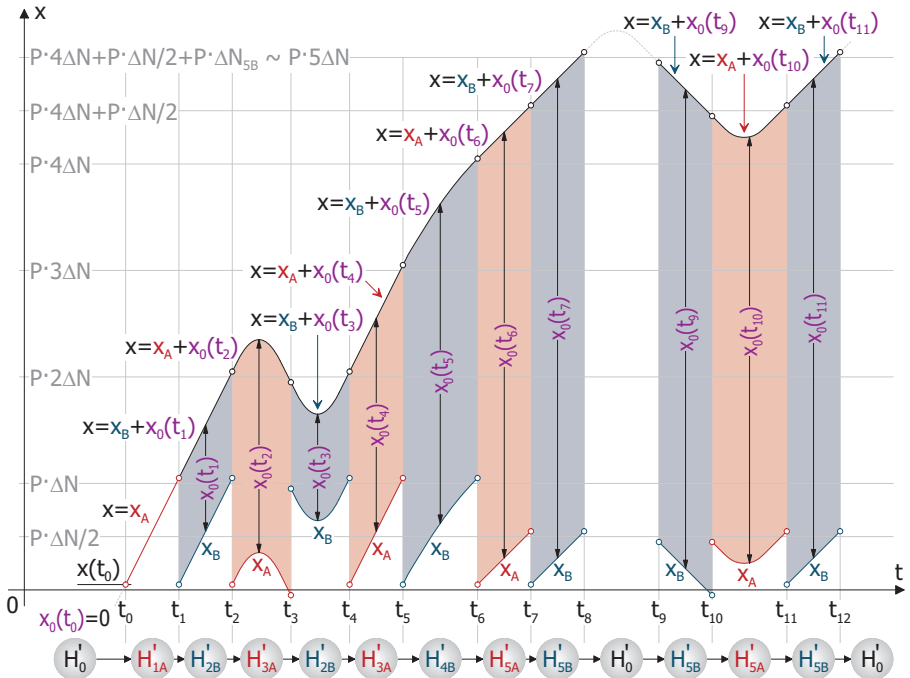**Figure 2.16: Reconstruction state machine**

Figure 2.17: Illustration of the principle of position reconstruction in C code

$$x_0(t_0) = 0.$$

<div align="right">**Eq. 2.4**</div>

Based on the position given by the current head and on current head's offset, the global position x will be calculated:

$$x(t_0) = x_0(t_0) + x_A(t_0).$$

<div align="right">**Eq. 2.5**</div>

The reconstruction state machine then changes state to H'$_{1A}$. In the next cycles, as long as USE$_A$ remains 1 and USE$_B$, 0, the position of head 1 (A) will be used to calculate the global position. When $N_A > \Delta N$, the Sensor to Digital Board will send the position information from both head 1 (A) and 2 (B), signalled by USE$_A$ = 1 and USE$_B$ = 1; this is time t_1 in Figure 2.17. The global position x will still be calculated based on the head 1 position information. Now the local position of head 2, $x_B(t_1)$, can also be calculated, in a similar manner to that of head 1:

$$x_{inc,B}(t_1) = \frac{P}{2\pi} \cdot atan2\left(S_B(t_1), C_B(t_1)\right)$$
$$if\left(S_B(t_1) < 0\right)$$
$$x_{inc,B}(t_1) = x_{inc,B}(t_1) + P$$
$$x_B(t_1) = P \cdot N_B(t_1) + x_{inc,B}(t_1).$$

<div align="right">**Eq. 2.6**</div>

Subsequently the offset of head 2 will be determined and stored in variable x_0:

$$x_0(t_1) = x(t_1) - x_B(t_1).$$

<div align="right">**Eq. 2.7**</div>

The state of the reconstruction state machine changes to H'$_{2B}$ and in the following cycles (between t_1 and t_2) the global position x will be calculated based on the position given by head 2 (B) and on the offset calculated at time t_1:

$$x(t_1 < t \le t_2) = x_0(t_1) + x_B(t_1 < t \le t_2).$$

<div align="right">**Eq. 2.8**</div>

At t_2 ($N_B > \Delta N$) a new transition, between heads 2 and 3 occurs, and the offset of the new head (3) $x_0(t_2)$ will again be calculated, in a similar manner. During state H'$_{3A}$ a reversal of the travelling direction is assumed. The global position continues to be calculated based on the "A" position information as previously described until t_3, when the Sensor to Digital data (USE$_B$ = 1, MC$_B$ = 2) indicates that the vehicle is back again in the region covered by head 2.

It must be noted that the position $x(t_2)$ (transition between heads 2 and 3 in positive direction) and the position at $x(t_3)$ (transition between heads 3 and 2 in negative direction) are not equal. There are two reasons for this: firstly, the phase difference that may exist between the two heads and secondly, since the position data is updated in the Sensor to Digital firmware every 2 µs, but read through the Sensor Bus every 100 µs by the control algorithm, there can be a difference between the position where the transition starts and the position where the transition data is for the first time available to the position calculation function. This has however no influence on the position calculation since the Sensor to Digital firmware guarantees that the signals from the previous and from the incoming head are sampled simultaneously at every transition until they are read through the Sensor Bus and made available to the position calculation function.

While the scale covers head 2 (state H'$_{2B}$) a new reversal (back to positive direction) is assumed and the vehicle travels now the entire region covered by the sensors, exiting at its right. The state of the reconstruction state machine changes, recalculating the offsets of the incoming heads, as indicated by the data received from Sensor to Digital Board and in parallel the global position x is calculated. This continues until time t_8, when the last

position, $x(t_8)$, is calculated. In the next control cycle ($t_8 + 100$ µs), $USE_A = 0$ and $USE_B = 0$ is read from the pre-processing unit, meaning that the scale is beyond head 5B. No position information is now available, and the reconstruction state machine returns to its initial state H'_0.

It is then assumed, in Figure 2.17, that the vehicle re-enters the sensors' region, but now from the right (time t_9). Now the offset of the firstly covered read-head (5B) is no longer zero; the offset of head 5B, calculated during a run in positive direction can be used. For example, $x_0(t_9) = x_0(t_7)$. The reconstruction state machine will change state to H'_{5B} and the information from head 5B ($S_B = S_5$, $C_B = C_5$ and N_B) will be used together with $x_0(t_9)$ to calculate the global position. The procedure is identical with the one discussed for positive direction (entering from left).

2.2.7 Correction of Sine/Cosine Signals

Measurements on the experimental setup have shown that the sine and cosine signals produced by the five read-heads contain amplitude and offset errors that cannot be neglected. These systematic errors of the sine/cosine signals will result in an error of the calculated position and must therefore be minimized. Parametric correction tables [28], [29] were used for the correction of the sine/cosine signals. Even though a reference position sensor is available at the experimental setup, its use for the generation of the correction tables was avoided.

The corrections are generated based solely on the shape of the sensors' sine/cosine signals: in a xy-representation, the sine and cosine signals coming from an ideal sensor will describe a circle, with the centre in the origin of the said xy plane, and all the periods of the ideal sensor will produce the same circle. The radius of this circle is actually not important, as long as the arctangent function is used to calculate the incremental position (the phase inside one given period, see also Eq. 2.2). That is, the amplitudes of sine and cosine can change, as long as their ratio remains unity.

Amplitude errors will change this ideal circle into an ellipse, and offset errors will move its centre away from the origin of the xy plane. The correction tables aim to bring the locus of the output signals from each read-head as close as possible to a circle, thus minimizing the systematic errors. The influence of the corrections on the locus described by the output signals of the sensors will be further described in section 2.2.8 (Experimental Results).

Two types of parametric corrections were tested: mean corrections and period corrections. In the case of mean corrections only three values are necessary for each read-head: the mean values of the offsets of the sine/cosine signals and the mean value of their amplitudes' ratio. The sensor signals can be acquired for the entire length covered by a read-head with a storage oscilloscope, and then the data is transferred in a PC, where the mean values are calculated. This method is simple to implement and requires very few correction values (three per read-head), but it cannot compensate for variations of the systematic errors along the sensor's length.

To further improve the quality of the position signal the more complex period corrections will be used: the amplitudes and offsets of the sine/cosine signals will be determined for each 40 µm period of each read-head and stored in correction tables. These tables will then be used by the control program for online corrections.

To calculate the period correction tables the Sensor to Digital Board will be used for the acquisition of the sensors' signals. A high enough number of samples (e.g. 100) in each period of each read-head is necessary. In order to be able to acquire this high number of data points, modifications were made in the Sensor Bus transfer protocol, as well as in the Sensor to Digital and PCI Interface firmware.

The Sensor to Digital firmware will sample the sine/cosine signals at a higher rate

(1.2 µs) and then send the data directly through the Sensor Bus (in a so-called Burst Mode). In this mode a transfer rate of 20 Mbit/s is achieved on the Sensor Bus. The PCI Interface FPGA will temporarily store the incoming data in a firmware-implemented FIFO buffer from where it will then be read from the control program and transferred on the hard disk of the control PC for subsequent (offline) processing. Because the position is not available during the acquisition of the correction data, no field oriented control can be used; the vehicle must be moved using voltage-frequency control, or by hand. The speed of the vehicle can change during the acquisition of the correction tables data, as long as it remains low enough, so that a sufficiently large number of samples per period is acquired.

The amplitudes and offsets of the sine/cosine signals for each period of each read-head are calculated offline, using the acquired data. The resulting correction values (more than 100,000) will be then stored on the control PC and used by the position calculation function in the control program. Based on the corrected values of sine and cosine a corrected incremental position will be calculated. Because the counters in the Sensor to Digital FPGA firmware use the uncorrected values of sine and cosine, the period numbers (N_A, N_B) read from Sensor to Digital must also be corrected in the control program:

- If the uncorrected sine and cosine determine a point in the 4th quadrant, and the corrected ones, a point in the 1st quadrant, a unit is added to the period counter;
- If the uncorrected sine and cosine determine a point in the 1st quadrant, and the corrected ones, a point in the 4th, a unit is subtracted from the period counter;
- Otherwise, the period counter remains unchanged.

After the correction of sines, cosines and period counters, the corrected position will be calculated in the same manner as the uncorrected one.
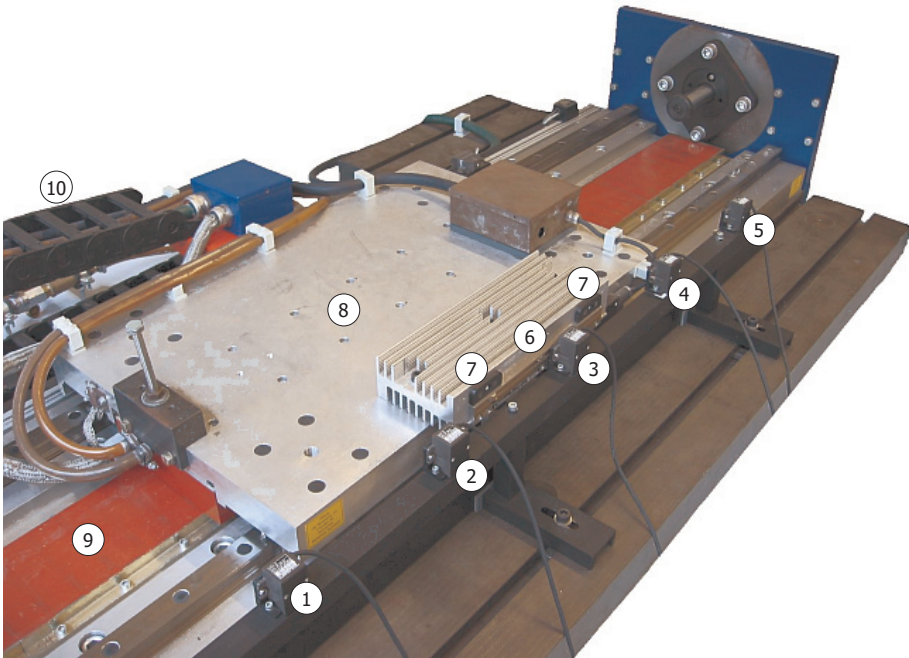

Figure 2.18: Experimental setup used to test the position acquisition

2.2.8 Experimental Results

The optical sensors system described in the previous sections was tested at the experimental setup shown in Figure 2.18. The optical read-heads (1) … (5) were mounted along the track (9) of a linear permanent magnet synchronous machine with active mover. The scale (6), along with the reference selection magnets (7) was mounted on the vehicle (8). The test machine was also already equipped with another optical sensor (not visible in Figure 2.18), mounted in the configuration used for linear machines with active vehicle (scale on the track, one read-head on the vehicle; see also Figure 2.1). The scale of this sensor has a 4 µm grating period (one tenth of the period of the tested sensors system), therefore it was possible to use it as reference sensor.

Figure 2.19 (a) shows the sine and cosine signals of read-head 1 of the tested system, for all the $\Delta N = 5250$ periods covered by the sensor. Ideally, all the points should be located on the circle also depicted in the figure; in reality, due to the amplitude and offset errors, they form an elliptical annulus, which is not centred in the origin of the xy (sine/cosine) plane. The thickness of the annulus is an indication of the variation of the systematic errors along the sensor.

Using mean corrections, as described in section 2.2.7, the representation of the corrected sine/cosine signals from Figure 2.19 (b) was obtained. The thickness of the annulus remains unchanged (mean corrections have no influence on the variation of the systematic errors), but the annulus is now centred in the origin of the xy plane. There is also an improvement of the mean amplitudes of the sine and cosine signals.

Using period corrections (Figure 2.19 (c)), the shape of the locus of the corrected sine/cosine signals comes very close to that of the unity circle, which indicates the reduction of the position error caused by the systematic errors. There still are some slightly "thick" parts in the diagram, oriented after the directions of the two bisectors of the xy plane. They are an indication that the sensor also has small phase errors, which errors were not compensated for by the correction tables.
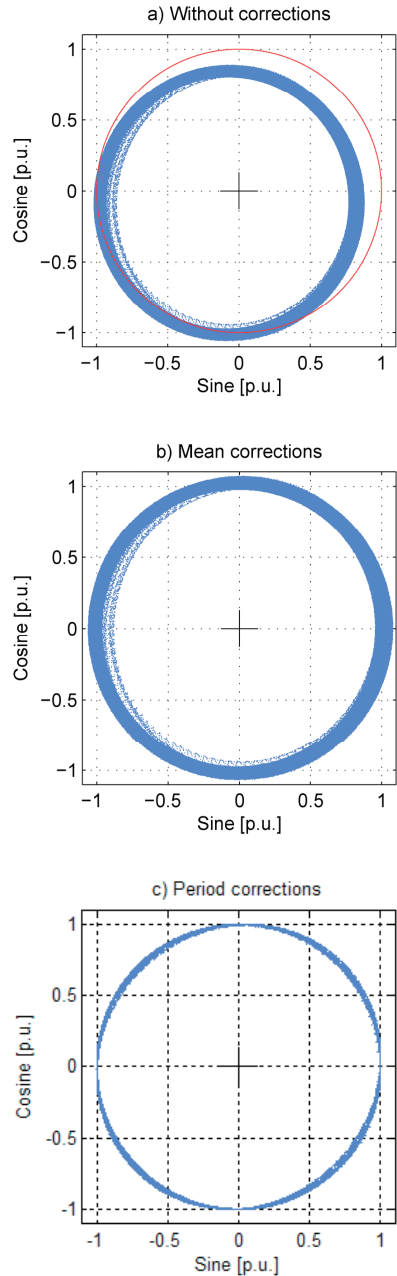

Figure 2.19 XY-Representation of the sine and cosine signals of head 1 (whole length)

The position deviation between the tested optical system and the optical sensor used as reference is shown in Figure 2.20 (a). The position of the tested system was acquired every 100 µs (as described in the previous sections) together with the one of the reference sensor, during a test run. The numbers 1 … 5 in the figure represent the region covered by each read-head.

In Figure 2.20 (a) it looks, as there is a jump in the position deviation calculated without corrections (the blue waveform), at the transition between read-heads 3 and 4, but this is not the case, as Figure 2.20 (b) shows. Figures (b) and (c) are zooms of Figure 2.20 (a), at the transitions between heads 2 – 3 and 3 – 4, respectively. In these zooms it can be seen that the position deviation is continuous, meaning that no jump occurs at the transition between read-heads in the position calculated based on the Sensor to Digital data.

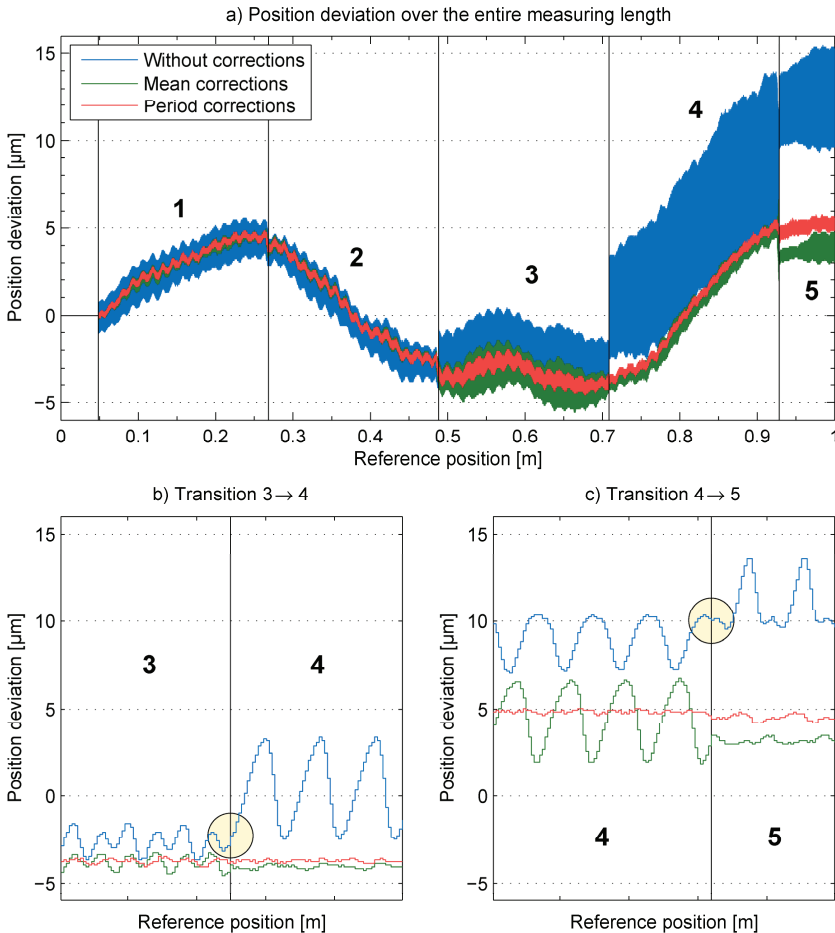Two components of the position deviation can be identified in Figure 2.20: the first
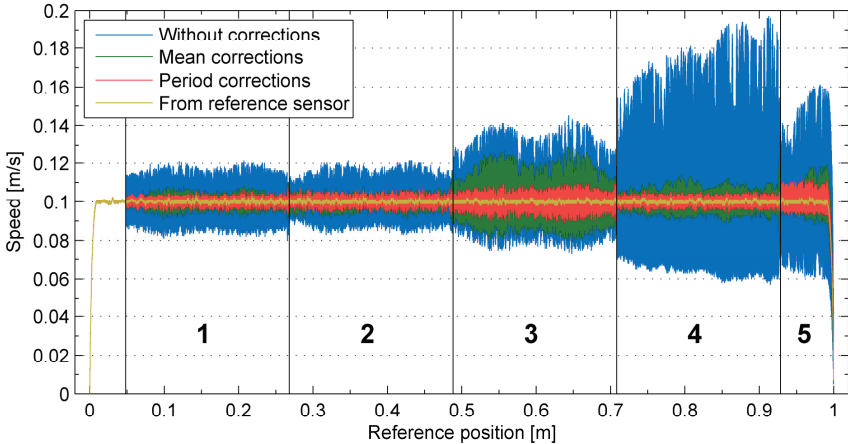

Figure 2.20: Position deviation

Figure 2.21: Measured speed without and with position correction

one has the same wavelength as the pitch of the sensors (40 μm); it is due to systematic errors and can be reduced by correcting the sine/cosine signals. The other one has a wavelength comparable with the entire measuring range and is not influenced by corrections. The scale of the tested system does not cause this low-frequent component, because in that case a similar variation should be observed for each of the five heads.

In a test, the first head was mounted in the place where the second one normally is. The same position deviation as the one of head 2 in Figure 2.20 was observed, so the low-frequent component of the position variation also does not depend on the read-heads themselves or on the signal processing in the position acquisition algorithm. It can only be attributed to the sensor used as reference, which is possible to present large variations of the accuracy in different regions, due to its mechanical mounting and many-years utilisation.

By using period corrections, the position deviation along the entire measuring range was reduced from ca. –5 … +15 μm to approx. ±5 μm. The component of position deviation due to systematic errors amounts to ca. 1 μm, peak-to-peak.

The reduction of the position error is important not only for accurate positioning, but also for the calculation of the speed [30]. Figure 2.21 shows the speed signals, as obtained by numerically deriving (in the 100 μs control interval) the uncorrected, mean corrected and period corrected positions of the test system, compared with the speed obtained from the reference sensor. No filtering was used for the speed signals, in any of the cases. A major reduction of the variation due to derivation was obtained by using the periodic correction tables.

2.2.9   Implementation of the Optical System

The optical system described in this section was implemented on the linear drive for material handling described in section 3 of this work. The details of the implementation of the optical sensors system are given in sub-section 3.2.

2.3 Capacitive Sensor

The optical sensors system described in the previous section achieves a high accuracy (especially after the correction of the systematic errors), but, due to the large number of read-heads, is expensive. It also requires complex electronics and complex processing algorithms for the synchronisation between read-heads. Therefore, simpler and cost effective alternatives should be investigated.

One of the alternatives can be the position sensor described in this section. The sensor, shown in Figure 2.22, works based on the capacitive principle: an electric field is generated by stationary electrodes on the fixed part of the sensor; the mobile part (attached to the mover of the linear machine) slides between the stationary electrodes, causing a position-dependent modulation of the electrical field.

The exact arrangement of electrodes and an electrical model of the sensor will be presented in subsection 2.3.2. The fully passive slider makes the capacitive sensor well suited for use in linear drives with passive vehicles. The sensor is commercially available from the Netzer / Sick-Stegmann company [31]. It was first tested as provided by manufacturer (with the original electronics and processing algorithm [32]). The measurements during the initial tests (see subsection 2.3.1) confirmed what was to be expected from datasheet: the original electronics generates the output signal with a time delay, which delay causes an increasing position error as the travelling speed increases. This makes the sensor unsuitable for position control of high dynamic linear drives.

Therefore, we developed the hardware and software for two new signal processing methods, which use from the original system only the arrangement of electrodes. The first method uses the same square-wave excitation scheme as the commercial system and the second method uses sinusoidal excitation. The new methods, presented in subsections 2.3.3 and 2.3.4, respectively, avoid the signal delay and velocity-dependent errors.

Figure 2.23 shows the linear drive used for the evaluation of the capacitive sensor (with the original, as well as with the improved processing algorithms): The linear motor (1) with active mover (2) is supplied via a drag chain (3) from the converter (4). The converter has integrated current control, so only the speed and position control loops will be implemented in the control software. An optical sensor (LIF 171R from Heidenhain company) is used for position feedback in the control loop, as well as for position reference. The reference sensor has a pitch of 8 µm and provides at its output TTL quadrature signals, with a resolution of 0.8 µm. The scale of the reference sensor (6) is mounted along the track of the liner machine, whilst the active read-head is fixed on the mobile arm (5), attached to the machine's mover. On the same mobile arm, the slider of
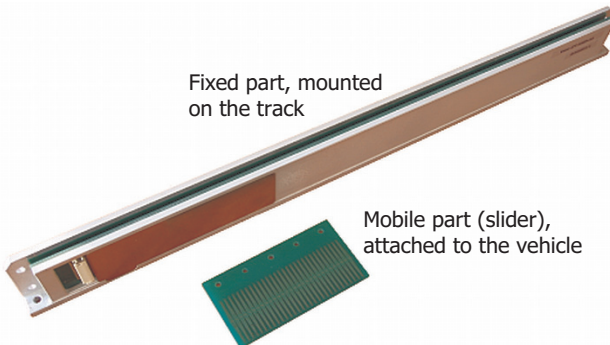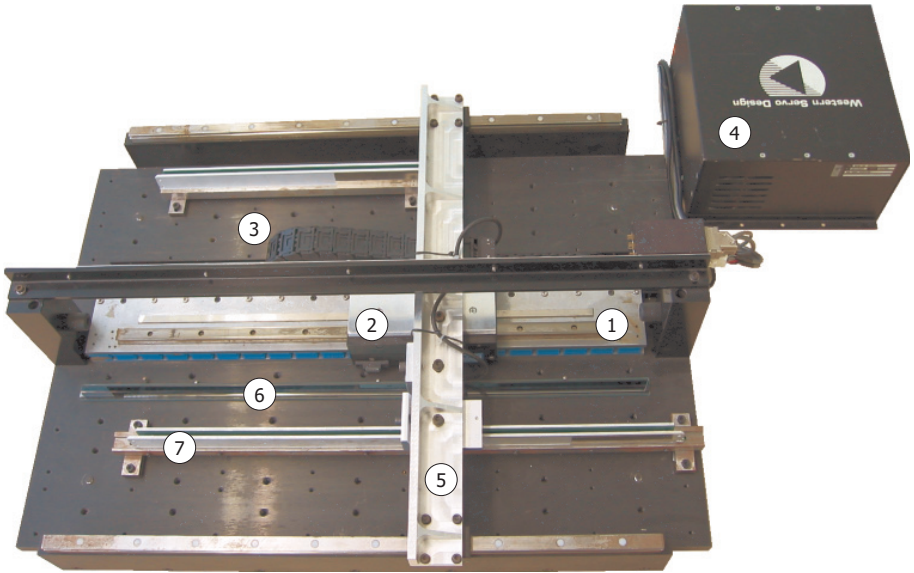

Figure 2.22: Capacitive sensor (LE^{2}C-050025-I-A5) from Netzer / Sick-Stegmann

Figure 2.23: Linear drive for testing the capacitive sensor

the evaluated capacitive sensor (7) is also mounted.

2.3.1 Initial Tests

An overview of the setup used in the initial evaluation [33] is given in Figure 2.24. For controlling the linear motor and acquiring the capacitive sensor's position a DS1102 Controller Board, equipped with a TMS320C31 Digital Signal Processor (DSP) was used. The original electronics of the capacitive sensor output incremental quadrature signals (RS-422 standard), with a resolution of 4096 counts per pitch (2 mm). The current reference for the converter is given through one of the Controller Board's digital-to-analog converters. The control code was written using Matlab Simulink and complied with the Real-Time Workshop (RTW) tool. The compiled code is downloaded through the Host Interface to the control DSP, and the ControlDesk software is used for simultaneously acquiring the two sensors' position.

The results (position differences between the tested and the reference) for three
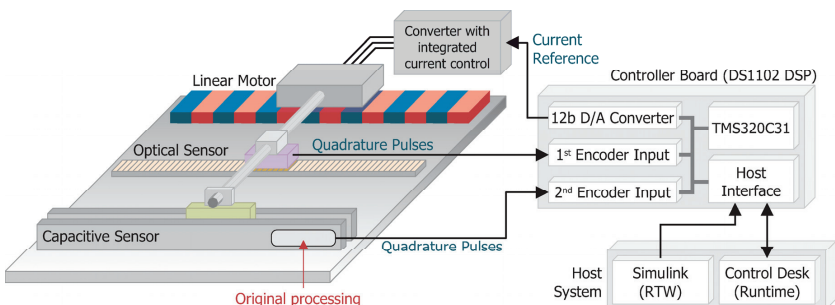
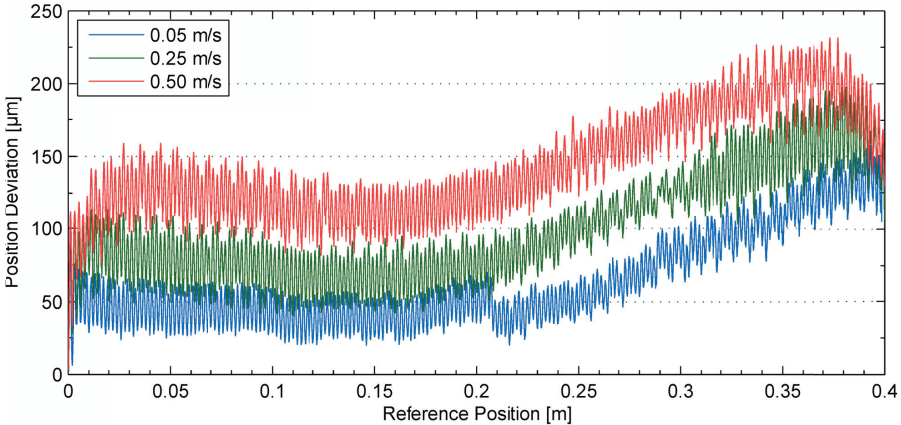**Figure 2.24: Overview of the setup used in initial tests**

Figure 2.25: Test results using the original signal processing

test-runs, at relatively slow speeds are shown in Figure 2.25. The position error due to the processing delays amounts to more than 120 µm at a speed of 0.5 m/s. For high speeds (e.g. 10 m/s) this error would be unacceptable for high-dynamic position control.

2.3.2 Electrical Model of the Capacitive Sensor

In order to develop an improved processing algorithm, a model of the capacitive sensor must first be derived. The sensing arrangement itself consists of three printed circuit boards (PCBs), as shown in Figure 2.26:

1) A stationary <u>Transmitting Plate</u>, with transmitting electrodes in the shape of rectangular copper strips. Four adjacent copper strips (named "a", "b", "c" and "d") cover one pitch P of the sensor (2mm). Each fourth strip is fed by one of the voltages $U_a – U_d$. All the electrodes j (j = a, b, c, d) from all the periods of the sensor are connected in parallel. The transmitting electrodes and their connection are shown in Figure 2.29.

2) A mobile <u>Modulating Plate</u> (slider) covered on one side with a sinusoidal copper area, which is the modulating electrode. On the other side is a copper plane, the coupling electrode, which is electrically connected with the modulating electrode. The modulating plate contains 30 periods (60mm), covering slightly more than 10% of the total length of the tested sensor. This plate encodes the position dependent information in the electric field generated by the excitation voltages $U_a – U_d$.

3) A second stationary plate, the <u>Receiving Plate</u>, on the other side of the slider, with one receiving electrode (also an unstructured copper plane); the charge received on this electrode contains the position information.

The real electric field distribution in the space between transmitting and receiving electrode may be calculated using 3D Finite Element Method (FEM). With a large number of 3D FEM calculations, where the modulating electrode (slider) is being moved by a small distance between successive FEM-calculations, the output signal U_{out} of the charge amplifier may be calculated. Such a procedure will be helpful to optimise the shape of the modulating electrode such that the position of the slider can be extracted precisely from U_{out} [34]. But in this work, the commercially available arrangement of electrodes is used and we concentrate on extracting the position information from U_{out}. For that purpose, a simplified analytical model of the capacitive sensor was developed, based on the assumption of homogeneous field distribution, which permits the approximation of the electric field through concentrated capacitances, as shown in Figure 2.30.
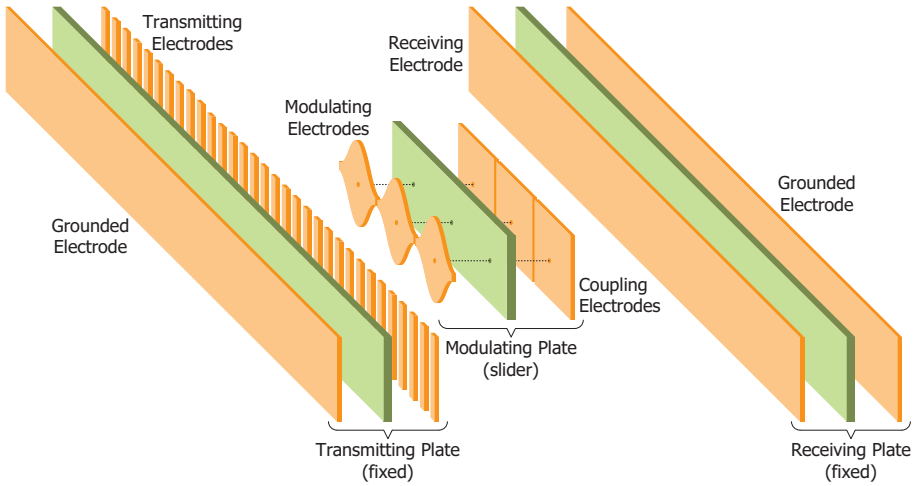
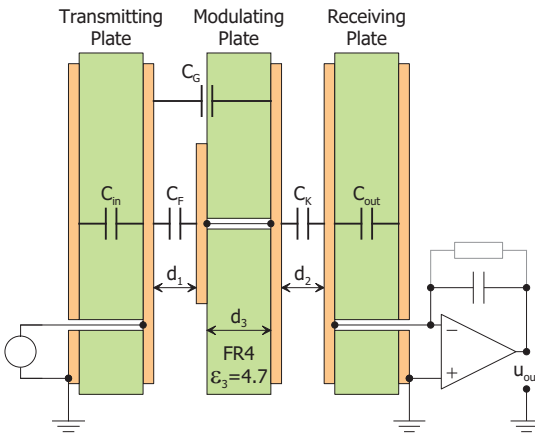**Figure 2.26: Exploded view of the capacitive sensor**
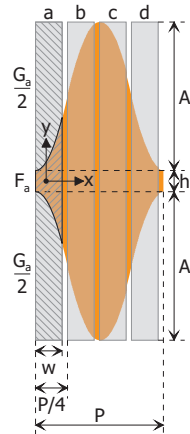

Figure 2.27: Cross-section of the sensor

Figure 2.28: Position-dependent overlaping areas
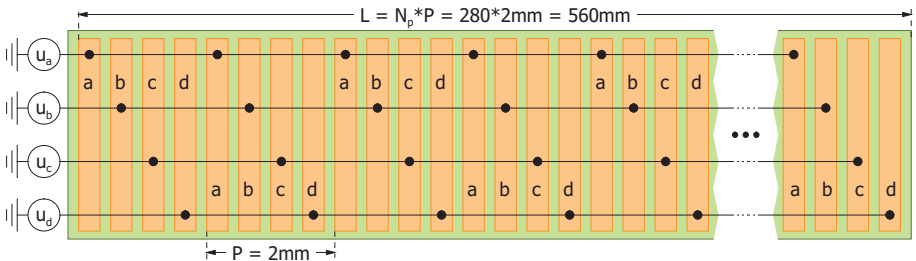

Figure 2.29: Transmitting electrodes

Besides the position dependent capacitances, there are several position independent ones, some of which can be neglected in the analysis. As the modulating plate is much shorter than the other two plates, there are constant capacitances ($C_{a,0}$, $C_{b,0}$, $C_{c,0}$, $C_{d,0}$) from the transmitting electrodes to the receiving electrode. These four capacitances are approximately equal and their common value will be referred to as C_0:

$$C_{a,0} = C_{b,0} = C_{c,0} = C_{d,0} = C_0 \, ,$$

Eq. 2.9

$$C_0 = \frac{(N_p - N_{mp}) \cdot \varepsilon_0 \cdot w \, (2A + h)}{d_1 + d_2 + d_3} \, .$$

Eq. 2.10

The position independent capacitance between coupling and receiving electrode is:

$$C_k = \frac{N_{mp} \cdot \varepsilon_0 \cdot P(2A + h)}{d_2}$$

Eq. 2.11

N_{mp} is the number of periods at the modulating electrode (30 periods); ε_0 is the electrical permittivity of air and ε_3 is the relative permittivity of FR4 [35]. A, h, P, w, d_1, d_2 and d_3 are dimensions of the capacitive sensor as shown in Figure 2.26 – Figure 2.29.

The values of the geometrical dimensions of the sensor, as well as the values of the resulting capacitances (calculated under the assumption of an homogenous electric field) are given in Appendix 5.2.

Capacitances $C_{j,in}$ (j = a, b, c, d) and C_{out} (see Figure 2.27 and Figure 2.30) can be neglected in the analysis. The first ones are in parallel with the voltage sources U_a – U_d (between the transmitting electrodes and ground), whilst C_{out} is short-circuited by the charge amplifier (see below).

Information on the position of the slider is provided by position dependent capacitances. Figure 2.28 shows the areas, where transmitting electrodes a – d face the modulating electrode. The area F_a depends on the position "x" by:

$$F_a(x) = h \cdot w + 2 \cdot \frac{A}{2} \int_{x - \frac{w}{2}}^{x + \frac{w}{2}} \left[1 + \sin\left(\frac{2\pi}{P} \chi - \frac{\pi}{2} \right) \right] d\chi \, .$$

Eq. 2.12

Solving the above integral, we get:

$$F_a(x) = \underbrace{w \cdot (h + A)}_{\text{Constant}} - \underbrace{\frac{P \cdot A}{\pi} \cdot \sin\left(\frac{w\pi}{P} \right)}_{\text{Constant}} \cdot \cos\left(\frac{2\pi}{P} x \right) \, .$$

Eq. 2.13

The area G_a is complementary to F_a:

$$G_a(x) = w \cdot (h + 2 \cdot A) - F_a(x) \, ,$$

Eq. 2.14

and together they define the position dependent capacitance:

$$C_a(x) = \frac{N_{mp} \cdot \varepsilon_0 \cdot F_a(x)}{d_1} + \frac{N_{mp} \cdot \varepsilon_0 \cdot G_a(x)}{d_1 + \frac{d_3}{\varepsilon_3}} \, ,$$

Eq. 2.15

which can also be written:

$$C_a(x) = C_{avg} - C_{var} \cdot \cos\left(\frac{2\pi}{P} x \right) \, .$$

Eq. 2.16

The constant values C_{avg} and C_{var} are:

$$C_{avg} = N_{mp} \cdot \varepsilon_0 \cdot w \cdot \left(\frac{h+A}{d_1} + \frac{A}{d_1 + d_3/\varepsilon_3} \right),$$

<div align="right">Eq. 2.17</div>

$$C_{var} = N_{mp} \cdot \varepsilon_0 \cdot \frac{PA}{\pi} \cdot \left(\frac{1}{d_1} - \frac{1}{d_1 + d_3/\varepsilon_3} \right) \cdot \sin\left(\frac{w\pi}{P} \right).$$

<div align="right">Eq. 2.18</div>

In a similar way, the position dependent capacitances between the other three electrode structures (b, c and d) and the modulating plate can be written. All the position dependent capacitances are given below:

$$\begin{cases} C_a(x) = C_{avg} - C_{var} \cdot \cos\left(\frac{2\pi}{P} x \right) \\ C_b(x) = C_{avg} - C_{var} \cdot \cos\left(\frac{2\pi}{P} x + \frac{\pi}{2} \right) \\ C_c(x) = C_{avg} - C_{var} \cdot \cos\left(\frac{2\pi}{P} x + \pi \right) \\ C_d(x) = C_{avg} - C_{var} \cdot \cos\left(\frac{2\pi}{P} x + \frac{3\pi}{2} \right). \end{cases}$$

<div align="right">Eq. 2.19</div>

These four capacitances consist of a constant part, C_{avg}, on which sinusoidal position dependent variations, of amplitude C_{var} and 90° phase shifted are superimposed.

Based on the above-determined capacitances, the equivalent circuit from Figure 2.30 is obtained for the capacitive sensor. In the following it will be assumed that at the output (receiving) electrode of the sensor a charge amplifier is connected. Taking into
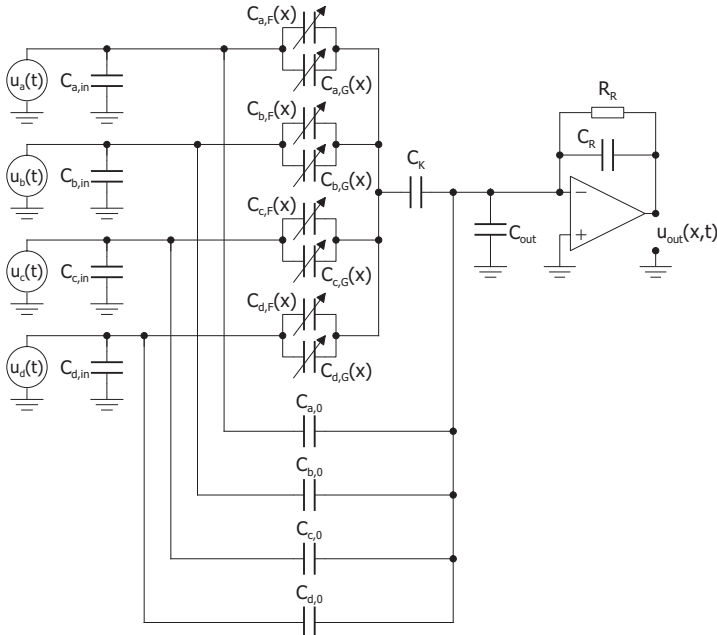

Figure 2.30: Equivalent circuit of the capacitive sensor

account the equivalent circuit, the output of the charge amplifier can be written:

$$U_{out}(x) = \frac{-sR_R}{1+sR_RC_R}\left\{\frac{C_a(x)U_a + C_b(x)U_b + C_c(x)U_c + C_d(x)U_d}{1 + \dfrac{C_a(x) + C_b(x) + C_c(x) + C_d(x)}{C_k}} + C_0U_0\right\},$$ **Eq. 2.20**

where U_0 is the sum of the four excitation voltages:

$$U_0 = U_a + U_b + U_c + U_d,$$ **Eq. 2.21**

and „s" is the Laplace operator. Introducing the line-to-line voltages:

$$\begin{cases} U_{ac} = U_a - U_c \\ U_{db} = U_d - U_b, \end{cases}$$ **Eq. 2.22**

the following expression of the output voltage is obtained:

$$U_{out}(x) = \frac{-sR_R}{1+sR_RC_R}\cdot\left\{\overbrace{\frac{C_kC_{var}}{C_k + 4C_{avg}}\left[U_{ac}\cos\left(\frac{2\pi}{P}x\right) + U_{db}\sin\left(\frac{2\pi}{P}x\right)\right]}^{(1)} + \right.$$

$$\left. + U_0\underbrace{\left[C_0 + \frac{C_kC_{avg}}{C_k + 4C_{avg}}\right]}_{(2)}\right\}.$$ **Eq. 2.23**

The term denoted with (1) contains the position information, and the term denoted with (2) stands for a component which decays with $\tau = R_RC_R$ (some milliseconds) due to the high-pass filter behaviour of the charge amplifier. This component can be completely eliminated if $U_0(t) = 0$. By using the notation:

$$K = -\frac{1}{C_R}\cdot\frac{C_kC_{var}}{C_k + 4C_{avg}},$$ **Eq. 2.24**

and assuming $U_0(t) = 0$ and $R_R \to \infty$, the output of the charge amplifier can be written:

$$U_{out}(x) = K\cdot\left[U_{ac}\cos\left(\frac{2\pi}{P}x\right) + U_{db}\sin\left(\frac{2\pi}{P}x\right)\right].$$ **Eq. 2.25**

2.3.3 Square-wave excitation

In this section, the operating mode of the capacitive sensor with square-wave excitation, like the one used in the commercially available system, will be investigated. For high dynamic drives, the delay introduced into the control loop by position acquisition must be as short as possible. Therefore, the PLL-like tracking filter [36] used in the commercial system will be avoided, and a new instantaneous demodulation method will be proposed.

Figure 2.31 shows the four rectangular excitation voltages $U_a - U_d$ and the two line-to-line voltages. They have a period of 15 µs and 50% duty cycle. The phase shift between two successive voltages is 90°. In order to extract the position from U_{out} (Eq. 2.25), instantaneous demodulation will be used: in each phase Ph1 … Ph4 the output voltage is sampled by an analog-to-digital converter. The four resulting samples are:

$$\begin{cases} U_{out,1}(x) = K \cdot U_{DC} \cdot \left[\cos\left(\frac{2\pi}{P}x\right) + \sin\left(\frac{2\pi}{P}x\right) \right] \\ U_{out,2}(x) = K \cdot U_{DC} \cdot \left[\cos\left(\frac{2\pi}{P}x\right) - \sin\left(\frac{2\pi}{P}x\right) \right] \\ U_{out,3}(x) = K \cdot U_{DC} \cdot \left[-\cos\left(\frac{2\pi}{P}x\right) - \sin\left(\frac{2\pi}{P}x\right) \right] \\ U_{out,4}(x) = K \cdot U_{DC} \cdot \left[-\cos\left(\frac{2\pi}{P}x\right) + \sin\left(\frac{2\pi}{P}x\right) \right] \end{cases}.$$
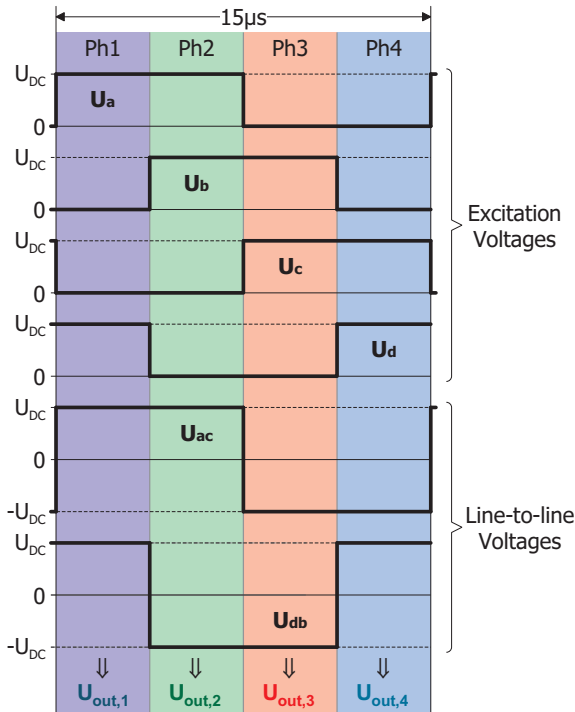
Eq. 2.26

Figure 2.31: Ideal square-wave excitation voltages

The sine and cosine sums in the above equations can be rearranged, yielding:

$$\begin{cases} U_{out,1}(x) = \overbrace{K \cdot U_{DC} \cdot \sqrt{2}}^{Constant} \cdot \sin\left(\frac{2\pi}{P}x + \frac{\pi}{4}\right) \\ U_{out,2}(x) = K \cdot U_{DC} \cdot \sqrt{2} \cdot \cos\left(\frac{2\pi}{P}x + \frac{\pi}{4}\right) \\ U_{out,3}(x) = -K \cdot U_{DC} \cdot \sqrt{2} \cdot \sin\left(\frac{2\pi}{P}x + \frac{\pi}{4}\right) \\ U_{out,4}(x) = -K \cdot U_{DC} \cdot \sqrt{2} \cdot \cos\left(\frac{2\pi}{P}x + \frac{\pi}{4}\right). \end{cases}$$

Eq. 2.27

From the sampled output voltages of two consecutive phases, position values can be calculated using the four-quadrant inverse tangent function, atan2:

$$\begin{cases} x_{1,2} = \frac{P}{2\pi} \cdot atan2\left(U_{out,1}, U_{out,2}\right) - \frac{\pi}{4} \\ x_{2,3} = \frac{P}{2\pi} \cdot atan2\left(-U_{out,3}, U_{out,2}\right) - \frac{\pi}{4} \\ x_{3,4} = \frac{P}{2\pi} \cdot atan2\left(U_{out,3}, U_{out,4}\right) - \frac{\pi}{4}. \end{cases}$$

Eq. 2.28

A new position value might be calculated each 3,75 µs (one fourth of the excitation period), if the data transfer and the controller are fast enough. In a practical application an average value may be calculated using values from the past in a user-defined time window, thus providing over sampling and filtering as desired.

Figure 2.32 shows the experimental setup used for the evaluation of the capacitive sensor, using rectangular excitation. Two identical ISA boards ("Reference Position Acquisition Board" and "Capacitive Sensor Acquisition Board"), located in the control PC, are used for the evaluation of the signals from the two sensors. Each board is equipped with a Complex Programmable Logic Device (CPLD), as well as with analog-to-digital and
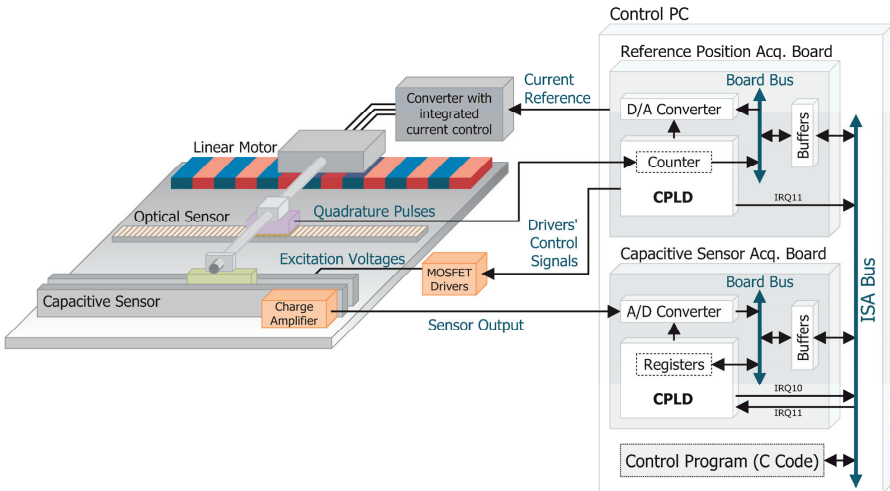

Figure 2.32: Overview of the setup used for evaluation with rectangular excitation

digital-to-analog converters. For simplicity, the components not used in the evaluation are not shown in Figure 2.32. The four excitation voltages for the capacitive sensor are generated by two dual MOSFET drivers (one driver generates two complementary voltages). The two control signals for the drivers are generated by the Reference Position Acquisition Board's CPLD, and they are synchronised with the acquisition of the reference position (from the optical sensor).

The charge amplifier is located on a small PCB, which is mounted directly on the receiving plate of the capacitive sensor, in order to keep parasitics and susceptibility to noise as small as possible. For the charge amplifier, a low-noise, low input current Bi-Mosfet operational amplifier [37] was chosen [38].

The output of the charge amplifier is transferred differentially to the Capacitive Sensor Acquisition Board, also to reduce the influence of the external disturbances. Here, the signal is sampled by the board's analog-to-digital converter, four times in each cycle of the excitation voltages (once in each phase Ph1 … Ph4, see Figure 2.31).

Synchronisation between the generation of the excitation voltages and the conversion start signal of the analog-to-digital converter occurs via the "IRQ11" signal on the ISA-Bus. This signal is asserted by the Reference Position Acquisition Board's CPLD every 100 µs. When this signal is active, the Capacitive Sensor Acquisition Board's CPLD firmware saves the four sampled voltages in the CPLD registers and asserts the "IRQ10" signal. This interrupt request triggers the execution of the Interrupt Service Routine in the control program. Here the position information from the two sensors will be read, through the ISA-Bus. The position information of the capacitive sensor consists of the four samples of the charge amplifier's output, taken in one cycle of the excitation voltages. Based on them, the position of the capacitive sensor can be calculated using Eq. 2.28.
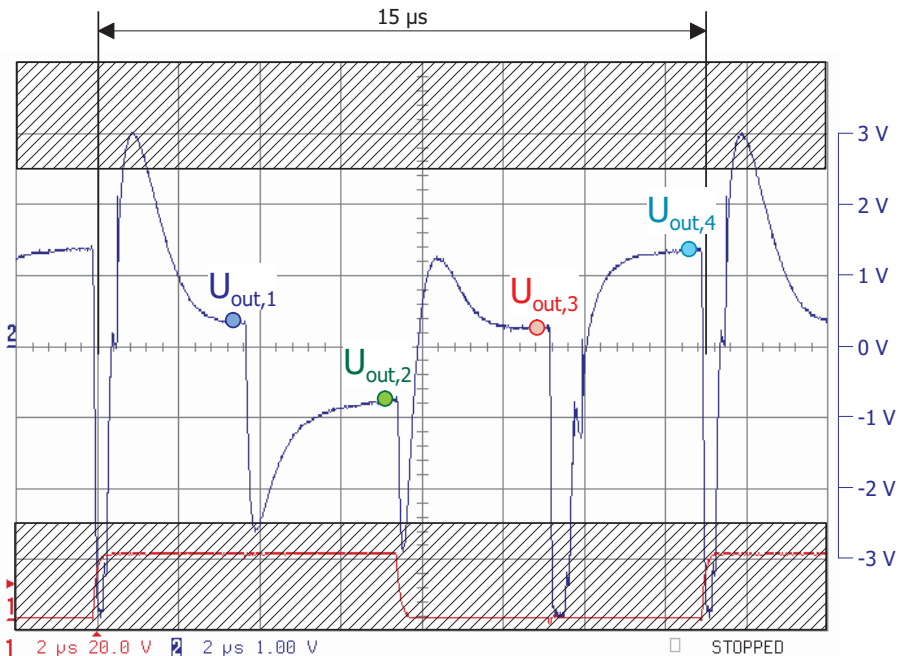
Figure 2.33: Capacitive sensor's output voltage (measured at the input of the A/D converter)

The position of the reference sensor will also be used as feedback for the position and speed controllers, the output of which (the linear motor's reference current) is written back on the ISA-Bus to the Reference Position Acquisition Board's CPLD, and from there it is loaded in the digital-to-analog converter of the board.

In Figure 2.31 ideal excitation voltages are depicted. With a real excitation e.g. during the rising edge of U_a and the falling edge of U_c, the sum of the two voltages may not remain constant, having a variation for some nanoseconds. This variation causes a spike at the input of the charge amplifier. An ideal charge amplifier would reflect this spike at its output according the last term of Eq. 2.23 – note the differentiating "s" in the nominator. After the spike, a real charge amplifier needs some time to settle – as measured in Figure 2.33 – therefore the sampling is done at the end of each 90° interval.

Figure 2.34 shows a large number of samples taken as in Figure 2.33, which generate the four sinusoidal position dependent signals $U_{out,1}$ … $U_{out,4}$ (note the time scale difference in the two figures). The offset and amplitude errors of the four signals are very large, so their correction is necessary. For the correction of the four position dependent
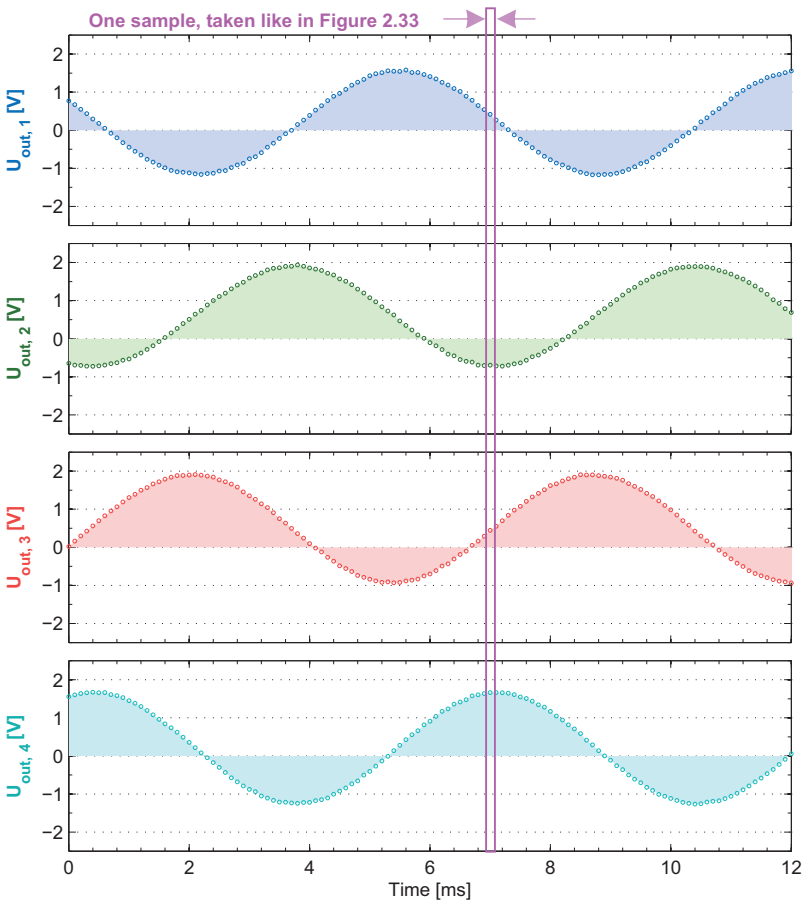
Figure 2.34: Sampled voltages

signals, a parametric table will be used to correct the amplitude and offset of each period of the four sinusoidal position dependent signals (similar to the period corrections described in section 2.2.7).

The results of the amplitude and offset correction are shown in Figure 2.35. Here the position dependent signals $U_{out,1} \ldots U_{out,4}$ are normalised to the input range of the analog-to-digital converter. After applying the period corrections, the shape of the locus of the corrected signals comes close to that of the unity circle, indicating the reduction of the position error caused by the amplitude and offset errors.

In Figure 2.36, the deviation between reference sensor and capacitive sensor is shown. The blue line shows the result when no correction is used, i.e. the signals from Figure 2.34 are used directly for atan2 calculation (Eq. 2.28). In the zoom it is clearly seen, that the deviation (ca. ±50 µm) has the wavelength of the pitch of the capacitive sensor (2 mm), due to the offset and amplitude errors.

In addition to this high-frequent deviation, there is a long-term integral deviation, (ca. 120 µm over the entire measuring length), which cannot be corrected without using
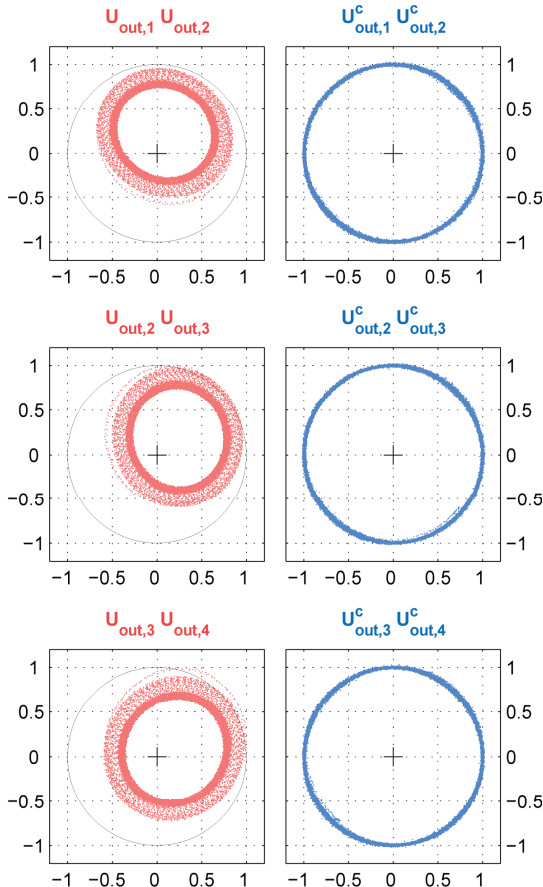
Figure 2.35: XY-Representation of the sampled voltages before and after period corrections

Position deviation over the entire measuring length
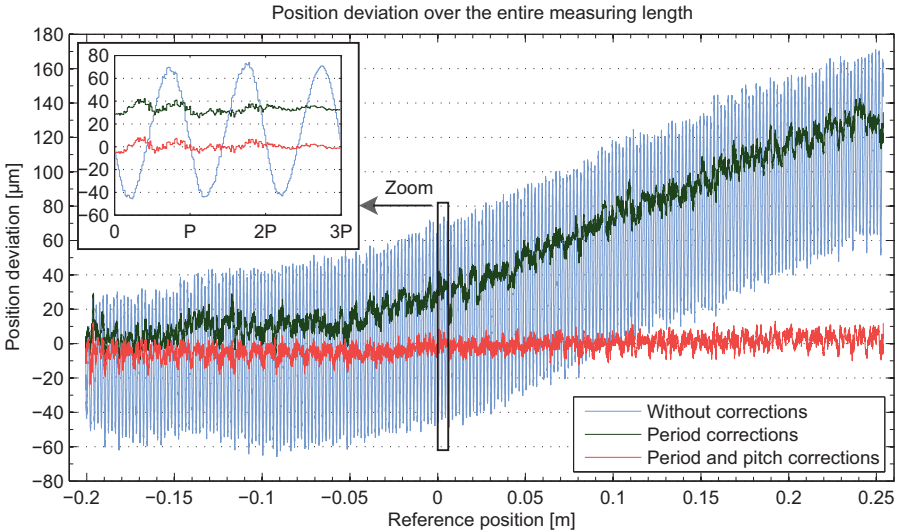
Figure 2.36: Position deviation (using square-wave excitation)

the reference sensor when generating the correction table.

The result of applying a period correction table (generated without the use of the reference sensor) is shown by the dark line. The local deviations are reduced to ±15 µm. This remaining local deviation may be due to harmonics in the signals of Figure 2.34, which are not compensated by the parametric correction table.

The integral deviation remains unchanged – it does not depend on the quality of the sampled signals. This deviation along the whole measuring length is likely to be due to imperfections in producing the pattern of transmitting electrodes and temperature expansion of FR4. One way to improve this would be a very precise manufacturing of the transmission electrodes on a substrate of temperature stable ceramics. Another way might be the usage of a reference sensor when the correction table is generated. This can compensate global imperfections during production but not temperature expansion of FR4.

Figure 2.36 shows in red line the results of applying a correction table, which was generated using the reference sensor. The integral error is compensated now. Such a correction table may be generated at the manufacturer's test bed during the final production test, before the sensor leaves the manufacturer.

2.3.4 Sinusoidal excitation

From Figure 2.33 it is obvious, that the transients generated by switching of the exciting voltages are a major source of problems. It is possible that transients may not have decayed sufficiently at the sampling instant, when analog-to-digital conversion starts. Moreover, for longer sensors (some meters), the capacitances to ground increase and large current pulses will be necessary to charge all the capacitances, producing electromagnetic interference (EMI). To avoid these problems, we propose sinusoidal excitation voltages:

$$\begin{cases} U_{ac}(t) = A_h \cos(\omega_h t) \\ U_{db}(t) = A_h \sin(\omega_h t), \end{cases}$$

<div align="right">**Eq. 2.29**</div>

with

$$\omega_h = 2\pi f_h.$$

<div align="right">**Eq. 2.30**</div>

The excitation frequency f_h is chosen e.g. in the 20 … 100 kHz range. By substituting Eq. 2.29 into Eq. 2.25, the following expression is obtained for the charge amplifier's output:

$$U_{out}(x) = A_h K \cdot \left[\cos(\omega_h t) \cos\left(\frac{2\pi}{P} x \right) + \sin(\omega_h t) \sin\left(\frac{2\pi}{P} x \right) \right].$$

<div align="right">**Eq. 2.31**</div>

Introducing the normalized position:

$$x_P = 2\pi/P \cdot x,$$

<div align="right">**Eq. 2.32**</div>

and reducing the trigonometric expression, Eq. 2.31 can be rewritten as:

$$U_{out}(x) = A_h K \cdot \cos(\omega_h t - x_P).$$

<div align="right">**Eq. 2.33**</div>

The normalized position x_P is the phase difference between the excitation input U_{ac} and the output of an ideal charge amplifier, U_{out}.

The simulation in Figure 2.37 illustrates the principle of phase modulation, as described by Eq. 2.33. The simulation was made for a very high traversing speed (20 m/s) only for better clarity. When the incremental position of the sensor is zero, the signals U_{ac} and U_{out} are in phase.

With increasing position, the phase between these two signals increases proportionally; for x = 1 mm (P/2), U_{ac} and U_{out} are in phase opposition. When x reaches P, the phase difference reaches 360°, and the two signals are again in phase.

By measuring the phase differences between U_{ac} and U_{out} – e.g. by measuring the time interval between the positive zero crossings of the two signals – the position of the capacitive sensor can be determined.

Figure 2.38 shows an overview of the setup used for the evaluation of the capacitive sensor with sinusoidal excitations. The same "Reference Position Acquisition Board" as for the evaluation with square-wave excitations will be used for the acquisition of the optical sensor's position and for the reference current's output.

For the acquisition of the capacitive sensor's position, an eZDSPF2812 development board, equipped with a TMS320F2812 DSP [39] from Texas Instruments will be used. The PWM outputs of the DSP are used to generate the control signals for the sinusoidal voltage generation (as discussed below). Comparators are now used to determine the zero-crossings of the phase-modulated signal U_{out}. The outputs of the comparators are fed to the Capture Units of the DSP.
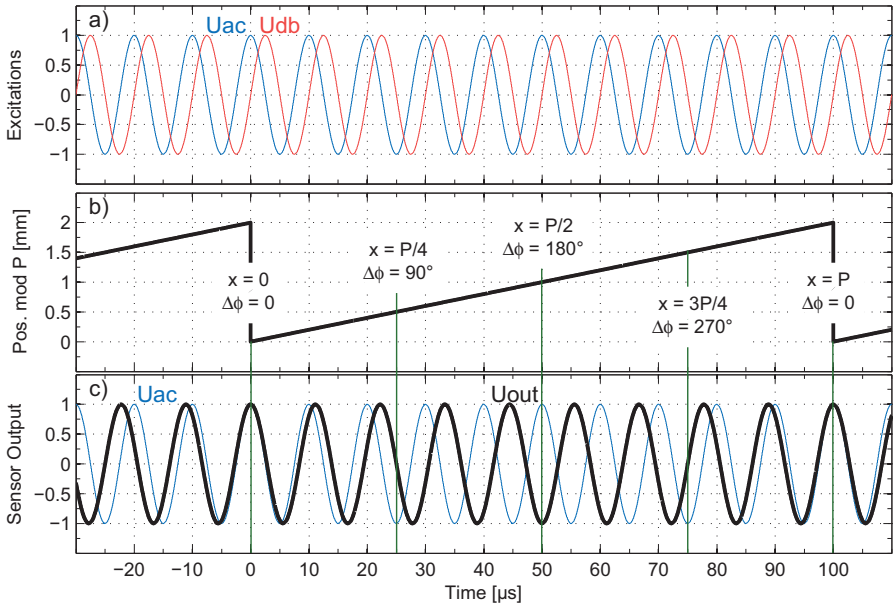
Figure 2.37: Phase modulation

There are two synchronisation signals between the DSP and the Reference Position Acquisition Board's CPLD. The first one, IRQ-SYNC is used to synchronise the 100 µs control interrupt (IRQ11) and the acquisition of the reference position with the generation of the excitation signals of the capacitive sensor. The second signal, DATA-SAVE, is used to synchronise the saving of the position from the two sensors, ensuring that the two positions are saved simultaneously. The data from the reference sensor is saved on the hard-disk drive of the control PC, while the capacitive sensor data is saved in the external RAM of the DSP Board. After an acquisition ends, the data from the two sensors is transferred (offline) to an auxiliary PC, where the position deviation is calculated.

One possibility to generate the four excitation-voltages is shown in Figure 2.39. For long encoders and high excitation voltages (providing improved signal-to-noise ratio), the sinusoidal voltage may be generated in two stages. First, a small MOSFET H-Bridge
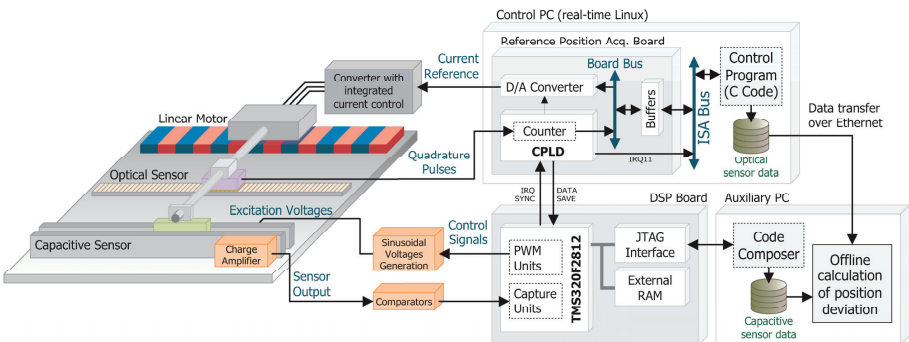
Figure 2.38: Overview of the setup used for evaluation with sinusoidal excitation
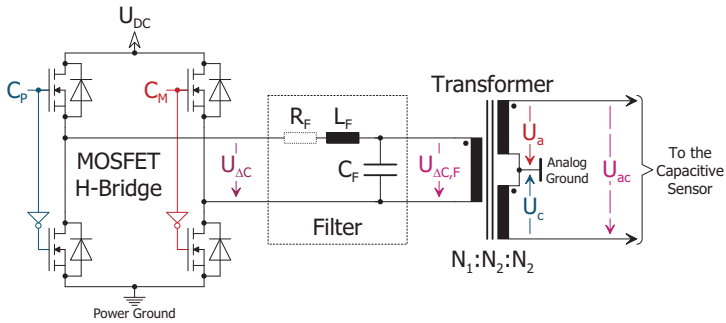
Figure 2.39: Possible implementation of the sinusoidal voltages generation

generates the PWM voltage $U_{\Delta C}$, as shown in Figure 2.39 and Figure 2.40.

Then, a passive L-C filter with a high quality factor Q suppresses the harmonics of the PWM voltage, yielding $U_{\Delta C,F}$. The line-to-line voltage $U_{ac} = U_a - U_c = A_h \cos(\omega_h t)$ is generated by a three-winding transformer, which also separates the Power Ground of the H-Bridge (switching) and the Analog Ground (charge amplifier). By appropriately choosing the winding numbers N_1 and N_2 of the transformer, high excitation voltages can be obtained even with a small voltage U_{DC}. The second line-to-line voltage $U_{db} = U_d - U_b = = A_h \sin(\omega_h t)$ is generated by a similar circuit (not shown in Figure 2.39).

The gate signals C_P and C_M for the H-Bridge are produced by the PWM circuitry of the DSP. They are 180° shifted and have a pulse width of 120°, which eliminates the 3rd harmonic from the voltage $U_{\Delta C}$.

All the frequencies contained in $U_{\Delta C}$ must be eliminated by filtering, except the fundamental component (frequency f_h). This would normally be achieved by a band-pass filter, with the band-pass frequency $f_h = 100$ kHz. The Bode diagram of such a filter (2nd order, with a quality factor Q = 20) is shown in Figure 2.41.

The phase response of the band-pass filter has the steepest slope for $f_h = 100$ kHz,
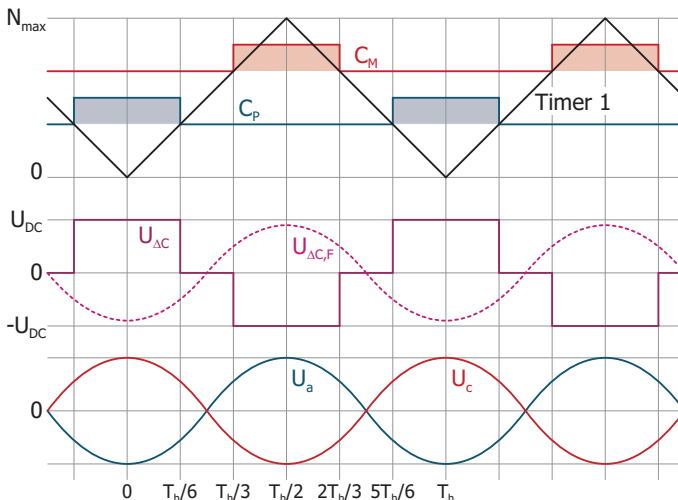
Figure 2.40: Signal diagram: Sinusoidal voltages generation
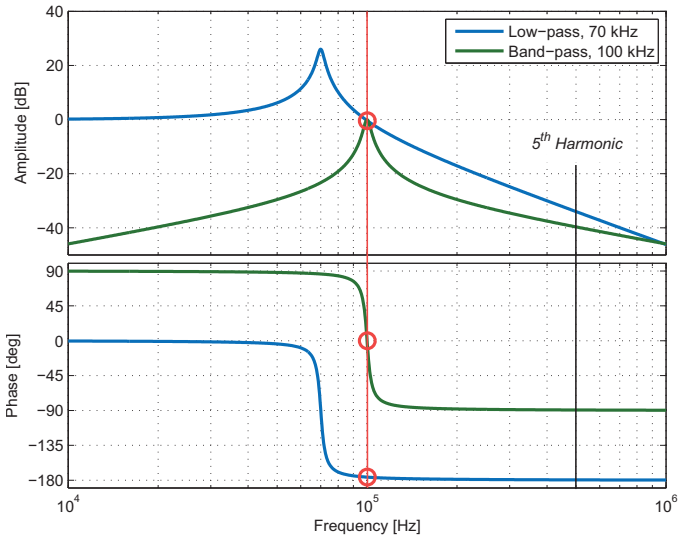
Figure 2.41: Filter Bode diagram

which can have a negative influence on the phase measurement used to determine the sensor's position. This is why a 2nd order low-pass filter will be used instead, with the same quality factor and a corner frequency of 70 kHz (see Figure 2.41).

The amplification of the low-pass filter at 100 kHz is very close to unity (0,95) and the phase at the said frequency is almost flat (-178°). The highest harmonic, which will appear in the filtered signal, is the 5th. Before filtering, it has an amplitude of 20% of the fundamental, and is attenuated by the filter to 2%. This means that the 5th harmonic will be present in the filtered signal with an amplitude of 4‰ from the fundamental of $U_{\Delta C}$.

For the experimental tests, a simplified method was used to generate the sinusoidal excitations, as schematically shown in Figure 2.42. The signals C_P and C_M (from the PWM units of the DSP) are subtracted by a differential input amplifier, in order to obtain $U_{\Delta C}$. The operational amplifier used also allows for biasing of the output, which was used to correct the offset of $U_{\Delta C}$. Filtering is realised by integrated filter circuits [40]. Due to restrictions of the Qf product of these circuits, the excitation frequency was limited to 20 kHz. A variable gain amplifier is used to correct the gain errors of $U_{\Delta C,F}$, and a rail-to-rail, differential output, operational amplifier generates the two complementary outputs U_a and U_c. The output amplifier's power supply is ±15 V, which allows for a swing of ±30 V in the line-to-line voltage U_{ac}.

For generating the other two voltages (U_b and U_d) an identical circuit was used, fed by S_P and S_M, which are 90° shifted with respect to C_P and C_M.

The output of the charge amplifier feeds a comparator, which detects the positive zero-crossings of the signal U_{out}. A Capture unit of the DSP uses the output of the comparator in order to determine the phase of U_{out}. The timer used by the Capture unit is hardware-synchronised with the timers used to generate the control signals. A second comparator could be used to determine the negative zero-crossings of U_{out}, reducing thus to half the acquisition latency.

In Eq. 2.33 it was assumed that the generation of the sinusoidal excitation voltages and the charge amplifier introduce no phase shift between the fundamental of $U_{\Delta C}$ and
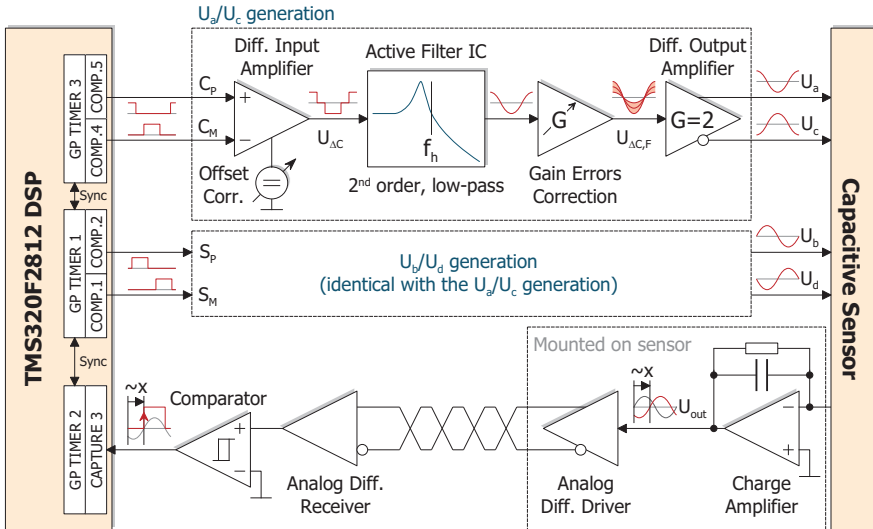
Figure 2.42: Proof-of-concept implementation of the sinusoidal voltages generation

U_{out}. In reality such a phase shift occurs, biasing the absolute position of the capacitive sensor. Eq. 2.33 can be rewritten:

$$U_{out}(x) = A_h K \cdot \cos\left[\omega_h t - (x_P + \varphi_C)\right].$$

Eq. 2.34

The unknown phase φ_C introduced by the filter and the amplifiers can be determined during initialisation, when the point of reference is crossed.

Figure 2.43 shows in blue line the difference between the reference sensor's position and the capacitive sensor's position measured position, without correction.

A correction table was generated without the use of a reference sensor by an off-line, zero-phase filtering process (using Matlab function filtfilt [41]). Only a correction for the average values was used, as there is not enough storage for a period individual correction table in the evaluation board. The result with this simple correction is shown in green line in Figure 2.43.

Figure 2.43 and Figure 2.36 show very similar results of the position deviation for the two different methods (sinusoidal and square-wave excitation, respectively), but we have to note, that only an average correction was used in Figure 2.43. Using a correction individually for each period of the sensor should improve the result of Figure 2.43.

The main difference between the two methods is the expense of the implementation. Using the square-wave excitation method requires a fast 12-bit analog-to-digital converter and computational resources to perform divisions and calculate the atan2 function.

For production of sensor systems in large quantities, it is profitable to do the necessary signal processing in an ASIC. It is much easier to implement the signal processing used for the sinusoidal excitation method in an ASIC: no analog-to-digital converter is necessary, and no resource-demanding calculations like division or atan2 function are required. This will result in a rather simple and cheap ASIC for the sinusoidal excitation signal processing. For lower production quantities it is no problem to implement the processing required by the sinusoidal excitation method in an FPGA.
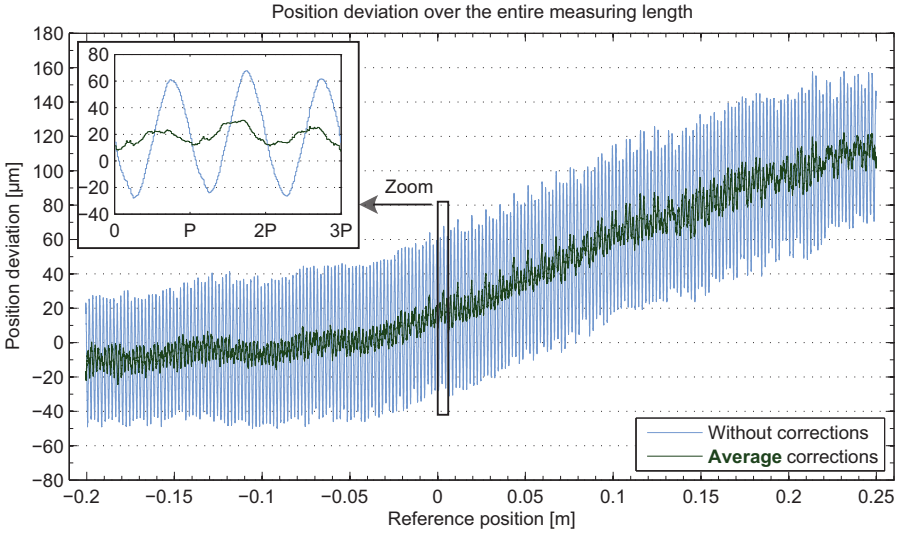
Figure 2.43: Position deviation (using sinusoidal excitation)

There are several ways to reduce the errors of capacitive position sensors, by software or by hardware. A software method is the reduction of the harmonics by more sophisticated correction tables. To generate them, large amounts of sampled data are necessary to identify the characteristics (amplitude and phase) of some dominant harmonics. The more sophisticated correction table should be loaded into a flash memory located at the sensor. This could be preferably done at the manufacturer's test bed during the final production test, before the sensor leaves the factory.

A hardware-oriented way to reduce harmonics is an improved shaping of the modulating or even of the transmitting electrodes. The model derived in section 2.3.2 and used in the implementation of both excitation methods is based on the assumption of homogenous field distribution. But in reality, the field will be quite different from this simple assumption. As discussed in section 2.3.2, a large number of 3D-FEM calculations may be used to optimise the shape of the modulating electrode such that the position of the slider can be extracted precisely from U_{out}. This implies, that the position dependent capacitances contain a pure sinusoidal position-dependency. The optimisation will be a time consuming, iterative design procedure, where many 3D-FEM calculations have to be performed for different shapes of the electrodes.

3 Linear Drive for Material Handling

In this chapter the position acquisition and control of a linear drive for material handling, currently under development at our department, is discussed. The chapter is organised as follows: First, in section 3.1, the system architecture is presented. After an overview of the power electronics and control topology, the linear machine used in our experimental setup is briefly introduced. Then, details regarding the power electronics and control cabinet, and the control software are given. The implementation at the linear drive of the optical sensors system described in section 2.2 is presented in section 3.2. In section 3.3, the implementation of the EMF-based, sensorless speed control at the transport section of the drive is discussed. Finally, section 3.4 presents the synchronisation between the position given by the sensors, and the estimated one, when the vehicle leaves or re-enters a processing station.

3.1 System Architecture

In order to allow for individual motion control of several vehicles, the active track of the linear machine is separated into many segments, each segment being fed by the power stack of a dedicated inverter. For the application discussed in this thesis, a relatively small number of vehicles is intended, so a system with one controller assigned to one vehicle is best suited, as it involves the lowest possible number of controllers.

An overview of the power electronics and control topology is given in Figure 3.1. There, an Inverter Bus provides communication between each Vehicle Controller (VC) and each Power Stack (PS) within every 100 μs control cycle. All power stacks are equipped with a CPLD-based "Power Stack Interface (PSI)" and each vehicle controller is attached via an FPGA-based "Vehicle Controller Interface (VCI)" to the Inverter Bus.

Access to the Inverter Bus is controlled in a stringent time-slot regime by an Inverter Bus Master. All actions within the controllers and all power stacks, including switching of the IGBTs and sampling of the fundamental component of the stator currents, are strictly synchronized to the timing generated by the Bus Master.
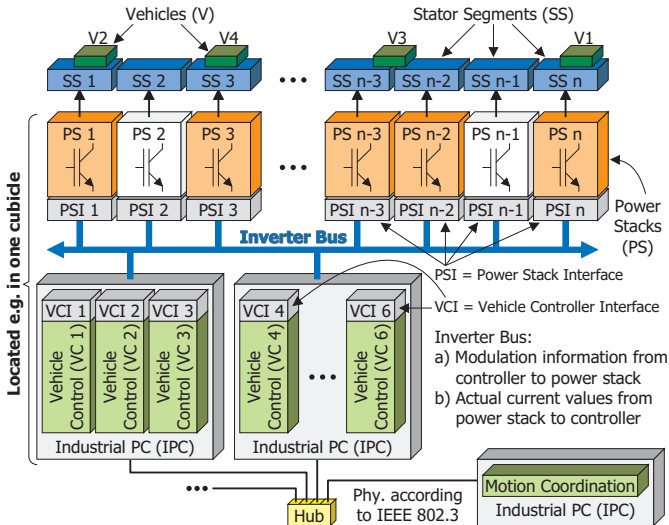
Figure 3.1: Overview of the system architecture

Physically, the Inverter Bus is a 16-bit parallel bus, implementing the RS-485 differential signalling standard, identical with the Sensor Bus described in Section 2.2.4 (actually, the Inverter Bus was developed at our department prior to the Sensor Bus, and the latest was designed in such a manner as to take advantage of the already existing hardware). The Inverter Bus Master requires no actual hardware, being integrated in the FPGA firmware of the first Vehicle Controller Interface.

Figure 3.2 presents a simplified diagram of the Inverter Bus communication protocol [42]. As aforementioned, there are three types of communication partners: the Inverter Bus Master, the Vehicle Controllers (through their respective VCIs) and the Power Stacks (also through their PSIs). Within every 100 µs control cycle, the Inverter Bus Master allocates a communication time slot of 10 µs for each Vehicle Controller. E.g., at time t = 0 µs in
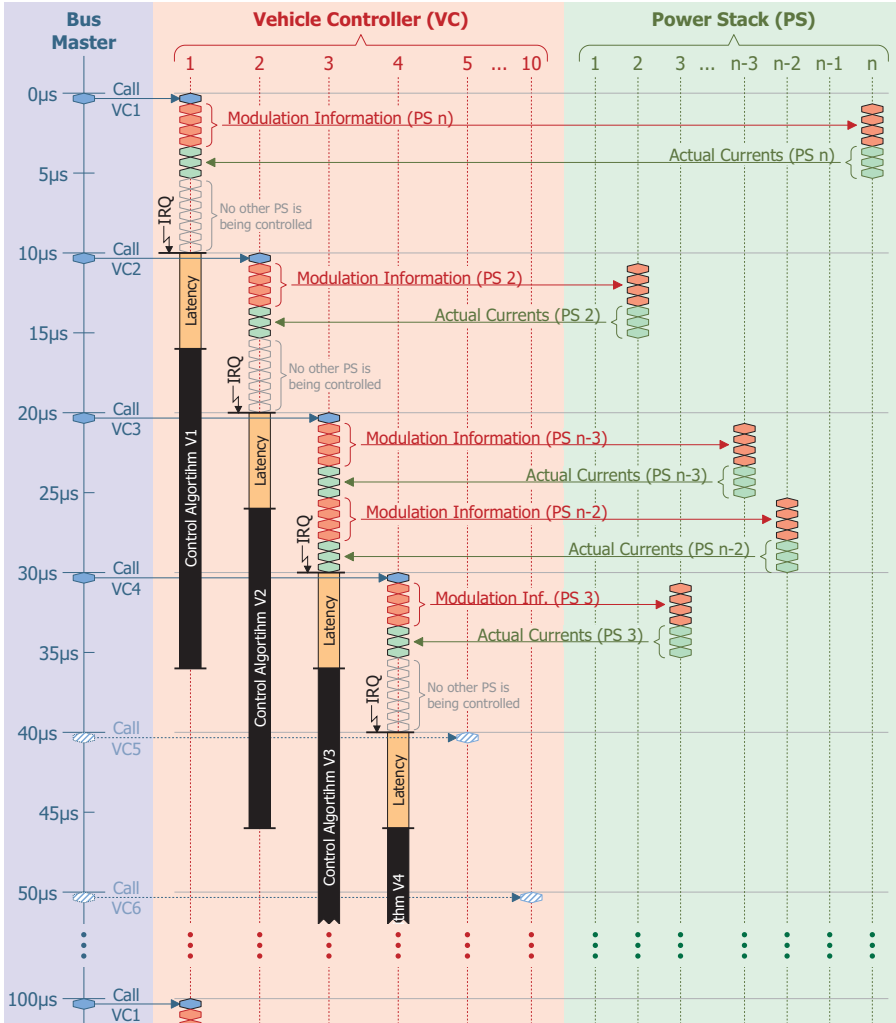
Figure 3.2: Inverter Bus Protocol. Source: [42]

Figure 3.2, the Bus Master calls the first Vehicle Controller (VC1). Each VC keeps track of the Stator Segment where its corresponding vehicle currently is, i.e. of the Power Stack whose currents must be controlled – in this case, PSn.

After being called, VC1 sends the modulation information to PSn (4 x 12 bits): the first 12 bits encode the address of the Power Stack (6 bit) and the commanded state of each inverter phase – upper IGBT on or lower IGBT on or both off – (6 bit). The next 3 x 12 bits are the switching times in the three inverter phases. After receiving the modulation information, the addressed PSn answers to VC1 by sending the actual values of the three phase currents (3 x 12 bits), which currents have been sampled in the middle of the zero-vector interval. The communication between VC1 and PSn takes about 5 µs (the half of the 10 µs time slot accorded to VC1).

A special situation appears, when a vehicle passes from one segment to the next one and, for a time, occupies two stator segments, as V3 in Figure 3.1. In this case, VC3 has to control the currents of both PS n-3 and PS n-2. The necessary communication utilises fully the 10 µs allocated to VC3 (t = 20 …30 µs in Figure 3.2).

After receiving actual currents values, a Vehicle Controller can proceed with the calculation of the algorithms concerning current, speed and position control for the corresponding control cycle. More than one VC can be calculated by one industrial PC within each 100 µs, under Linux-RTAI operating system.

3.1.1 Linear Machine

*Preface: As mentioned previously, the design of a new type of electrical machine and all the mechanical constructions was done by Institute for Electrical Machines, Traction and Drives at Technische Universität Braunschweig. Our department is responsible for hard- and software of power electronics and control of the system. In order to understand the operation of the control system, some information on the electrical machine and the mechanical construction is included here, although this is not part of the author's work.*

As a high thrust-force density in combination with a rather large air gap and a curvilinear track is necessary for the application, a Permanent Magnet Synchronous Linear Machine (PMSLM) in the long-primary (moving magnets) configuration corresponds best to the requirements. The linear machine designed for this project has three-phased, single layer, concentrated coils (tooth windings), which simplifies the modular construction of stator sections of different lengths.

In order to further increase the modularity, the stator is constructed from individual stator elements, as the one shown in Figure 3.3 (a), which stator elements are formed by pressing from a Soft Magnetic Compound (SMC). The shape of these SMC elements allows for small angular and rotational displacements in every space direction between
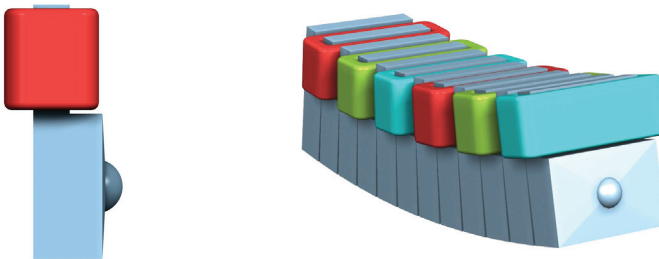
a) Stator element with winding   **b) Stator elements in up/downhill configuration**

Figure 3.3 Stator elements. Source: [43]
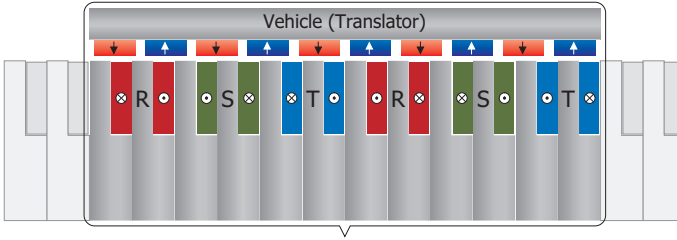
Figure 3.4: Elementary machine

Characteristic	Value
Number of poles	10
Pole pitch	24 mm
Magnet height	8 mm
Vehicle (translator) length	240 mm
Vehicle yoke material	Solid iron
Air gap	1…10 mm
Number of stator slots	12
Slot pitch	20 mm
Stator material	Somaloy® 500

Table 3.1: Linear machine characteristics. Source: [43]

adjoining stator teeth. In this way, curved sections of the track can be constructed – see, e.g. in Figure 3.3 (b) a row of SMC elements arranged in up/downhill configuration.

An elementary machine – the smallest possible functional unit – is sketched in Figure 3.4. It consists of 12 SMC elements (teeth) on which 6 concentrated coils are wound (one coil every second tooth). Corresponding to them, there are 10 magnetic poles on the vehicle ("10/12" configuration). The main characteristics of one elementary machine are listed in Table 3.1.

The linear machine used in our experimental setup has an oval track, with the dimensions of ca. 3 m x 4 m. It is composed of 18 independently fed stator segments, which are grouped in 5 sections. The main data for the segments of each section is listed in Figure 3.5, where:

- n_w – number of turns of the coils
- n_p – number of stators connected in parallel
- n_s – number of elementary machines connected in series
- F* – nominal thrust force
- I* – nominal current
- δ* – nominal air gap
- δ – actual air gap

Section 1, containing SS1 and SS2 is a straight section. The two segments are double sided (p = 2), and each side of each segment is formed by the series connection of two elementary machines (k = 2). The optical sensors system described in section 2.2 of this work is implemented on this section of the linear machine (see chapter 3.2). All the other sections of the machine are single sided (p = 1).

The nominal values of the currents I* from Figure 3.5 were used for the dimension-
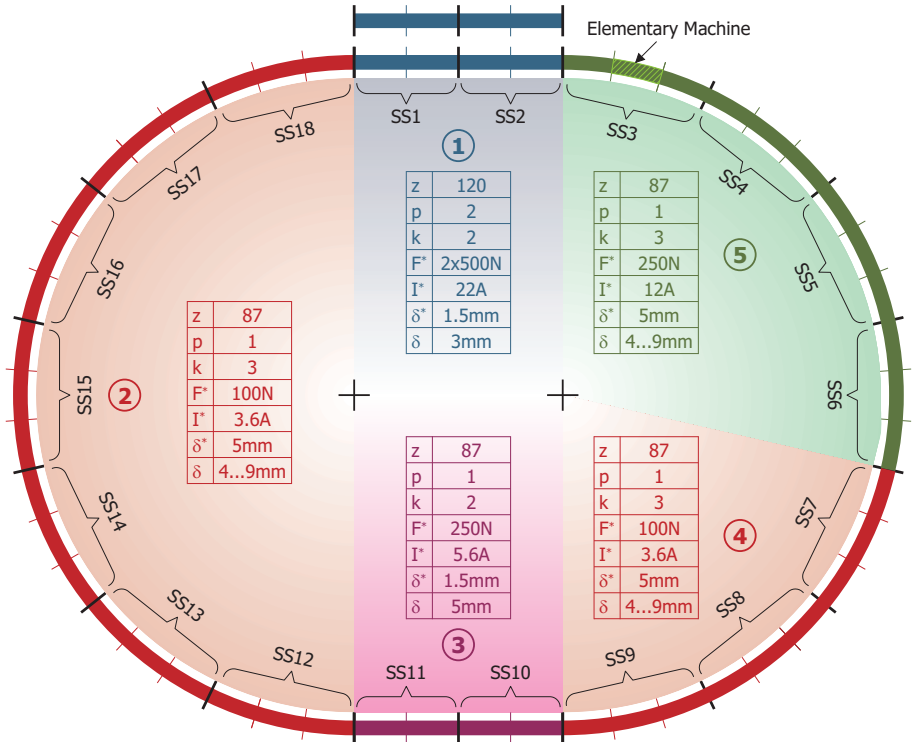
Figure 3.5: Segments data

ing of the inverters for each section of the track (see 3.1.2.1). The nominal values of the electrical force F* can be obtained with the nominal values of the current I* when the air gaps of the different sections are at their nominal values δ*. However, it was not possible to set the air gaps to their nominal values, due to the high magnetic forces, which would have lead to the breaking of the SMC stator elements. The larger actual values of the air gap, δ, require higher currents in order to achieve the nominal forces (which currents are in some parts of the track higher than the ones that can by provided by the inverters).

It must also be noted that the air gaps δ are not only larger than their nominal values, but, within a given section, they also vary – leading to variations of the inductances and of the EMF constants of the segments.

The machine was delivered to our department in more stages. First, only section 1, together with a rigid vehicle was available (as shown in Figure 3.6). This temporary setup was used for the initial testing of the control algorithms (see also section 3.3.3). When the other sections of the linear machine, together with an articulated vehicle (capable of travelling through the curved section of the track), were available, it was possible to test the control algorithms on the entire linear machine (see section 3.3.4, section 3.4).

Figure 3.7 shows a transversal cut through the double sided section of the machine, together with the vehicle [43]. The two sides of the stator are mounted, at a 70° angle, in retaining brackets, which brackets are spaced along the track at distances equal to the length of the vehicle (240 mm). On the brackets there are also mounted two parallel tubes, on which the vehicle travells on the carrying rollers and which also serve as guides (see
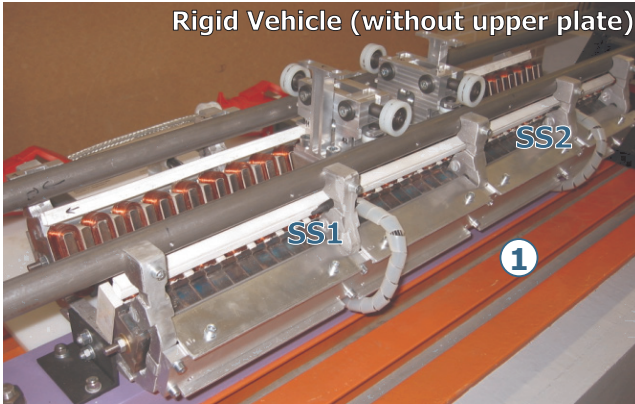
Figure 3.6: Straight, double sided section of the machine, with rigid vehicle

the horizontal guiding rollers in Figure 3.7).

On the bottom side of the vehicle there is the magnets carrying block, which slides between the V-shaped stators. On this block the permanent magnets are mounted (10 on each side). On the topside of the vehicle there is a load carrying plate, which also serves as a support for the measuring scale of the optical sensors.

In Figure 3.8 a photo of the articulated vehicle [43] is shown. The main difference between the rigid vehicle and the articulated one is that the magnets carrier block of the last is split in two halves, which are connected through a spherical joint, allowing them to
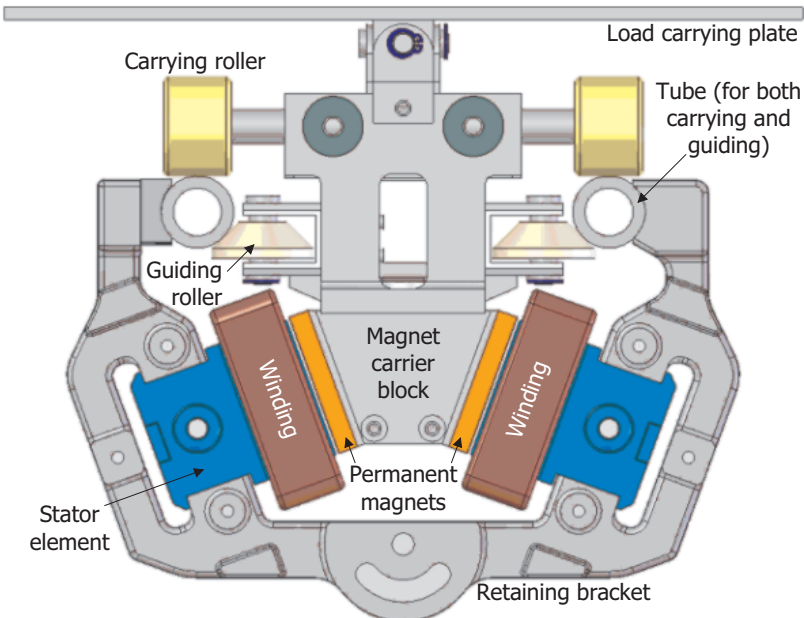
Figure 3.7: Transversal cut through the double sided section of the machine. Source: [43]

rotate with a small angle (horizontally, as well as vertically) with respect to each other. The two magnet carriers and the spherical joint are held together by a spring-tensioned steel cable, which passes through the centre of the sphere.

The entire linear machine at our experimental setup is depicted in Figure 3.9. The curved sections of the track are super-elevated, in order to allow for high-speed travelling. This super-elevation requires a precise, three-dimensional bending of the two guiding tubes, which is difficult to realise.

In order to compensate for unavoidable variations from the nominal value of the distance between the tubes, the outer guiding rollers (the ones that are located towards the outside of the track, i.e. opposed to a single sided stator) of the articulated vehicle are mounted on springs. This lets the vehicle balance the variations in the distance between the tubes, but with the disadvantage of a highly increased friction force between the vehicle and the track.

Measurements of the static friction, conducted at different points along the carriage-way, have shown it to be very high (up to 200 N), at some points even higher than the nominal electrical force.

Due to the modular construction of the linear machine and to the tolerances in the dimensions of the SMC stator elements, there are gaps in the machine's winding between consecutive stator segments, which lead to a phase displacement between the two corresponding EMF vectors. The influence of these gaps on the control algorithm will be discussed in detail in section 3.3.
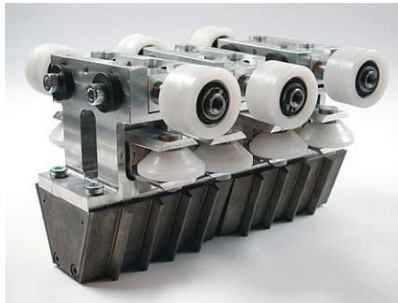
Figure 3.8: Articulated vehicle (without magnets and load carrying plate). Source: [43]
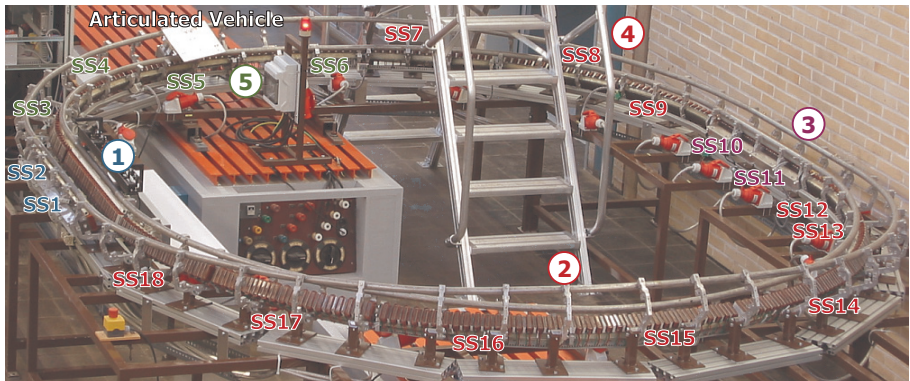
Figure 3.9: Entire linear machine

3.1.2 Power Electronics and Control

The power electronics and the Control PC are all hosted in a cabinet, located near the linear machine (centralised architecture). From this central cabinet, the outputs of the 18 power stacks are fed through cables (4 … 11 m) to the 18 segments of the machine. Two cabinets, as described in section 3.1.2.1, were built at our department, one for each university lab involved in the project. The Control PC, including the software architecture and the real-time control software is presented in section 3.1.2.2.

3.1.2.1 Cabinet

Figure 3.10 shows the interior view of one cabinet. The 18 Power Stacks (from the Simodrive 611 series [44], Siemens company) are located in the upper side of the cabinet were chosen according to the nominal currents listed in Figure 3.5. There are devices of two different power ratings:

- **PS1 … PS6** are of type 6SN1123-1AA00-0CA1, containing 50 A IGBT modules. Their nominal current, at the 5 kHz switching frequency used in our system, amounts to 19 A (rms), whilst with a S6-type load-cycle they can provide a 22 A (rms) current. These Power Stacks feed sections 1 and 5 of the linear machine, which have the highest power demand.

- **PS7 … PS18**, of type 6SN1123-1AA00-0BA1, contain 25 A IGBT modules and have a nominal current of 7.5 A (rms) at 5 kHz. They feed the remaining three sections of the linear machine.
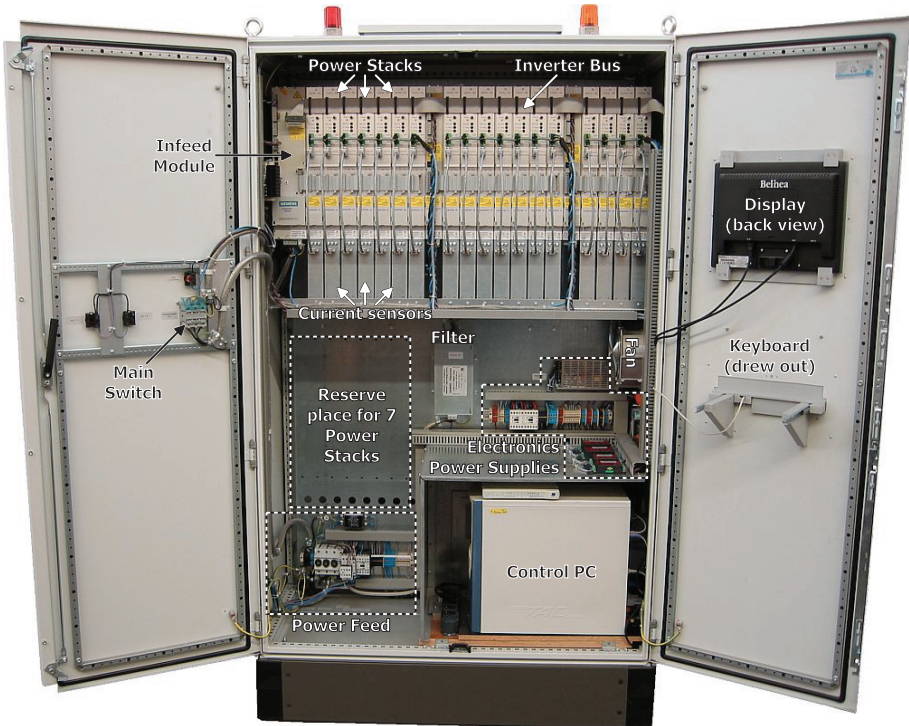
Figure 3.10: Cabinet

The Power Stacks are equipped with internal current sensors, but they are only used for over-current protection. The feedback values of the phase currents needed in control are acquired using external LEM sensors, which provide better quality signals (as required e.g. by sensorless control).

A diode bridge with integrated brake chopper, contained in the Infeed Module, supplies the DC link bus, which feeds all the Power Stacks. This module (6SN1145-1AA01-0AA1) has a nominal power of 10 kW and a peak power of 25 kW.

The Infeed Module also supplies the internal electronics (IGBT drivers, internal current sensors, protection circuitry, etc.) of the first seven Power Stacks, while for supplying of the internal electronics of the remaining Power Stacks two Monitoring Modules (6SN1112-1AC01-0AA1) are necessary. The Infeed Module and the Monitoring Modules are connected to their respective Power Stacks through a Siemens proprietary Devices Bus (Gerätebus) – see the diagram in Figure 3.11.

Each Power Stack is equipped with a plug-in Power Stack Interface card, developed at our department [45]. This card:

- implements the Inverter Bus communication protocol;
- generates the 6 control signals for the IGBT drivers, based on the values of the switching times received through the Inverter Bus from the Vehicle Controller;
- digitises the three current signals from the external LEM sensors, and sends them back to the Vehicle Controller;
- monitors the two internally measured currents of the Power Stack (over-current protection);
- monitors the temperature signal available from the Power Stack (over-temperature protection);

All 18 Power Stack Interfaces are connected through the Inverter Bus to the Vehicle Control Interface Board, located in a PCI slot of the Control PC. The Vehicle Control Interface Board and the Position Interface Board are two identical PCI cards, as described in section 2.2.5. The only difference between the two is the FPGA firmware, which implements the Inverter Bus communication protocol for the first one, and the Sensor Bus protocol for the second one. The Position Interface Board is connected through the Sensor Bus to the sensors pre-processing unit (Sensor 2 Digital), which is located outside of control cabinet.

The cabinet circuitry is connected to the 400 V grid by the hand-operated main switch K0, located on the front door of the cabinet. When this switch is closed at power-up, only the Control PC and the interface electronics are supplied (1L3 – N).

In order to supply the Siemens devices (the Infeed Module, the two Monitoring Modules and the 18 Power Stacks) the self-holding contactor K1 must also be closed. This is realised in two steps: first the control software must close contact K2 (located on the Supply Control Board, which will be described later in this section). It is not possible to supply the power electronics if the control program is not running. In the second step, the T2 ("On") push-button, located on the front door of the cabinet must be pressed by the operator. Pressing the T1 ("Off") push-button opens the contactor K1 and disconnects the supply of the Siemens devices.

Closing the K1 contactor supplies the electronics of the Siemens devices, but not the DC link. An extra step is needed for this: the control software must also close the contact K3 (located, like K2, on the Supply Control Board). This determines the closing of an internal contactor of the Infeed Module, thus supplying the power sections of the Siemens devices.
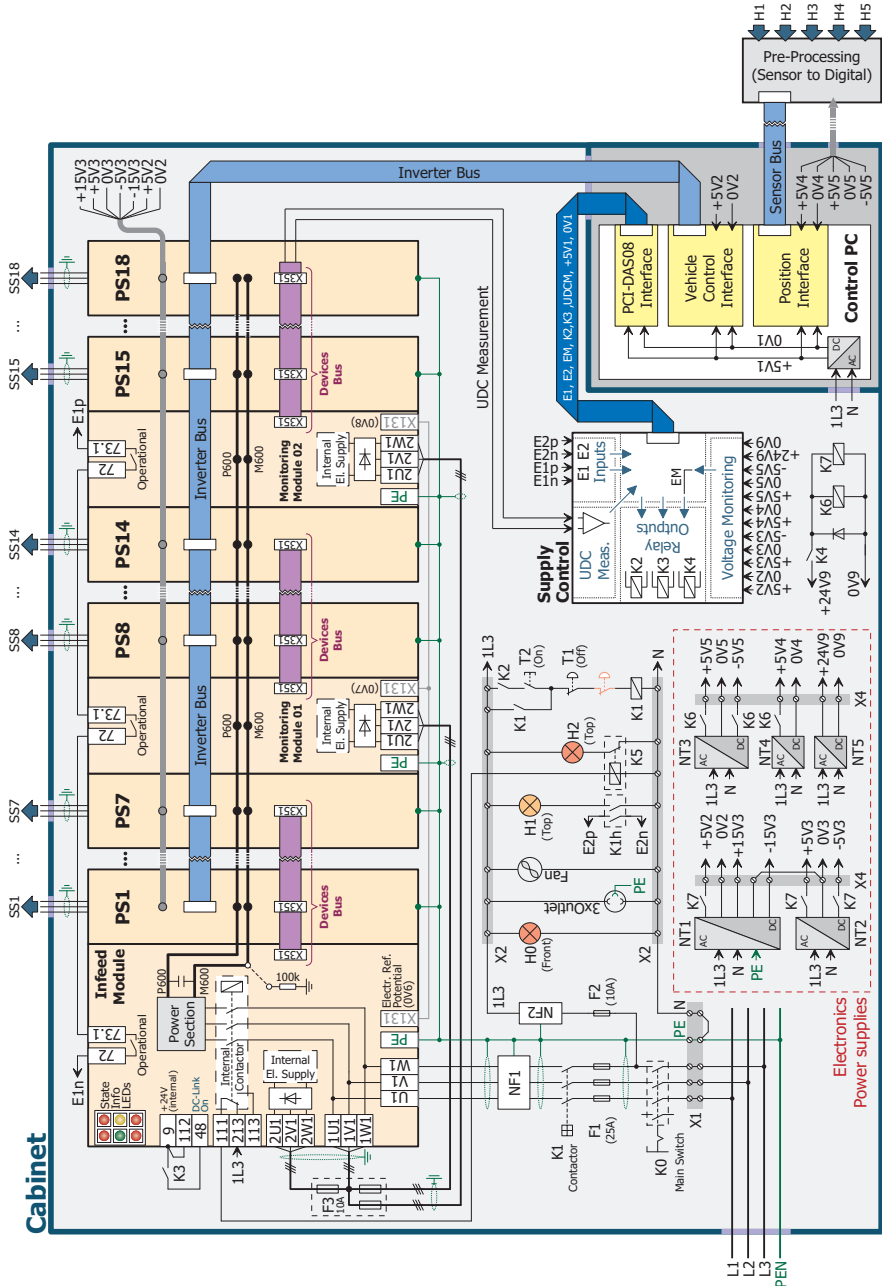
Figure 3.11: Cabinet Diagram

After the Control PC has closed the contact K2, it will monitor the input E2. This input is connected to an auxiliary switch of the contactor K1 and signalises the state of the contactor.

During the running time of the control algorithm, the Control PC also monitors the state of the Infeed Module and of the two Monitoring Modules, through the input E1. Faults like phase failure or under-voltage can be detected with the help of this operation state signal.

The power supply units NT1 … NT5 generate the supply voltages required by the interface electronics:

- The Power Stack Interfaces and the Vehicle Control Interface Board are fed with the +5V2 voltage for the digital circuits and with the ±5V3 and ±15V3 for the analogue ones.

- The Position Interface Board and the sensors pre-processing unit (Sensor to Digital) are fed with +5V4 (digital) and with ±5V5 (analogue).

- The +24V6 generated by NT5 are used to supply the relays and switches on the Supply Control Board.

The Supply Control Board is connected through the PCI-DAS08 [46] Interface Board (from Measurement Computing company) to the Control PC. The functions to the realisation of which the Supply Control Board takes part are:

- Switching (on/off) and monitoring of the DC link voltage (through the E1 and E2 inputs, and the K2 and K3 relay outputs – as described earlier).

- Measuring of the DC link voltage: The Siemens Devices Bus provides a voltage proportional to the DC link voltage. This signal is routed to the Supply Control Board, where is conditioned and then sent further to the PCI-DAS08 Board. Here the signal is digitised and made available to the control software.

- Switching (on/off) and monitoring of the interface electronics power supplies. The K4 relay on the board is used to control the K6 and K7 switches. At power-up these switches are simultaneously closed, only after all the supply voltages (+5V2, ±5V3, +5V4, ±5V5) have all reached their operational values. This ensures the simultaneous supply of all the interface circuitry, independent of the power-up time constants of the different power supply units (this is especially critical for the analog-to-digital converters, which have both analogue and digital power supplies). These voltages are also monitored during operation, and in case of a failure the K6 and K7 switches are opened. By interrupting the +5V2 power supply of the Power Stack Interfaces, all the Power Stacks' IGBTs are also turned off. The state of the interface electronics power supplies is transmitted to the Control PC, through the EM signal.

A detailed diagram of the Supply Control Board is given in Appendix 5.3.

3.1.2.2 Control Software

For the implementation of the control algorithm, a standard PC (equipped with an Intel Pentium 4 processor and 512 MB of memory) was used. The only hardware modification brought to the PC was the addition of the three PCI interface boards (see Figure 3.11 in the previous section).

Using a PC for the implementation of the control provides some advantages compared with other possible architectures (e.g. based on a DSP board), among them being a fast CPU, with fully integrated floating-point support (FPU) and a large amount of primary memory (RAM), which can be used as a transient recorder (this is especially useful during the development and testing of the control algorithm).

As operating system a Vector Linux 5.8 SOHO distribution [47], based on the Linux kernel 2.6.17 [48] was chosen. Included in this distribution are also tools for the software development (code editor, compiler, etc.)

The Linux kernel by itself is not fully pre-emptive [49] (i.e. there are certain kernel functions which cannot be interrupted), so it cannot be directly used for applications which require hard real-time (very fast and deterministic response to real-time events).

In order to be able to meet the real-time requirements needed in control, the Real Time Application Interface (RTAI) kernel extension [50], developed at Dipartimento di Ingegneria Aerospaziale from Politecnico di Milano, was used. This extension introduces a small real-time kernel between the system hardware and the Linux kernel, used to manage the hardware events (see Figure 3.12). This real-time kernel captures the hardware interrupts and realises the scheduling of the corresponding real-time handlers (interrupt service routines), with very short delays [51], [52].

From the point of view of the real-time kernel, the Linux kernel is just another task, which is run in the idle time (the time when no real-time task is active), and which can be interrupted at any time when a hardware interrupt requires service; thus, the Linux kernel becomes fully pre-emptive.

A kernel patch, containing the RTAI Hardware Abstraction Layer (HAL) was applied to the Linux kernel source code, and the kernel was re-compiled. In this process certain features, which might have interfered with the real-time control were also disabled (e.g. APM – Advanced Power Management, or CPU frequency scaling). Then RTAI 3.5 was installed. After installation, the RTAI components can be loaded and unloaded using the normal Linux mechanism (insmod – for inserting a module into kernel, rmmod – for removing a module from kernel, etc.), so it is possible to turn on and off all or some of the real-time capabilities of the operating system, as required.

The control of the linear machine is implemented in the "Real-Time Module" (Figure 3.12), which is running in the kernel space. Because in this module very fast response times are necessary, the use of some Linux system functions (e.g. keyboard input, display output, hard disk access) must be avoided, due to the delays they can introduce.

A second application program, the "User Interface" will be used to manage these functions (user input, data saving on the hard disk, etc.). This application has no real-time requirements, so it can run in the user space, like any normal Linux application. The User Interface program, a screenshot of which is presented in Figure 3.13, was generated with the help of Glade 2.12.1 [53], an interface builder based on the GTK+ [54] toolkit.

The communication between the Real-Time Module and the User Interface is accomplished through real-time FIFOs – Linux character devices, used for inter-process communication. From the Real-Time Module, the RTAI-provided FIFO manipulation functions (create, read, write, destroy, etc.) are used, whilst the User Interface, located in the user space, can access these FIFOs like normal (sequential) files.
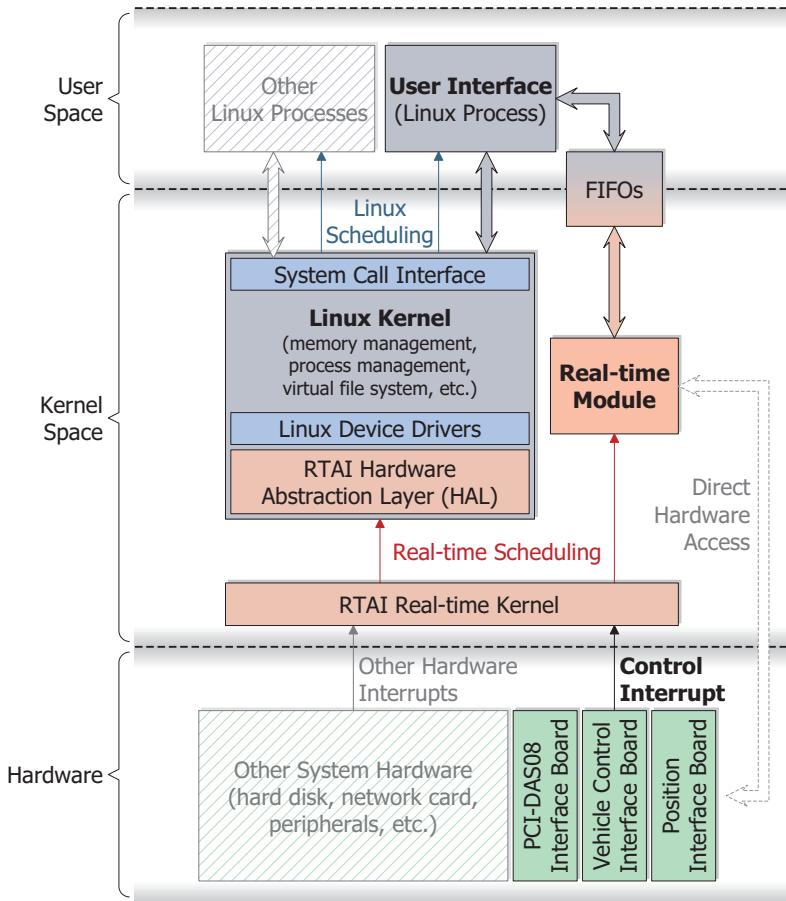
Figure 3.12: Linux-RTAI scheduling

In order to coordinate the starting and stopping of the Real-Time Module and User Interface a small Linux shell script was written:

- The shell script first prepares the real-time environment, loading into the kernel the necessary RTAI modules:
  - → rtai_hal.ko – the hardware abstraction layer
  - → rtai_sched.ko – the real-time scheduler
  - → rtai_fifo.ko – real-time FIFOs related functions
  - → rtai_math.ko – this module redefines the mathematical libraries functions, for use in the real-time environment.
- After the RTAI modules are successfully loaded, the real-time module used for the machine control (rt_control.ko – referred to as the Real-Time Module throughout this section) is also loaded. When first loaded, the Real-Time Module is not addressing any Power Stack Interface, the feeding of the DC Link is disabled, and no control algorithm is running.
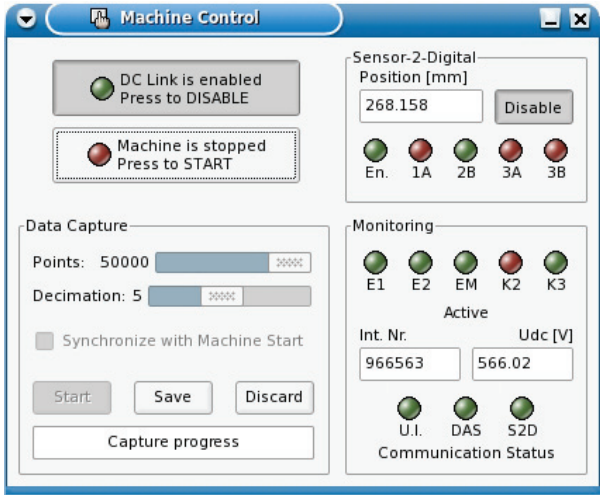
Figure 3.13: Screenshot of the User Interface window

- After the Real-Time Module is loaded, the script runs the User Interface program.
- Through the User Interface commands can now be sent to the Real-Time Module (power-up the DC Link, start the control algorithm, capture data, etc.).
- As long as the User Interface is still active the shell script hangs, waiting for its termination.
- When the user closes the User Interface program, the shell script resumes, unloading all the previously loaded kernel modules, in reverse order.

In Figure 3.14, a measurement of the interrupt latency, interrupt jitter and control algorithm processing time is depicted. The measurement was done while the complete control algorithm (including simultaneous current control for two stator segments, position acquisition from the optical sensors, and sensorless, EMF-based speed control) was running on the Control PC, under RTAI-Linux.

The signal in the upper part of the figure is the control interrupt, generated by the Vehicle Control Interface Board; it was measured directly on the PCI bus.

The second signal is one output bit of the Control PC's parallel port, which is set at the beginning of the Interrupt Service Routine and reset at its end – thus, this signal remains high as long as the control algorithm is executing.

The interrupt latency is defined as the time interval elapsed between the beginning of an interrupt request (the falling edge of the interrupt signal) and the beginning of the execution of the associated Interrupt Service Routine (the rising edge of the parallel port output signal, in this case).

The latency is mainly determined by delays in the interrupt controller hardware (the time needed for the interrupt to be transmitted from the PCI bus to the interrupt pin of the CPU) and by the time necessary for the context switch – when the CPU receives the interrupt request it finishes the current instruction, saves the values of its internal registers in the stack and then loads the appropriate interrupt vector (the start address of the Interrupt Service Routine) into the Program Counter [55]. For the considered system the interrupt latency has a mean value of 6.2 µs.
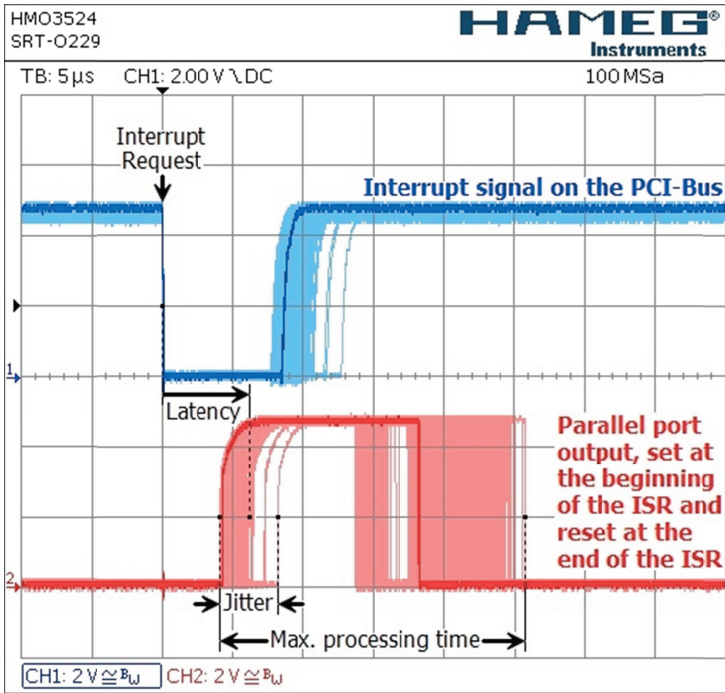
Figure 3.14: Measured interrupt latency, interrupt jitter and processing time

The variation of the interrupt latency is the interrupt jitter. One cause for this variation is e.g. the Direct Memory Access (DMA). When a device is granted Direct Memory Access, it will take control of the Memory Bus, so, if an interrupt request comes during a DMA transfer, the CPU must wait until the DMA transfer is complete in order to regain control of the Memory Bus.

For reducing the interrupt jitter and also the data transfer and interrupt overhead on the PCI bus, some un-essential devices (audio hardware, USB, serial port, etc.) were deactivated from the BIOS. For our Control PC, an interrupt jitter of ±2 µs was measured (see Figure 3.14).

The maximum processing time of the control algorithm for one vehicle does not exceed 22 µs (21.7 µs in Figure 3.14) – thus, it is possible to implement the control for two or three vehicles in the same PC.

It must be noted that, as previously stated, the Linux kernel code executes during the time when the Service Routine of the control interrupt is not running, so, if the execution time of the control algorithm is close to the 100 µs control cycle, it can have a negative impact on the responsiveness of the non-real-time part of the operating system.

Figure 3.15 shows a more detailed overview of the software architecture. The machine parameters, as well as the correction tables for the optical sensors, are stored in files on the hard disk of the Control PC. When the User Interface is started it will read the contents of these files and transfer it, using the real-time FIFOs, to the Real-Time Module (which, at this point, is already loaded into kernel).
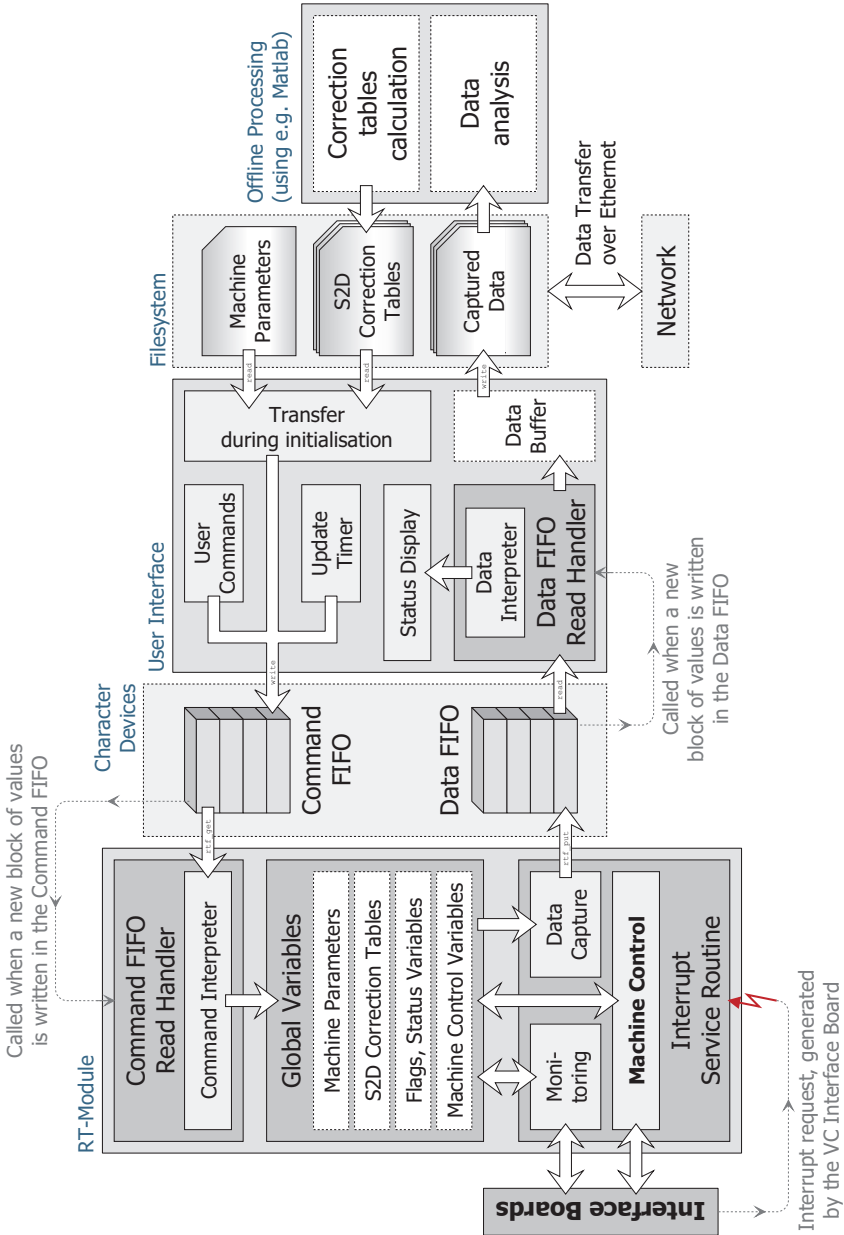
Figure 3.15: Software Architecture

The control data acquired from the Real-Time Module is also saved on the hard disk of the Control PC. From here it can be analysed locally (offline), using Matlab or sent over Ethernet to another computer for further processing.

For the communication between the Real-Time Module and the User Interface two unidirectional real-time FIFOs are used:

The Command FIFO (character device located at /dev/rtf0), used to send commands, together with their parameters, from the User Interface to the Real-Time Module.

The Data FIFO (character device located at /dev/rtf1), through which the values of the status and control variables of the Real-Time Module are transmitted to the User Interface.

Each of the two FIFOs has an associated read-handler (a function that is called automatically when a writing occurs in the respective FIFO – see Figure 3.15).

The following commands are defined for the Command FIFO:

- CMD_SET_PARAMS – Set the parameters of the machine segments.

  This command is sent from the User Interface during initialisation, once for each segment of the linear machine. The arguments for this command are:

  → The segment number for which the parameters are to be set.

  → The parameters of the respective segment (current PI controllers parameters, EMF observers parameters, etc.).

- CMD_S2D_CORRS – The correction tables for the optical sensors are sent to the Real-Time Module as arguments of this command.

  The command is also sent during initialisation, once for each logical read-head (see sections 2.2.6 and 2.2.7 for the definition of the logical read-heads and for the calculation of the correction tables). The command's arguments are:

  → The number of the logical read-head for which the corrections are sent.

  → The number of periods of the given read-head, for which the corrections were determined.

  → The amplitude and offset corrections of the sine and cosine signals for each period.

- CMD_DC_LINK_POW – Enable or disable the feeding of the DC Link.

- There is only one argument for this command, namely the required state of the DC Link.

- CMD_MACHINE_ON_OFF – Starts or stops the control algorithm of the linear machine. This command also has only one argument – the required state of the control algorithm.

  The control algorithm can be started from the User Interface only after the DC Link was powered-up, while the command for DC Link powered-down can only be given after the stopping of the control algorithm.

- CMD_CAPTURE – Used to acquire data from the Real-Time Module. As a result of this command, data will be sent from the Real-Time Module to the User Interface, using the Data FIFO (discussed below).

  There are two arguments for this command: the number of points to be acquired and the decimation of the data points.

  The capture command is used for two purposes:

  → To periodically acquire data from the Real-Time Module (using a 200 ms software Update Timer, implemented in the User Interface), in order to update the

> status and variables displays of the interface. In this case the command will be sent with the arguments 1 for the number of points and 1 for decimation.

→ To make a longer, user-initiated data capture (using the "Data Capture" section of the User Interface, see also Figure 3.13) In this case the user can choose the number of data points (up to 50,000), and the decimation (between 1 and 10). The length of the capture buffer was chosen by experience and it proved sufficient in most cases; but, if necessary, the amount of captured data can be increased (the practical upper limit is only the amount of available memory of the system). During a user-initiated data capture the periodical data request command is suppressed.

The second FIFO used in the communication is the Data FIFO. The Real-Time Module writes in this FIFO as a result of a capture command from the User Interface (see above). The User Interface is receiving the data on a read-handler function (similar to the one in the Real-Time Module, see Figure 3.15) and updates the "Status display" section or, if it is the case of a user-initiated capture, gathers all the received data in an internal buffer, which can be subsequently saved on the hard disk.

The Real-Time Module will write in the Data FIFO a number of data blocks, as required by the "number of points" argument of the capture command received; the "decimation" argument defines how many interrupts occur between successive data savings. One block of data contains up to 30 status and control variables of the Real-Time Module, all acquired during the same control interrupt. Data included here:

- The number of the interrupt when the data block was acquired.
- The state of the monitoring (for details, see Figure 3.18 and the accompanying explanations).
- The value of the DC Link voltage (if the DC Link was powered-up).
- The position and speed given by the optical sensors system (if the vehicle is inside sensors' region), including the current read-head number(s).
- Observed position and speed (when the vehicle is outside sensors' region).
- The currents, reference voltages and observed EMFs for each of the two controlled stator segments.

In the following, the structure of the Real-Time Module is briefly described. A more detailed structure, including all the defined functions and the main data definitions is given in Appendix 5.4.

The Real-Time Module consists of four principal functions:

- Module Initialisation function, called when the module is loaded into the kernel.
- Module Clean-up function, called at the module's unload from the kernel
- Command FIFO Read-Handler, called whenever a command is sent from the User Interface
- The Interrupt Service Routine, called every 100 µs through the Vehicle Control Interface Board's interrupt, and where the control algorithm is implemented.

In Figure 3.16 a block diagram of the module initialisation function is shown. When the module is loaded into the kernel, the required hardware (the three PCI interface boards) is first looked for.

The boards are detected using the Vendor-ID (VID) and Device-ID (DID) fields of the PCI structures associated with every PCI device [25]. If at least one of these interface boards is not present, then the control algorithm cannot function properly – so the initialisation function will not proceed: it will register an error message in the kernel message buffer and then exit with a failure code.
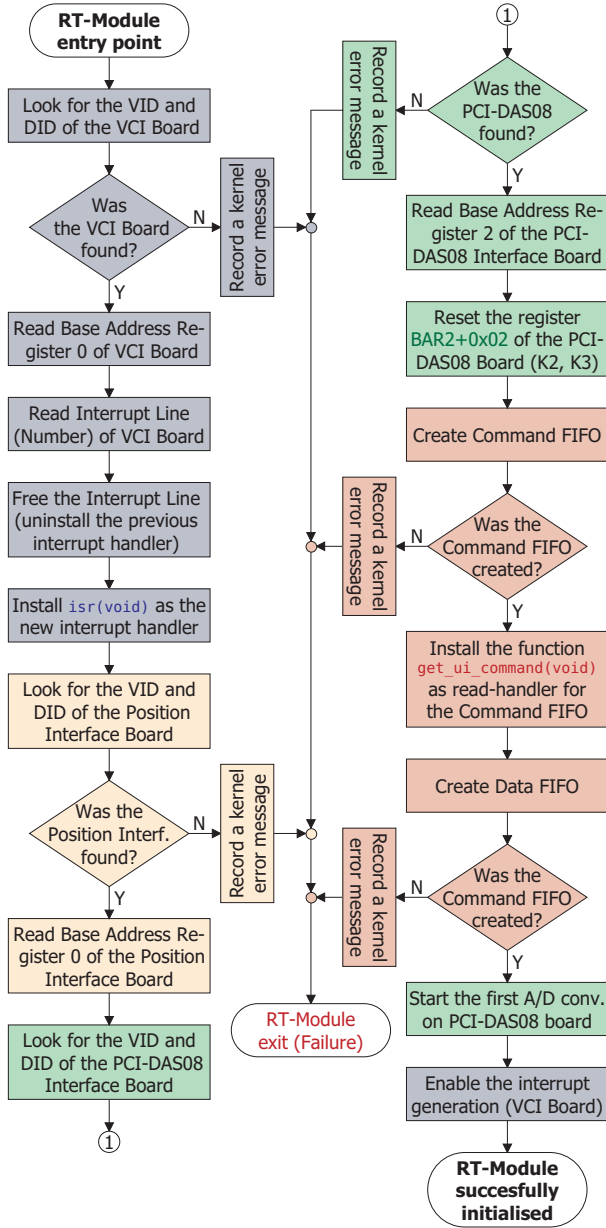
Figure 3.16: Real-Time Module initialisation

The Base Addresses of the PCI devices are not hardwired – as it is the case e.g. with the Industry Standard Architecture (ISA) Bus – but allocated by BIOS at start-up. They also have to be read from the PCI structures associated with the three PCI interface boards. Additionally, the interrupt number corresponding to the PCI slot where the Vehicle Control Interface Board is located must also be determined, in order to be able to install the Interrupt Service Routine.

When all the PCI boards were located and initialised, the module initialisation function will proceed with the creation of the two communication FIFOs. Again, if an error occurs, it will be recorded, and the function terminates with a failure code. At this step, the read-handler for the Command FIFO is also installed. The initialisation then continues by starting the first analog-to-digital conversion of the DC Link voltage on the PCI-DAS08 Interface Board, and, as a last step, the interrupt generation on the Vehicle Control Interface Board is enabled. The module is now loaded and prepared to respond to the hardware control interrupt and to commands from the User Interface.

Before the Real-Time Module is unloaded, the module clean-up function will be called (see Figure 3.17 for a block diagram of this function). The first thing to do here is to disable the control interrupt generation; then the Interrupt Service Routine can be unregistered. After that, the two FIFOs are destroyed and the relay outputs K2 and K3 are deactivated (just as a precaution measure). The module can now be removed from the kernel.

The Interrupt Service Routine calls in its turn three sub-functions (in this order):

- monitoring – Implements the Monitoring State Machine (discussed below).
- machine_control – Here is the control algorithm for the linear machine implemented.
- data_capture – If data is requested from the User Interface (see CMD_CAPTURE), then this function is responsible for writing it in the Data FIFO
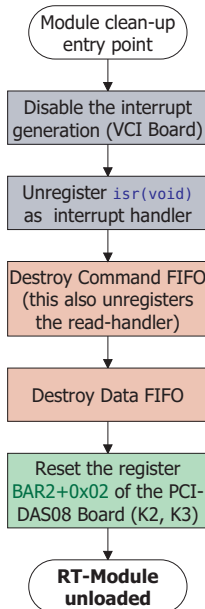
Figure 3.17: Module cleanup

Figure 3.18 shows the state diagram of the Monitoring State Machine implemented in the Real-Time Module. This state machine is updated every second (every 10,000 interrupts). The inputs and outputs of the monitoring state machine are briefly described below (see also section 3.1.2.1):

- **E1** (input, active low) – state of the Infeed Module and of the two Monitoring Modules; E1 = 0 signalises that the said modules are operational;
- **E2** (input, active low) – state of contactor K1, through which the Siemens devices are supplied; E2 = 0 signalises that K1 is closed;
- **EM** (input, active high) – state of the interface electronics' power supplies; EM = 0 signalises the failure of (at least one) power supply;
- **K2** (output, active high) – enables the "On" button T2, which in turn, when pressed, closes contactor K1;
- **K3** (output, active high) – drives an internal contactor of the Infeed Module which, when closed, supplies the DC Link

The states of the Monitoring State Machine are described in Table 3.2.
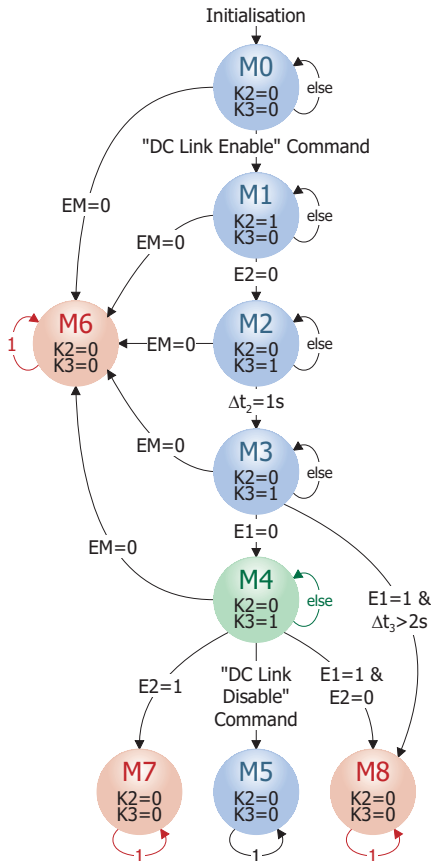
Figure 3.18: Monitoring State Machine

State	Description
M0	Initial state K2 and K3 are both open and the state machine is waiting for a DC Link power-up enable command from the User Interface
M1	Wait for T2 to be pressed The DC Link power-up was enabled from User Interface; the relay K2 is now closed, and in order to feed the Siemens' electronics the user must press the "On" button (T2) on the cabinet's door. The transition to next state occurs when input E2 is activated (E2 = 0), i.e. Siemens electronics is supplied.
M2	Close K3 In this state the feeding of the DC Link is activated (K3 = 1). E2 is already active, so K2 can now be deactivated (K2 = 0). The transition to next state occurs automatically, after 1 s.
M3	Wait for E1 Wait for the Infeed Module and the two Monitoring Modules to become operational (E1 = 0).
M4	Active state The DC Link is fed and the machine control can operate. The Power Stack Interfaces can be addressed only if the Monitoring State Machine is in this state.
M5	Normal stop The user has commanded the power-down of the DC Link from the User Interface, this being possible only after stopping the machine control algorithm. This state has no exit; in order to restart the control algorithm after disabling the feeding of the DC Link, the user must stop and restart the Real-Time Module and the User Interface.
M6	Interface Electronics' Power Supply Failure When the Monitoring State Machine is in one of the states M0 … M4 and a failure occurs in the electronics' power supply (EM = 0), K2 and K3 will be opened (thus deactivating the feeding of the DC Link) and the operation of the state machine will block in this state.
M7	DC Link Power Supply Failure The DC Link supply was stopped because E2 became inactive. Cause: the "Off" button (T1) on the cabinet's door was pressed before deactivating the DC Link power supply from the User Interface.
M8	Infeed / Monitoring Module Failure The DC Link supply was stopped because E1 became inactive, while E2 was still active (transition from M4) or, 3 seconds after the closing of K3 ($\Delta t_2 + \Delta t_3$), E1 was still not active (transition from M3)

Table 3.2: Description of the states of Monitoring State Machine

States M5 – M8 are all final states (they have no outbound transitions), with the same outputs (K2 = 0, K3 = 0). They were introduced in order to communicate to the User Interface (and further to the user) the reason why the normal execution was stopped.

A block diagram of the machine control function is given in Figure 3.19 and Figure 3.20. At the start of the control function, a position update request is sent to the Sensor to Digital Board (via the Position Interface Board). Because the communication protocol on the Sensor Bus takes about 3.5 µs (see section 2.2.4), some other tasks can be completed before the position information becomes available.

Next, the state of the analog-to-digital conversion on the PCI-DAS08 board is

checked (the conversion was started in the previous interrupt or, in the case of the first interrupt, by the module's initialisation function).

If the conversion is finished, then the value of the DC Link voltage is read. If not, a communication error flag is set, based on which the addressing of the Power Stack Interfaces will be disabled. This flag is also sent to the User Interface, which will notify the user about the encountered error.

After reading the DC Link voltage, the values of the measured currents for the two controlled stator segments are read from the Vehicle Control Interface Board. If the control
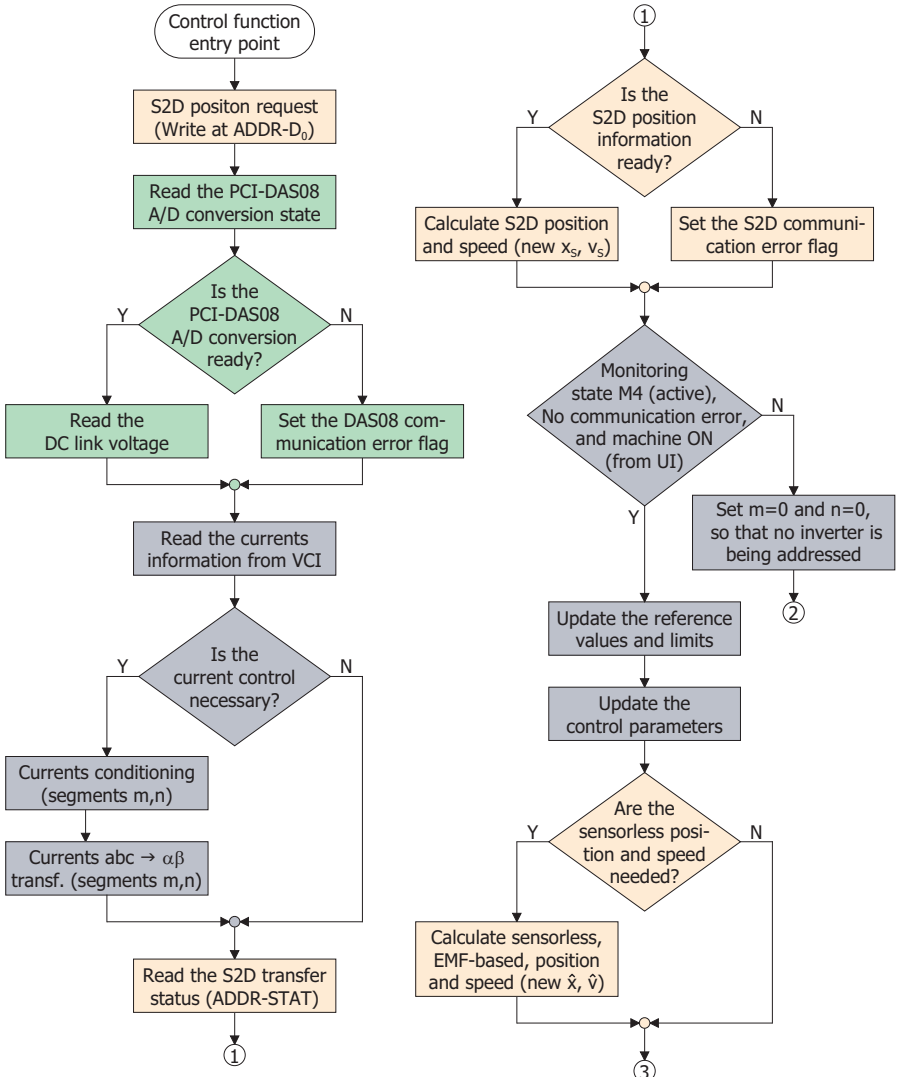
Figure 3.19: Diagram of the control function (1/2)
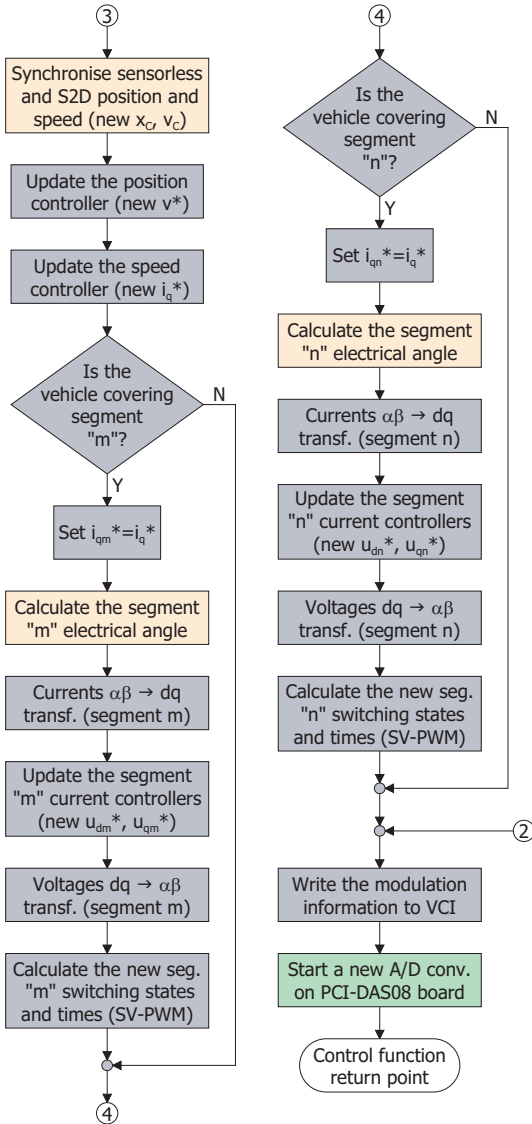
Figure 3.20: Diagram of the control function (2/2)

algorithm is enabled, then the current conditioning function (which performs the conversion from converter units to amperes, along with an offset correction) will be called; afterwards, the α–β components (two-phased, fixed statoric) of the currents are calculated.

It must be noted that the current reading and the modulation information writing from / to the Vehicle Control Interface Board are also necessary when the control algorithm is disabled, for the correct functioning of the board's firmware [45].

At this point the position information from the Sensor to Digital Board must be available. This is checked by reading the value from ADDR-STAT (see section 2.2.5). As in the case of the PCI-DAS08 board, if a communication error occurred, an error flag is raised. If the position information is available, the sensor's position and speed are calculated according to section 2.2.6.

In order to proceed with the calculation of the control algorithm, three conditions must now be simultaneously fulfilled:

- The Monitoring State Machine must be in state M4 – meaning that the DC Link is powered-up and there is no failure of the power supplies.
- There must be no communication error encountered up to this point.
- The user must have enabled the starting of the linear machine.

If these conditions are fulfilled, then the control algorithm proceeds by updating the values of the control references, limits and parameters (this can be the case e.g. when passing from a stator segment to the next one, which has different parameters). Then, it is checked whether the vehicle is in a region where sensorless control is necessary. If yes, updated values of the observed position and speed are calculated. The EMF-based sensorless control is described in detail in section 3.3.

Then, the synchronisation function, between the sensor's position and speed and the observed ones is called. The transition between sensors-equipped and sensorless regions is described in section 3.4. As result of this function the values of the position and speed to be used in control are obtained, and they will be given as feedback values to the position and speed controllers, from which the reference value of the current results.

In the following, depending on which stator segment(s) are covered by the vehicle, the corresponding current controllers are calculated.

First, the electrical angle of the vehicle, associated to each of the controlled stator segments, is calculated based on vehicle's position. Because of the gaps in the machine's winding (see sections 3.1.1 and 3.3), different values of the electrical angle can result for consecutive stator segments.

The electrical angles are then used for field orienting – the d and q components of the currents in the vehicle-attached reference system are calculated, and given as feedback values to the current controllers.

Figure 3.21 presents a vectorial block diagram of a discrete PI controller with anti-windup (general case). The parameters of the PI current controllers for all the stator segments of the linear machine are given in Table 3.3.

The error vector is defined as the difference between the reference and the measured values of the controlled quantity (at the sampling instant k):

$$\mathbf{e}[k] = \mathbf{x}^*[k] - \mathbf{x}[k].$$

Eq. 3.1

The integral component of the PI control is then calculated:

$$\mathbf{u}[k] = \mathbf{u}[k-1] + \left(\mathbf{e}[k] + \mathbf{e}[k-1]\right)\frac{T_s}{2}\frac{K_P}{T_I},$$

Eq. 3.2

where T_S is the sampling interval (100 µs), and K_P and T_I are the proportional gain and the integral time constant of the controller, respectively.

Then the output of the controller (actuating variable), is determined, first without limitation:

$$\mathbf{w}[k] = \mathbf{u}[k] + K_P \cdot \mathbf{e}[k].$$

Eq. 3.3
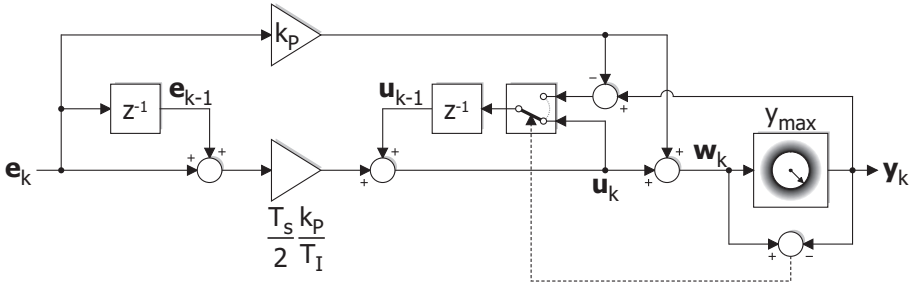
Figure 3.21: Discrete PI-Controller with anti-windup

Sec-tion	Stator Segments	Phase Resistance R_{ph} [Ω]	Phase Inductance L_{ph} [mH]	Current control parameters	
				$K_{p,i}$ [V/A]	$T_{i,i}$ [ms]
1	1,2	0.63	6.13	20.43	9.73
5	3, 4, 5, 6	0.89	9.96	33.20	11.19
4	7, 8, 9	0.89	9.96	33.20	11.19
3	10, 11	0.60	6.76	22.53	11.27
2	12 – 18	0.89	10.14	33.80	11.39

Table 3.3: Current controllers parameters

Afterwards, the actuating vector is limited to y_{max} and the output of the controller results:

$$\mathbf{y}[k] = \begin{cases} \mathbf{w}[k] \, , \text{if } \|\mathbf{w}[k]\| \le y_{max} \\ \dfrac{\mathbf{w}[k]}{\|\mathbf{w}[k]\|} \, y_{max} \, , \text{if } \|\mathbf{w}[k]\| > y_{max} \end{cases}$$

Eq. 3.4

If a limitation of the output occurred, the integral term is also adjusted (anti-windup):

$$\mathbf{u}[k] = \begin{cases} \mathbf{u}[k] \, , \text{if } \|\mathbf{w}[k]\| \le y_{max} \\ \mathbf{y}[k] - K_P \cdot \mathbf{e}[k] \, , \text{if } \|\mathbf{w}[k]\| > y_{max} \end{cases}$$

Eq. 3.5

After the reference voltage vectors of the controlled stator segments were calculated as outputs of the current controllers, they are transformed from the vehicle-attached reference system (dq) back to the stator-attached one (αβ). Then, the modulation information (switching states and times) is determined [45] and written to Vehicle Control Interface Board, from where it will be sent, via the Inverter Bus, to the corresponding Power Stack Interfaces.

As a last step, a new analog-to-digital conversion of the DC Link voltage is started on the PCI-DAS08 board, this concluding the control algorithm for a given sampling instant.

3.2 Implementation of the Optical Sensors System

In this section, the implementation of the optical sensors system presented in section 2.2 of this work, at the linear drive for material handling (see the description of the linear machine in section 3.1.1) will be discussed.

First, the mechanical mounting of the optical sensors will be presented, and then measurements concerning the quality of the sine / cosine signals, the correction of these signals and the reconstructed position will be shown.

Because on the linear drive for material handling there is no reference position sensor available, only the quality of the sine / cosine signals will be used as a measure of the quality of the resulting position signal.

Figure 3.22 shows the mechanical mounting of the optical read-heads and scale at the double-sided section of the linear machine, with rigid vehicle. The read-heads are mounted on an L-shaped profile attached to the machine's carrying / guiding tube. The measuring direction of the sensors system was chosen along z-axis (see also Figure 5.7 in Appendix 5.5) because, as specified by the machine's manufacturer, this axis presents a higher guiding stiffness.

The distance between the optical read-heads and the scale (which is mounted on the underside of the load carrying plate on the vehicle) must be kept in the 0.6 … 0.9 mm range for the adequate operation of the optical system [26]. In order to avoid an impact between the scale and the read-heads, a collision protection system was also installed for each read-head (see also Figure 5.8).

During the testing of the optical system it was noticed that, due to the mounting position of the read-heads, variations of the ambient light's intensity could lead, under
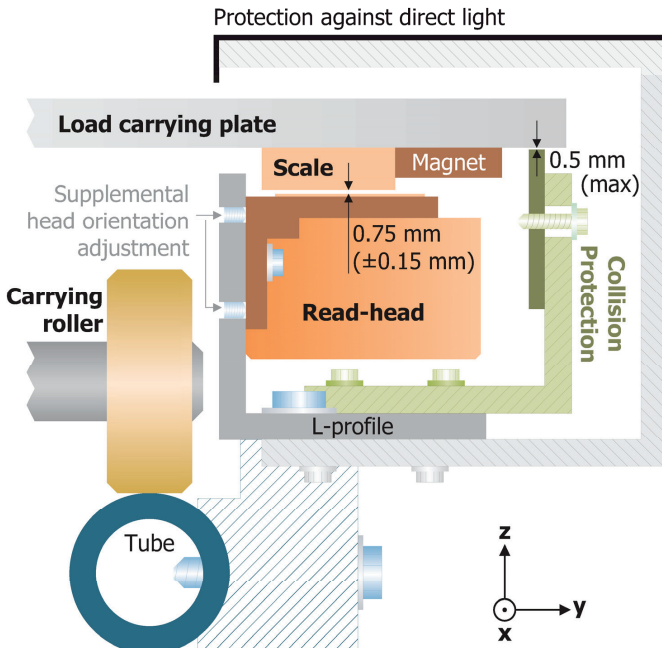
Figure 3.22: Mounting of the optical sensors system
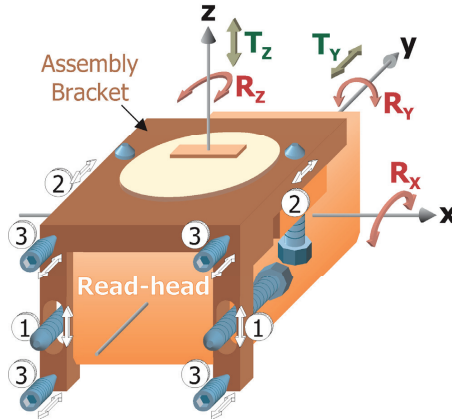
Figure 3.23: Read-head adjustments

Mounting Points	Adjustments
①	Tz – Translation along z-axis Ry – Rotation around y-axis (in the xz-plane)
②	Ty – Translation along y-axis Rz – Rotation around z-axis (in the xy-plane)
③	Ty – Translation along y-axis Rz – Rotation around z-axis (in the xy-plane) Rx – Rotation around x-axis (in the yz-plane)

Table 3.4: Possible orientation adjustments

certain conditions, to the generation of incorrect reference signals by read-heads which were not covered by the scale. To avoid this problem, a protection against direct light was also mounted above the L-profile.

In order to obtain the best possible signal quality (sine, cosine, reference), the read-heads were mounted in such a manner as to allow for small orientation adjustments in five axes, as shown in Figure 3.23. The read-heads are not mounted directly on the L-profile, but by means of assembly brackets.

The oblong holes on the assembly brackets ① allow for translational adjustments along the z-axis, as well as for rotational adjustments around the y-axis, whilst the similar oblong holes ② on the read-heads make possible translational adjustments along the y-axis and rotational adjustments around the z-axis.

To also allow for rotations around the x-axis, the supplemental set screws ③ are used. These four screws are inserted in the L-shaped profile, close to the corners of the assembly brackets' base (see also Figure 3.22). They can also be used for additional adjustments (translation along the y-axis and rotation around the z-axis).

All the possible adjustments of the read-heads' orientation at the different mounting points are summarised in Table 3.4.
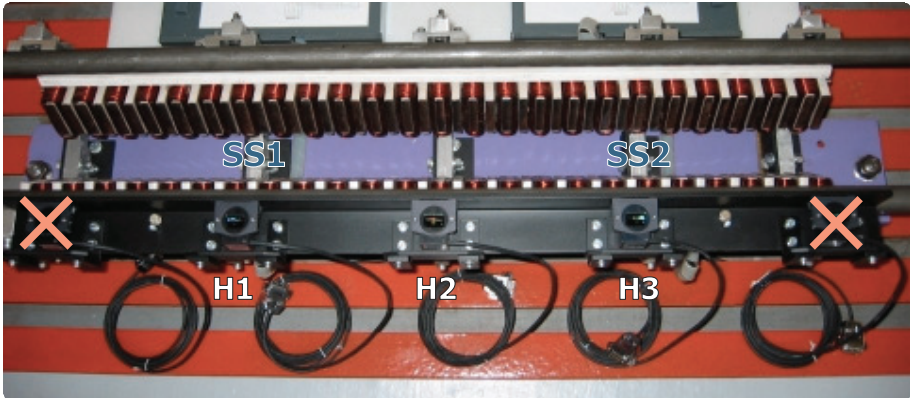
Figure 3.24: Straight section of the linear machine, with the optical sensors mounted

In Figure 3.24 the straight, double-sided section of the linear machine (stator segments 1 and 2) is depicted, with the optical read-heads mounted. Because this section is just four elementary machines (i.e. 0.96 m) long, a travelling distance of only 0.72 m is available (with the 0.24 m long vehicle remaining fully inside the section), and only three read-heads can be completely covered. Therefore, the leftmost and the rightmost read-heads were not used.

The FPGA code on the Sensor to Digital Board and the position calculation function in the control program were modified accordingly, by reducing the number of states in the Current Head State Machine and in the Reconstruction State Machine, respectively (see section 2.2.3 and section 2.2.6). It must be noted that the distance between the read-heads was also slightly reduced, to 0.2 m (5000 sine / cosine periods), so the total measuring length now amounts to 0.6 m.

This 3-read-heads configuration was also kept when the entire linear machine (all the 18 segments together with the articulated vehicle) was available. Even though now the travelling length is extended in both directions (by stator segments 3 and 18), when covering the leftmost and the rightmost read-heads (in the original 5-read-heads configuration), the articulated vehicle is already partly into the curved section of the track, and the scale is rotated with respect to these read-heads, making the sine / cosine signals unusable for position determination.

The articulated vehicle is less stable in the z-axis than the rigid one (mainly due to the spring-mounting of the outer guiding rollers). In order to increase the guiding stiffness along the measuring axis, the collision protection system shown in Figure 3.22 and in Figure 5.8 was converted into a guiding system for the load carrying plate, as sketched in Figure 3.25.

The load carrying plate now travels on the guiding rollers, the height of which was set about one millimetre higher than the normal level of the load carrying plate, assuring that the contact is always maintained.

The guiding rollers are spaced so that at every point inside the sensor's region, the load carrying plate is guided by at least two of them, thus maintaining the correct orientation of the scale with respect to the optical read-heads.

Figure 3.26 shows the underside of the load carrying plate, with the guideway for the rollers. The ends of this guideway are angled upwards, to allow smooth entering into and exiting from the guided section.
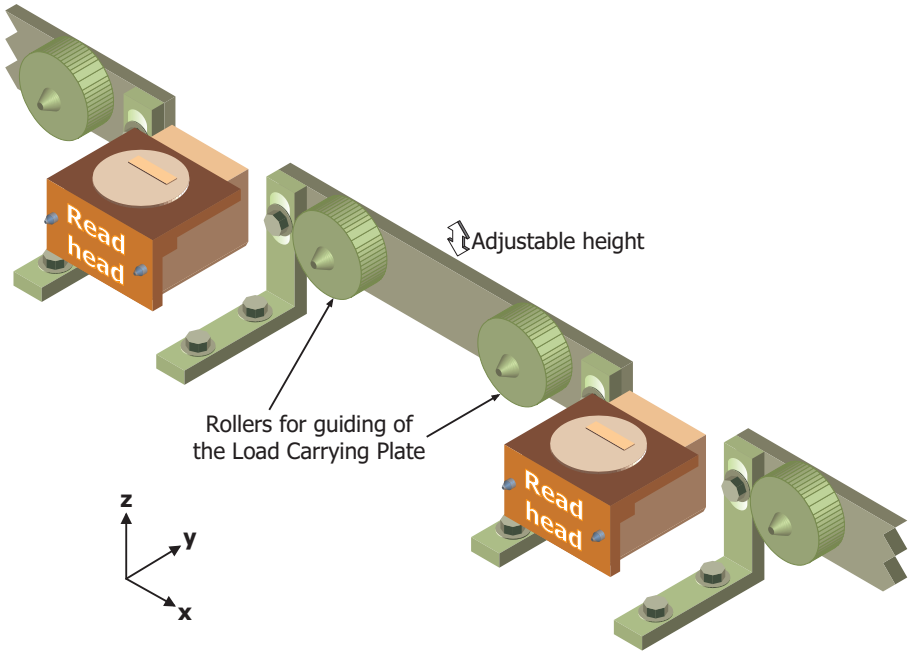
Figure 3.25: Sensors guiding system
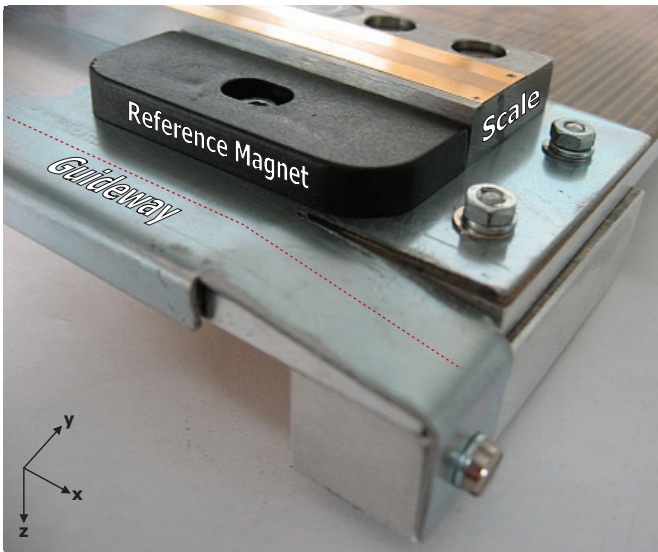
Figure 3.26: Underside of the load carrying plate

Figure 3.27 shows measurements of the sine / cosine and reference signals at different positions in the sensor's region (after the adjustment of the three read-heads). The scale was mounted on the rigid vehicle, without the guiding rollers (mounting as in Figure 3.22). The sine / cosine signals were measured at the inputs of the four analog-to-digital converters, whilst the reference signals were measured at the outputs of the two comparators (see Figure 2.7).

Ideally, the sine / cosine signals generated by the optical sensors would have an amplitude of 1 V_{pp} over the entire measuring length; the analog-to-digital converters have an input range of 5 V_{pp}, so a gain of 5 would be necessary at the Sensor to Digital board's signal conditioning stage. However, the gain was set to 4.3 (86%) so that in the case of small, local increases of the amplitudes over the nominal value, the analog-to-digital converters' outputs would not saturate. With this gain, 4.3 V_{pp} signals are expected at the inputs of the analog-to-digital converters, i.e. ±2.15 V, which corresponds to approx. ±1760 converter units. The per-unit values in the following figures (Figure 3.28 … Figure 3.30) take into consideration this reduced gain.

In Figure 3.27 (a) and (g) the relationship between the reference signals and the

a) Beginning of Read-head 1
$S_A = S_1, C_A = C_1$

b) Middle of Read-head 1
$S_A = S_1, C_A = C_1$

c) Transition Read-head 1 → Read-head 2
$S_A = S_1, C_A = C_1$
$S_B = S_2, C_B = C_2$

Figure 3.27: Measured sine/cosine signals

d)

Middle of Read-head 2
$S_B = S_2$, $C_B = C_2$

e)

Transition Read-head 2 → Read-head 3
$S_B = S_2$, $C_B = C_2$
$S_A = S_3$, $C_A = C_3$

f)

Middle of Read-head 3
$S_A = S_3$, $C_A = C_3$

g)

End of Read-head 3
$S_B = S_3$, $C_B = C_3$
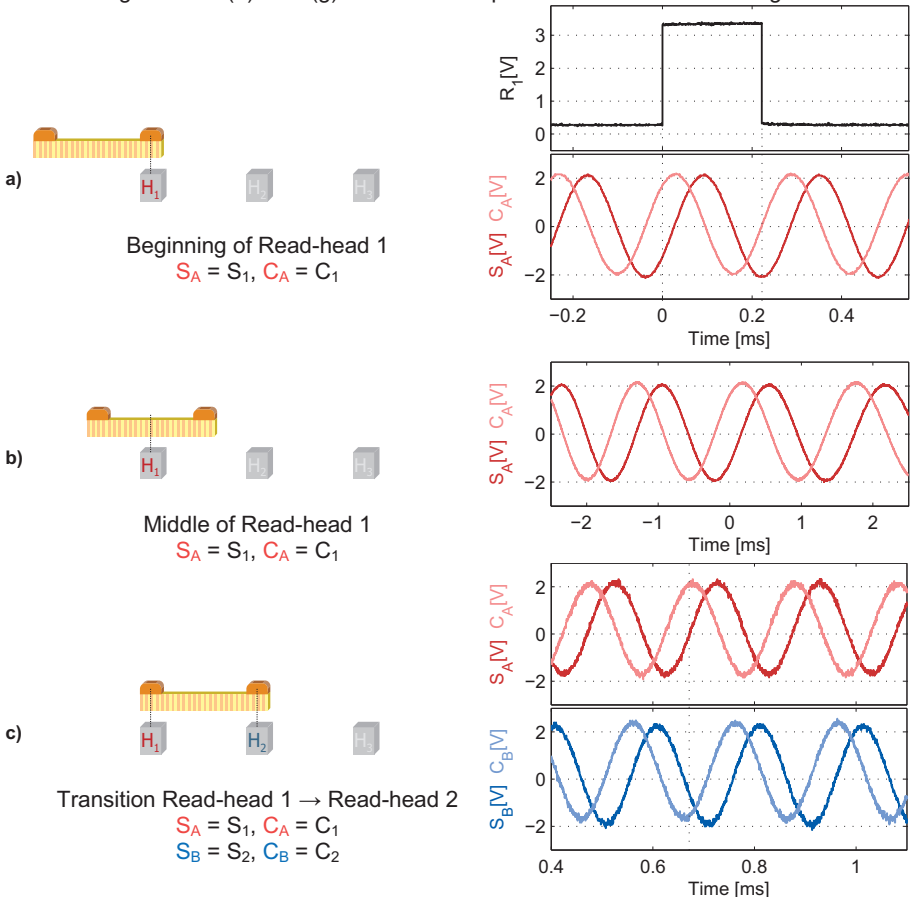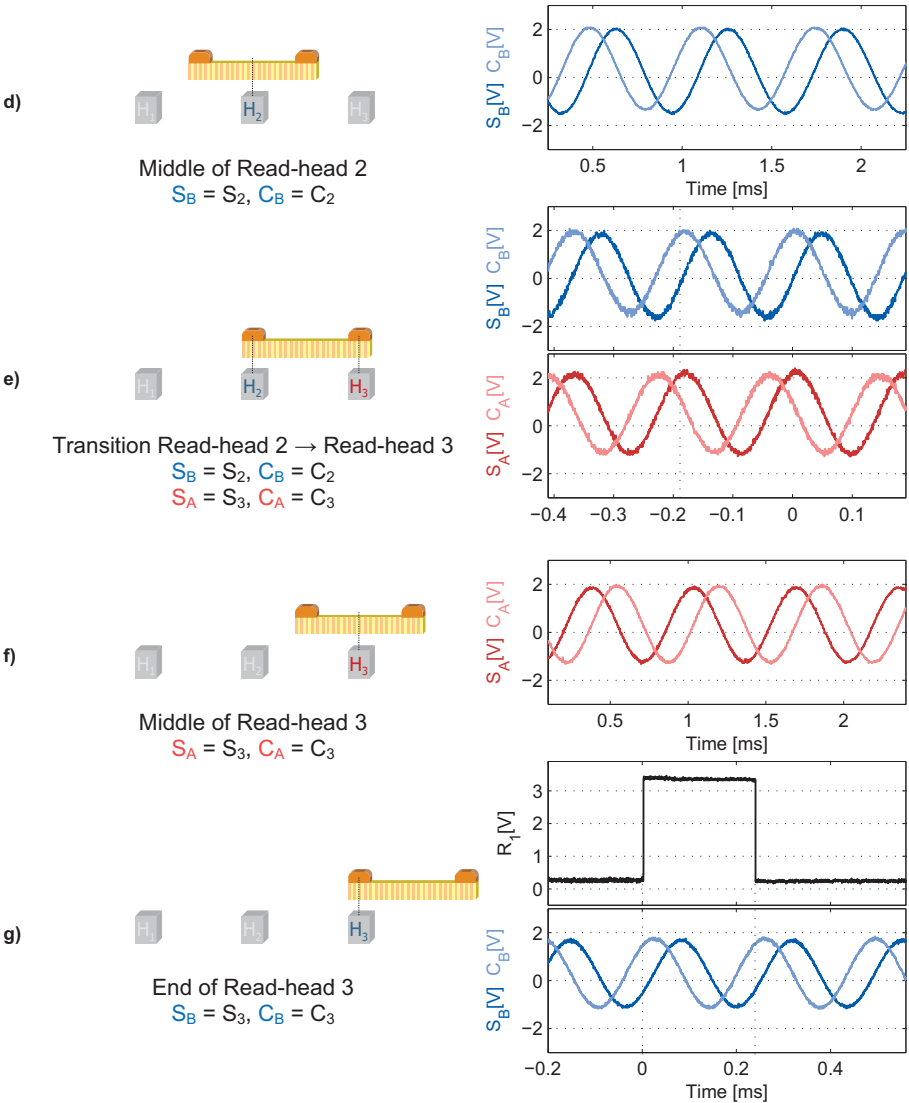
Figure 3.27: Measured sine/cosine signals (continuation)

positive zero crossings of the sine signals can be noticed – exactly one positive zero crossing of the sine signal while the corresponding reference signal is high (compare also with Figure 2.6). The phase differences between successive read-heads can be seen in Figure 3.27 (c) and (e).

A decrease of the amplitudes of the sine / cosine signals, as the vehicle travels towards the end of the sensors' region (in positive direction) can also be observed in Figure 3.27. This is also visible in the xy-representation of all the periods of the four logical read-heads from Figure 3.28 (on the left column).
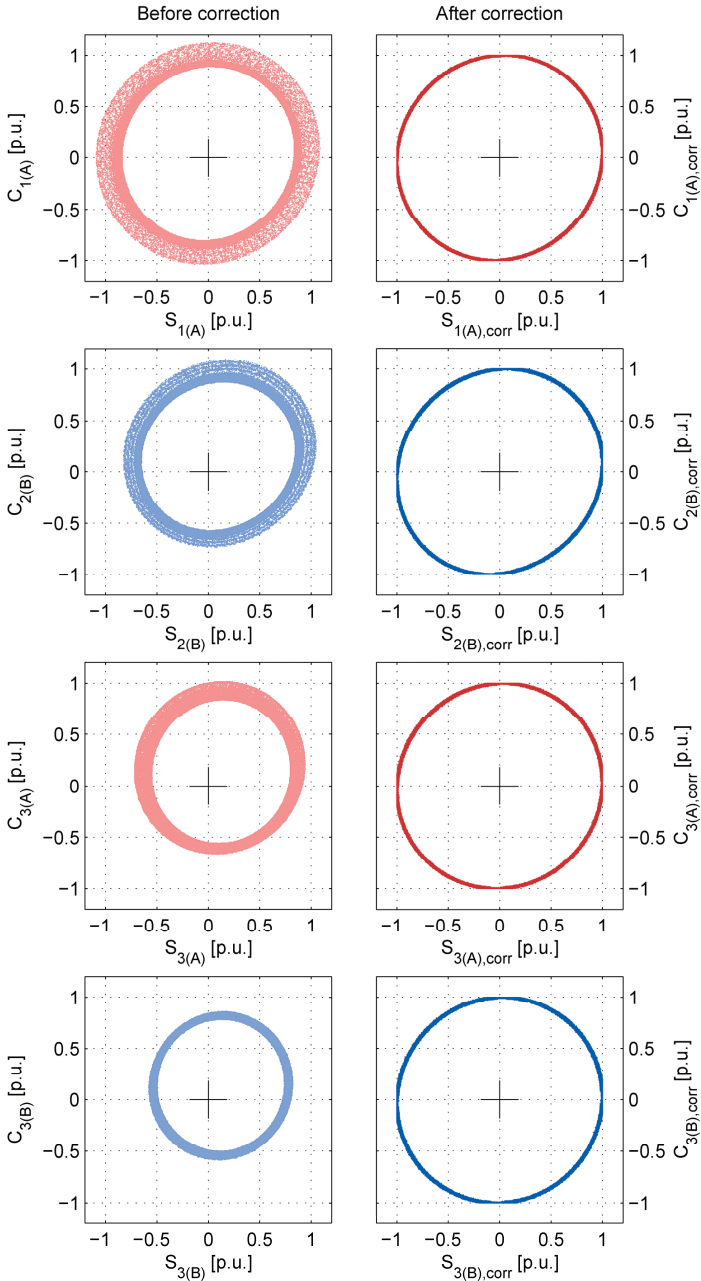
Figure 3.28: XY-Representation of the sine/cosine signals (<u>rigid</u> vehicle)
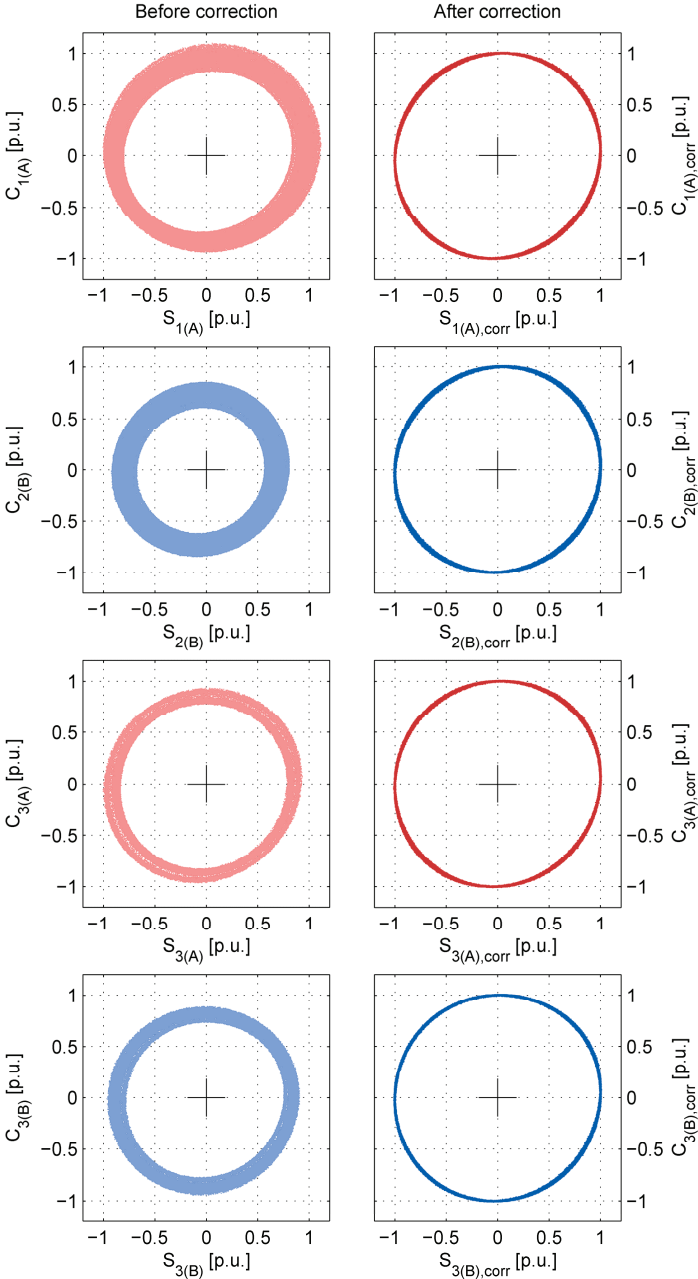
Figure 3.29: XY-Representation of the sine/cosine signals (<u>articulated</u> vehicle)

The cause of this amplitude decrease is an increase of the distance between the scale and the read-heads along the travelling length. Using the sensors guiding construction shown in Figure 3.25 and Figure 3.26 reduces this variation, as shown in the left column of Figure 3.29 – an xy-representation of the sine / cosine signals with the scale mounted on the articulated vehicle and with sensor guiding.

On the right sides of Figure 3.28 and Figure 3.29 the corrected sine / cosine signals are depicted (the individual period correction method described in section 2.2.7 was used.)

In Figure 3.30, the data acquired for the calculation of the correction tables for the articulated vehicle is shown. For this acquisition the vehicle was moved by hand, in positive direction, throughout the entire measuring length.
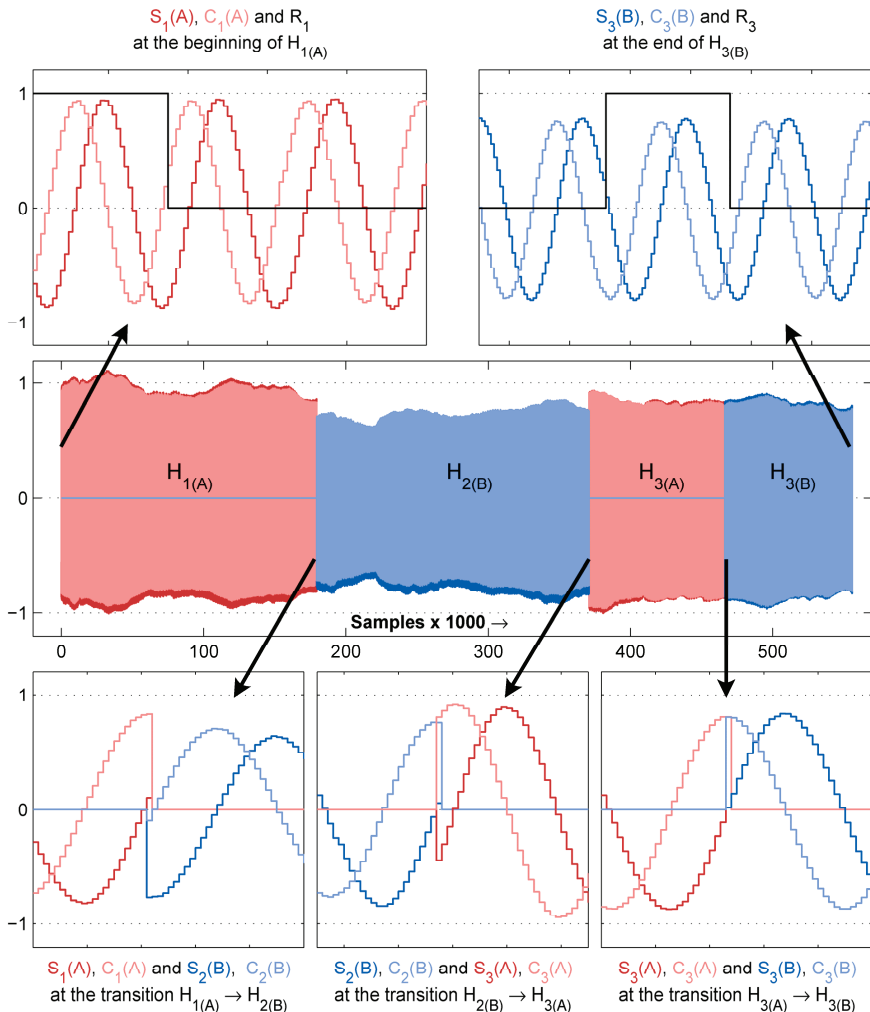
Figure 3.30: Correction tables data (articulated vehicle)
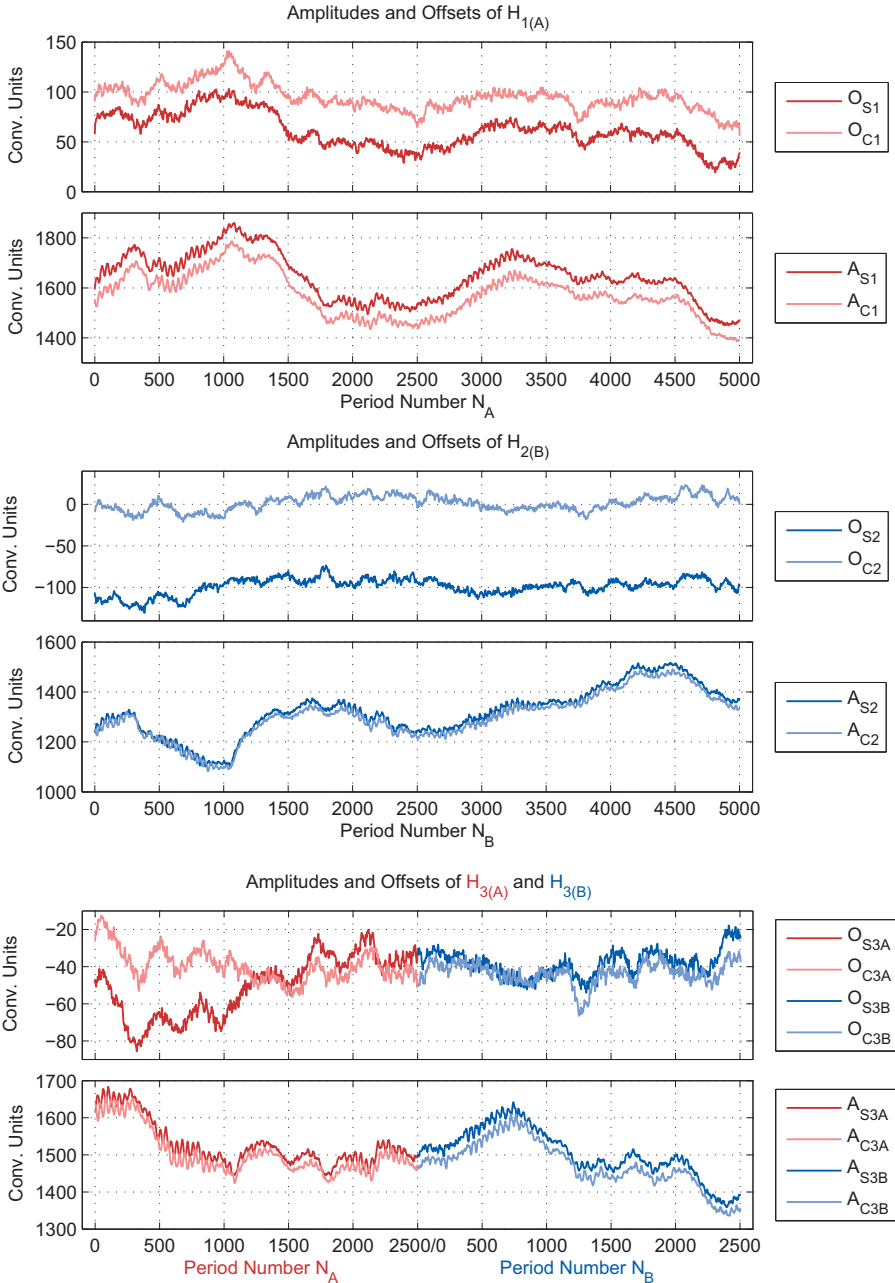
Figure 3.31: Correction tables (articulated vehicle)

Only the sine, cosine and reference signals were received, in burst mode, from the Sensor to Digital Board. The period numbers for each logical read-head are determined offline, based on the positive zero-crossings of the corresponding sine signal and then, for each period of each read-head the amplitudes and offsets of the sine and of the cosine signals are calculated. The resulting correction tables can be seen in Figure 3.31. This data is saved in files to be used, online, by the control program.

Figure 3.32 shows an online measurement of the reconstructed position signal – compare with Figure 2.17 in section 2.2.6 (Position Reconstruction in C Code). The articulated vehicle was moved again by hand, over the entire measuring length, in both positive and negative moving directions (with the control program running, but with no inverter being addressed). Entering into the sensors' region and exiting from the sensors' region (in both directions) are also covered by the measurement.

In the lower part of the figure the corresponding speed signal is displayed. The speed was calculated (online) as the numerical derivative of the position, over the 100 µs control interval and without any filtering. A zoom of the speed signal, at the transition between read-heads 1 and 2 can be seen in Figure 3.33. The high-frequency variations in the speed are derivation noise, which is generated by variations of the time intervals between successive position readings, as explained below.

As described in section 2.2.3, the Sensor to Digital Board acquires new position
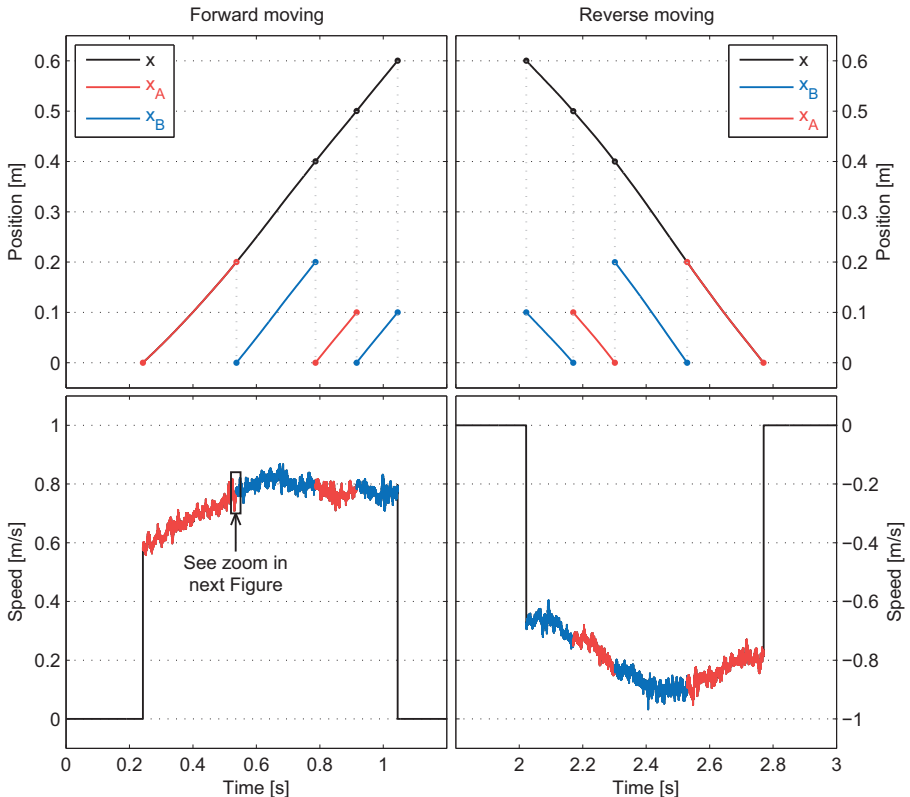
Figure 3.32: Measurement of the reconstructed position and of the corresponding speed
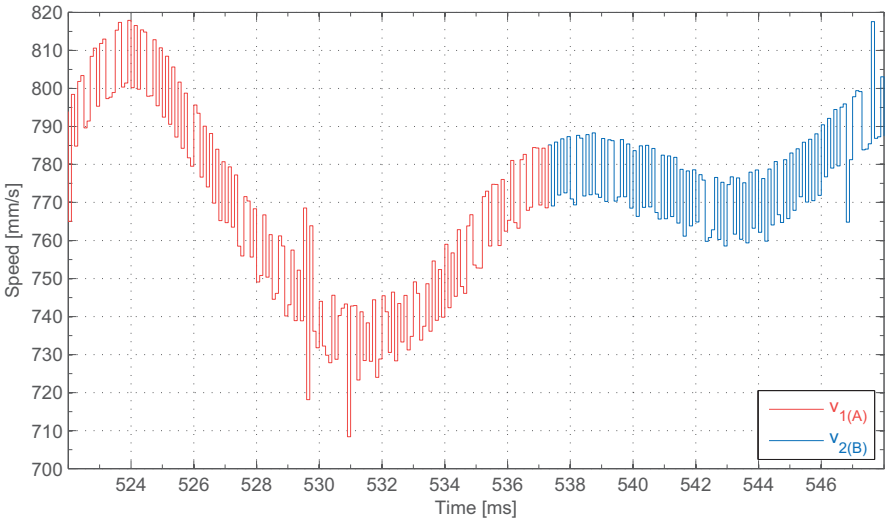
Figure 3.33: Zoom of the measured speed (at the transition between heads 1 and 2)

information every 2 µs. These 2 µs time intervals are generated with the help of the board's 80 MHz oscillator. On the other hand, the position requests are generated by the control program, at the beginning of the interrupt service routine, which is triggered by a hardware interrupt request coming from the Vehicle Control Interface Board. The hardware interrupt timing is based on the Control PC's PCI clock. Because there is no synchronisation between the two clock sources, a clock drift can occur, so the time elapsed between the sampling of two successively read positions can have a variation of ±2 µs with respect to the 100 µs control cycle.

This can be verified by analysing Figure 3.33: a ±2 µs variation of the time between the sampling instants, at a mean travelling speed of approx. 750 mm/s (as in Figure 3.33), produces a variation of the sampled position of approx. ±1.5 µm, which, over the 100 µs control cycle introduces a noise of ca. ±15 mm/s in the speed signal. The interrupt jitter (see Figure 3.14) also contributes to the derivation noise, by additionally increasing the variation of the time between the sampling instants (see the "spikes" in Figure 3.33).

Because of its high frequency, the derivation noise could be almost completely eliminated by a first order low pass filter with a small time constant (300 … 500 µs), which filter would introduce only a small phase delay in the speed control loop. However, using such a filter has disadvantages at entering in the sensor's region: Without filtering, the first speed value is available with only 100 µs delay with respect with the first position value (as soon as the first two position values are acquired); when using the low-pass filter, it can take up to ca. 2 ms until the filtered speed reaches steady state – which translate into ca. 20 mm at the beginning of the sensors' region where no speed signal is available (considering the maximum travelling speed of 10 m/s).

In order to eliminate the derivation noise without the use of any filter, the Sensor to Digital Board's firmware must be modified by adding a counter, which keeps track of the number of 2 µs sampling intervals elapsed between two successive position readings coming from the control program. Then, the Sensor Bus transfer protocol must also be changed, to send the value of this counter together with the other position information back to control, to be used in the calculation of the numerical derivative of the position.

3.3 EMF-Based, Sensorless Speed Control

In the present implementation of the control algorithm for our application, it is assumed that a vehicle will start / stop only inside a processing station, using feedback from the position sensors.

This makes the use of the EMF-based estimation method adequate for the transport sections (outside the processing stations), even though this method does not work at zero speed. Expanding the functionality of this linear drive system (e.g. by implementing vehicle buffering before entering a processing station) requires future work on sensorless position acquisition methods using signal injection.

For the sensorless, EMF-based determination of position and speed, a structure consisting of three observers will be used:

- Two EMF observers, one for each of the controlled stator segments.
- One mechanical observer, which uses the two observed EMF-vectors in order to estimate the position and speed.

The integration of this three-observers structure in the algorithm of one vehicle controller is depicted in Figure 3.34.

The EMF observers, the mechanical observer, as well as the implementation of the three-observers structure at the linear drive for material handling, including experimental results, will be discussed in the following subsections.
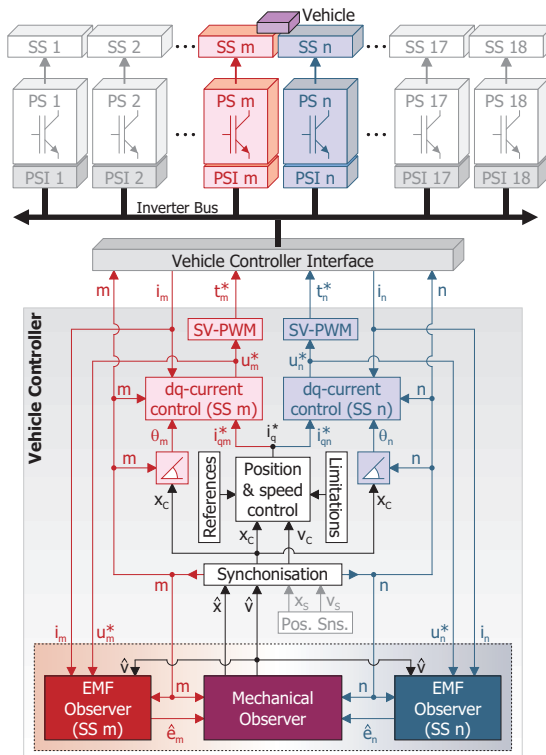
Figure 3.34: The three-observers structure, integrated in the control diagram

3.3.1 EMF Observers

Throughout this subsection, all quantities involved will refer to the same stator segment. For simplicity, any subscripts referring to the considered segment's number will be omitted.

A sinusoidal, fundamental frequency model will be used for the electrical subsystem of each segment of the linear machine. The voltage equation of the considered segment, in the stationary (α–β) reference frame, can be written:

$$\begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} = R \cdot \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + \frac{d}{dt}\begin{bmatrix} \Psi_\alpha \\ \Psi_\beta \end{bmatrix},$$

Eq. 3.6

where u_α, u_β, i_α, i_β, Ψ_α, Ψ_β are the components of the segment's voltage, current and flux linkage vectors', respectively. R is the phase resistance of the segment.

The flux linkage vector can be further expressed as the sum of the current dependent flux and the flux linkage generated by the permanent magnets:

$$\begin{bmatrix} \Psi_\alpha \\ \Psi_\beta \end{bmatrix} = \begin{bmatrix} \Psi_{L,\alpha} \\ \Psi_{L,\beta} \end{bmatrix} + \begin{bmatrix} \Psi_{PM,\alpha} \\ \Psi_{PM,\beta} \end{bmatrix} = L \cdot \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + \begin{bmatrix} \Psi_{PM,\alpha} \\ \Psi_{PM,\beta} \end{bmatrix},$$

Eq. 3.7

with **L** being the inductance matrix:

$$L = \begin{bmatrix} L_0 + L_1 \cos 2\theta' & -L_1 \sin 2\theta' + L_2 \\ L_1 \sin 2\theta' + L_2 & L_0 + L_1 \cos 2\theta' \end{bmatrix}$$

Eq. 3.8

In the above equation L_0, L_1 and L_2 represent the mean value of the self-inductance, the amplitude of the inductance variation, and the mean value of the mutual inductance, respectively. This last term (L_2) does not appear in rotating motors; it is specific to linear machines, being caused by end effects [56], [57]. θ' is the electrical angle of the vehicle within the considered pole pitch and is related to the local position x' by:

$$\theta' = \frac{\pi}{\tau_P} \cdot x',$$

Eq. 3.9

τ_P being the machine's pole pitch (24 mm).

It was experimentally determined that, due to the relatively large air gap, the linear machine presents only small saliencies – the amplitude L_1 of the inductance variation, as well as the end-effects-determined inductance L_2, are considerably smaller than the mean value L_0 of the self-inductance (ca. 1 … 2%). Therefore, L_1 and L_2 will be subsequently neglected, and the inductance matrix **L** will be approximated as:

$$L \approx \begin{bmatrix} L_0 & 0 \\ 0 & L_0 \end{bmatrix}.$$

Eq. 3.10

The following expression will be considered for the permanent magnets flux linkage:

$$\begin{bmatrix} \Psi_{PM,\alpha} \\ \Psi_{PM,\beta} \end{bmatrix} = \Psi_{PM,0} \cdot \begin{bmatrix} \cos \theta' \\ \sin \theta' \end{bmatrix}$$

Eq. 3.11

Using the above equation, the vector of the voltage induced in one stator segment by one vehicle can be expressed as:

$$\begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix} = \frac{d}{dt}\begin{bmatrix} \Psi_{PM,\alpha} \\ \Psi_{PM,\beta} \end{bmatrix} = \Psi_{PM,0} \cdot \begin{bmatrix} -\sin \theta' \\ \cos \theta' \end{bmatrix} \cdot \frac{d\theta'}{dt}$$

Eq. 3.12

By introducing in Eq. 3.12 the electrical angular speed ω, the speed v of the vehicle and the EMF constant K_E, respectively:

$$\omega = \frac{d\theta'}{dt}, \quad v = \frac{\tau_P}{\pi}\omega, \quad K_E = \frac{\pi}{\tau_P}\Psi_{PM,0},$$

Eq. 3.13

the EMF vector can be written:

$$\begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix} = K_E \cdot \begin{bmatrix} -\sin\theta' \\ \cos\theta' \end{bmatrix} \cdot v$$

Eq. 3.14

The EMF constant K_E (the ratio between the modulus of the EMF vector and the speed) is determined experimentally for each segment of the machine. This constant depends (through $\Psi_{PM,0}$) on the air-gap δ, which has variations, especially in the curved sections of the track (see Figure 3.5). For each segment, the mean value of K_E will be considered. The procedure used to determine the EMF constant, as well as the resulting values for all 18 segments of the linear machine, are given in Appendix 5.6.

Assuming that the speed v varies much slower than the electrical quantities, the derivative of the EMF vector can be expressed as:

$$\frac{d}{dt}\begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix} = K_E \cdot \begin{bmatrix} -\cos\theta' \\ -\sin\theta' \end{bmatrix} \cdot \frac{\pi}{\tau_P} \cdot v^2$$

Eq. 3.15

Introducing Eq. 3.14 in the above expression, the derivative of the EMF vector yields:

$$\frac{d}{dt}\begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix} = \begin{bmatrix} 0 & -\dfrac{v\pi}{\tau_P} \\ \dfrac{v\pi}{\tau_P} & 0 \end{bmatrix} \cdot \begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix}$$

Eq. 3.16

Eq. 3.6 can be rewritten as follows:

$$\begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} = R \cdot \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} + \frac{d}{dt}\begin{bmatrix} \Psi_{L,\alpha} \\ \Psi_{L,\beta} \end{bmatrix} + \begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix}$$

Eq. 3.17

Based on Eq. 3.16 and Eq. 3.17, the electrical subsystem of one segment of the linear machine can be modelled by the following state equations [62]:

$$\frac{d}{dt}\begin{bmatrix} \Psi_{L,\alpha} \\ \Psi_{L,\beta} \\ e_\alpha \\ e_\beta \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & -\dfrac{v\pi}{\tau_P} \\ 0 & 0 & \dfrac{v\pi}{\tau_P} & 0 \end{bmatrix}\begin{bmatrix} \Psi_{L,\alpha} \\ \Psi_{L,\beta} \\ e_\alpha \\ e_\beta \end{bmatrix} + \begin{bmatrix} 1 & 0 & -R & 0 \\ 0 & 1 & 0 & -R \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} u_\alpha \\ u_\beta \\ i_\alpha \\ i_\beta \end{bmatrix}$$

Eq. 3.18

The EMF observer will be based on the above model. As error function (correction term) of the observer, the difference between the measured and the estimated current dependent flux vectors is used:

$$\begin{bmatrix} \varepsilon_{\Psi,\alpha} \\ \varepsilon_{\Psi,\beta} \end{bmatrix} = \begin{bmatrix} \Psi_{L,\alpha} \\ \Psi_{L,\beta} \end{bmatrix} - \begin{bmatrix} \hat{\Psi}_{L,\alpha} \\ \hat{\Psi}_{L,\beta} \end{bmatrix} = L \cdot \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} - \begin{bmatrix} \hat{\Psi}_{L,\alpha} \\ \hat{\Psi}_{L,\beta} \end{bmatrix}$$

Eq. 3.19

Using this error function and considering the model of the permanent magnet synchronous machine, the EMF observer is implemented by the following state equations:

$$\frac{d}{dt}\begin{bmatrix}\hat{\Psi}_{L,\alpha}\\\hat{\Psi}_{L,\beta}\\\hat{e}_{\alpha}\\\hat{e}_{\beta}\end{bmatrix}=\begin{bmatrix}0&0&-1&0\\0&0&0&-1\\0&0&0&-\dfrac{\hat{v}\pi}{\tau_P}\\0&0&\dfrac{\hat{v}\pi}{\tau_P}&0\end{bmatrix}\begin{bmatrix}\hat{\Psi}_{L,\alpha}\\\hat{\Psi}_{L,\beta}\\\hat{e}_{\alpha}\\\hat{e}_{\beta}\end{bmatrix}+\begin{bmatrix}1&0&-R&0\\0&1&0&-R\\0&0&0&0\\0&0&0&0\end{bmatrix}\begin{bmatrix}u_{\alpha}^*\\u_{\beta}^*\\i_{\alpha}\\i_{\beta}\end{bmatrix}+\begin{bmatrix}G_{\Psi}&0\\0&G_{\Psi}\\G_e&0\\0&G_e\end{bmatrix}\begin{bmatrix}\varepsilon_{\Psi,\alpha}\\\varepsilon_{\Psi,\beta}\end{bmatrix}\qquad\textbf{Eq. 3.20}$$

$\hat{\Psi}_{L,\alpha}$, $\hat{\Psi}_{L,\beta}$, $\hat{e}_{\alpha}$, and $\hat{e}_{\beta}$ denote the estimated components of the current dependent flux and of the EMF vectors, respectively, while $\hat{v}$ is the estimated speed. Because the measured values of the machine's phase voltages are not available, their reference values, u_{α}^* and u_{β}^*, will be used instead. G_{Ψ} and G_e are the gains of the EMF observer. A block diagram representation of the EMF observer is given in Figure 3.35.

Because of the speed-dependent cross coupling of the observed EMF components, the dynamics of the EMF observer cannot be analysed independently of the mechanical observer (which calculates the estimated speed $\hat{v}$ based on the estimated EMF), unless it is assumed that the estimated speed follows exactly the real speed ($\hat{v} = v$).

It is however possible to eliminate the speed-dependent cross coupling from the EMF observer, which will allow the independent parameterisation of the EMF and of the mechanical observers.
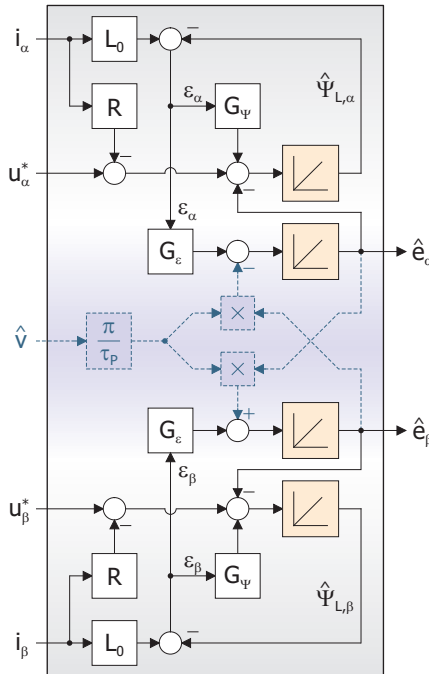
Figure 3.35: Block diagram of an EMF observer

By assuming:

$$\frac{d}{dt}\begin{bmatrix} \hat{e}_\alpha \\ \hat{e}_\beta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

Eq. 3.21

the resulting state equations of the EMF observer will be written as:

$$\frac{d}{dt}\begin{bmatrix} \hat{\Psi}_{L,\alpha} \\ \hat{\Psi}_{L,\beta} \\ \hat{e}_\alpha \\ \hat{e}_\beta \end{bmatrix} = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} \hat{\Psi}_{L,\alpha} \\ \hat{\Psi}_{L,\beta} \\ \hat{e}_\alpha \\ \hat{e}_\beta \end{bmatrix} + \begin{bmatrix} 1 & 0 & -R & 0 \\ 0 & 1 & 0 & -R \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} u_\alpha^* \\ u_\beta^* \\ i_\alpha \\ i_\beta \end{bmatrix} + \begin{bmatrix} G_\Psi & 0 \\ 0 & G_\Psi \\ G_e & 0 \\ 0 & G_e \end{bmatrix}\begin{bmatrix} \varepsilon_{\Psi,\alpha} \\ \varepsilon_{\Psi,\beta} \end{bmatrix}$$

Eq. 3.22

.

Eq. 3.22 describes a so-called "disturbance observer" [58], which assumes that the EMF term is a disturbance in the voltage equation (see Eq. 3.17). The consequences of this assumption on the estimation error will be analysed later on.

Introducing the estimation error vector:

$$\begin{bmatrix} \tilde{\Psi}_{L,\alpha} \\ \tilde{\Psi}_{L,\beta} \\ \tilde{e}_\alpha \\ \tilde{e}_\beta \end{bmatrix} = \begin{bmatrix} \Psi_{L,\alpha} \\ \Psi_{L,\beta} \\ e_\alpha \\ e_\beta \end{bmatrix} - \begin{bmatrix} \hat{\Psi}_{L,\alpha} \\ \hat{\Psi}_{L,\beta} \\ \hat{e}_\alpha \\ \hat{e}_\beta \end{bmatrix},$$

Eq. 3.23

and considering the EMF observer described by Eq. 3.22 and the machine model from Eq. 3.18, the dynamics of the estimation error will be described by the following state equations:

$$\frac{d}{dt}\begin{bmatrix} \tilde{\Psi}_{L,\alpha} \\ \tilde{\Psi}_{L,\beta} \\ \tilde{e}_\alpha \\ \tilde{e}_\beta \end{bmatrix} = \underbrace{\begin{bmatrix} -G_\Psi & 0 & -1 & 0 \\ 0 & -G_\Psi & 0 & -1 \\ -G_e & 0 & 0 & 0 \\ 0 & -G_e & 0 & 0 \end{bmatrix}}_{\mathbf{A}}\begin{bmatrix} \tilde{\Psi}_{L,\alpha} \\ \tilde{\Psi}_{L,\beta} \\ \tilde{e}_\alpha \\ \tilde{e}_\beta \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & -\frac{v\pi}{\tau_P} \\ \frac{v\pi}{\tau_P} & 0 \end{bmatrix}}_{\mathbf{B}}\begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix}$$

,

Eq. 3.24

$$\begin{bmatrix} \tilde{e}_\alpha \\ \tilde{e}_\beta \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{C}}\begin{bmatrix} \tilde{\Psi}_{L,\alpha} \\ \tilde{\Psi}_{L,\beta} \\ \tilde{e}_\alpha \\ \tilde{e}_\beta \end{bmatrix}$$

with **A** being the state matrix, and **B** and **C** – the input and output matrixes, respectively.

It must be noted that the same estimation error dynamics (the same state matrix **A**) is obtained, if the EMF observer described by Eq. 3.20 is used, under the assumption $\hat{v} = v$. In order to determine the gains G_Ψ and G_e of the observer, the characteristic polynomial

$$\det\left(s\mathbf{I}_4 - \mathbf{A}\right) = 0$$

Eq. 3.25

will be used, i.e.:

$$s^4 + \left(2G_\Psi\right)s^3 + \left(G_\Psi^2 - 2G_e\right)s^2 + \left(-2G_\Psi G_e\right)s + G_e^2 = 0.$$

Eq. 3.26

By matching the coefficients from Eq. 3.26 with the ones of a polynomial with two negative double roots

$$\left(s-p_1\right)^2\left(s-p_2\right)^2 = 0, \quad p_1, p_2 \in \mathbb{R}, \quad p_1 < 0, p_2 < 0, \tag{Eq. 3.27}$$

the gains of the observer can be written as:

$$\begin{cases} G_\Psi = -\left(p_1 + p_2\right) > 0 \\ G_e = -p_1 \cdot p_2 < 0 \end{cases} \tag{Eq. 3.28}$$

The transfer function associated to the state representation from Eq. 3.22, having the EMF vector as input and the EMF estimation error as output:

$$\begin{bmatrix} \tilde{e}_\alpha \\ \tilde{e}_\beta \end{bmatrix} = \mathbf{F}(s) \begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix}, \tag{Eq. 3.29}$$

is given by:

$$\mathbf{F}(s) = \mathbf{C}\left(s\mathbf{I}_4 - \mathbf{A}\right)^{-1}\mathbf{B}, \tag{Eq. 3.30}$$

which is represented by the following expression:

$$\mathbf{F}(s) = \begin{bmatrix} 0 & -\dfrac{v\pi}{\tau_P} \cdot \dfrac{s+G_\Psi}{s^2+G_\Psi s - G_e} \\ \dfrac{v\pi}{\tau_P} \cdot \dfrac{s+G_\Psi}{s^2+G_\Psi s - G_e} & 0 \end{bmatrix}. \tag{Eq. 3.31}$$

In the previous equations s represents the Laplace operator. From Eq. 3.31, it can be demonstrated ([58], [59]) that the error in the estimated electrical angle θ_{err}, introduced by the assumption from Eq. 3.21, is bounded by:

$$\theta_{err} \le atan\left(|v| \cdot \frac{\pi}{\tau_P} \cdot \frac{G_\Psi}{-G_e}\right) \tag{Eq. 3.32}$$

For a given maximal allowable electrical angle error θ_{err}^*, the EMF observer's gain ratio Γ can be calculated:

$$\Gamma = \frac{G_\Psi}{-G_e} = \frac{\tau_P \cdot \tan\theta_{err}^*}{|v| \cdot \pi}, \quad \Gamma > 0 \tag{Eq. 3.33}$$

Combining this expression of the gain ratio with the expression obtained for the EMF observer's gains (Eq. 2.28), the following relation between the observer's poles results:

$$p_2 = -\frac{1}{\Gamma + 1/p_1}. \tag{Eq. 3.34}$$

In order to have a stabile estimator, the following relation must also be fulfilled:

$$p_1 < -\Gamma^{-1}. \tag{Eq. 3.35}$$

The gains of the EMF observer can be now expressed as:

$$\begin{cases} G_\Psi = -\dfrac{\Gamma \cdot p_1^2}{\Gamma \cdot p_1 + 1} \\ G_e = -\Gamma \cdot G_\Psi \end{cases} \tag{Eq. 3.36}$$

The pole p_1 can be chosen in order to obtain the desired dynamics for the EMF observer, while the gain ratio Γ will ensure that the orientation error will not exceed the maximal allowable value θ^*_{err}.

For example, if a maximal orientation error $\theta^*_{err} = 25°$ is required (this corresponds to a reduction of less than 10% in the electrical force), at a maximal speed of 10 m/s, then the resulting gain ratio yields:

$$\Gamma\left(\theta^*_{err}\right) = \frac{24 \cdot 10^{-3}[m] \cdot \tan\left(25°\right)}{10[m/s] \cdot \pi} = 356.233 \cdot 10^{-6}[s/rad] \qquad \text{Eq. 3.37}$$

Choosing, e.g.

$$p_1 = -5000 \, [rad/s], \qquad \text{Eq. 3.38}$$

satisfies the condition

$$p_1 < -\Gamma^{-1} = -2807.153 \, [rad/s], \qquad \text{Eq. 3.39}$$

and the second pole of the EMF observer results:

$$p_2 = -\frac{1}{\Gamma + 1/p_1} \cong -6400 \, [rad/s] \qquad \text{Eq. 3.40}$$

The resulting observer's gains are:

$$G_\Psi = 11.4 \cdot 10^3 \, [rad/s], \quad G_e = 32 \cdot 10^6 \, [(rad/s)^2]. \qquad \text{Eq. 3.41}$$

3.3.2 Mechanical Observer

Several contributions can be found in literature (e.g. [60], [61]), in which the components of the estimated EMF vector are used directly for the calculation of the estimated position and speed:

$$\hat{x} = \text{atan}\left(\frac{\hat{e}_\alpha}{\hat{e}_\beta}\right), \qquad \hat{v} = \frac{|\hat{\mathbf{e}}|}{K_E} = \frac{\sqrt{\hat{e}_\alpha^2 + \hat{e}_\beta^2}}{K_E} \qquad \text{Eq. 3.42}$$

This approach is relatively simple and can yield good results, e.g. for rotating PM synchronous machines, but it cannot be used at our linear drive, because:

- Due to the gaps in the machine's winding at the transition between segments, there will be a reduction in the modulus of the sum of the two EMF vectors, even at constant speed and constant air-gap δ.
- There are variations of the air-gap δ, especially in the curved sections of the track.

In order to overcome these problems, a mechanical observer will be used, which observer only uses the phase information contained in the estimated EMF vector(s) for determining the position and speed. The following 3rd order model will be used to describe our linear machine's mechanical subsystem:

$$\begin{cases} \dfrac{dF_L}{dt} = 0 \\ \dfrac{dv}{dt} = \dfrac{F_E}{M_v} - \dfrac{F_L}{M_v} - \dfrac{B_v}{M_v}v \\ \dfrac{dx}{dt} = v \end{cases} \qquad \text{Eq. 3.43}$$

In Eq. 3.43, F_E and F_L represent the electrical force and the load force, respectively; v is the vehicle's speed and x represents its (absolute) position. M_v is the mass of the vehicle and B_v is the friction coefficient. The load force F_L is assumed to be constant.

When the vehicle is completely inside one stator segment, the electrical force F_E can be expressed as:

$$F_E = 3/2 \cdot K_E \cdot i_q,$$

<div align="right">Eq. 3.44</div>

where K_E is the mean EMF constant of the considered segment, and i_q is its quadrature (force-producing) current.

If the vehicle's entering and leaving of the segment is also considered, then the current-force coefficient $K_F(x)$ of that segment will be defined as follows:

$$K_F(x) = \begin{cases} 0, & x \le x_B - \Delta x_V \\[2mm] 3/2 \cdot K_E \cdot \dfrac{x - (x_B - \Delta x_V)}{\Delta x_V}, & x_B - \Delta x_V < x \le x_B \\[2mm] 3/2 \cdot K_E, & x_B < x \le x_E \\[2mm] 3/2 \cdot K_E \cdot \dfrac{(x_E + \Delta x_V) - x}{\Delta x_V}, & x_E < x \le x_E + \Delta x_V \\[2mm] 0, & x > x_E + \Delta x_V \end{cases},$$

<div align="right">Eq. 3.45</div>

where Δx_V represents the length of the vehicle ($10 \cdot \tau_P$ = 240 mm), and x_B and x_E represent the beginning and the end positions of the segment (defined as the first, respectively the last position where the vehicle is completely inside the considered segment – see also Figure 3.36).

When the vehicle is completely outside the segment, the current i_q produces no force, $K_F(x) = 0$; when the vehicle enters/leaves the segment, a linear increase/decrease of $K_F(x)$ is assumed.

For two consecutive stator segments, m and n, the electrical force F_E can be expressed as:

$$F_E = \left[K_{F,m}(x) + K_{F,n}(x)\right] \cdot i_q$$

<div align="right">Eq. 3.46</div>

In Eq. 3.46 it is assumed that the force-producing current i_q is controlled to the same value for both segments.

An illustration of the principle of determination of the current-force coefficient at the transition between two consecutive stator segments m and n (which have different EMF constants $K_{E,m}$ and $K_{E,n}$) is given in Figure 3.36.

Based on the mechanical model of the machine (Eq. 3.43), the mechanical observer will be implemented using the state equations:

$$\begin{cases} \dfrac{d\hat{F}_L}{dt} = G_F \cdot \varepsilon_x \\[3mm] \dfrac{d\hat{v}}{dt} = \dfrac{F_E^*}{M_v} - \dfrac{\hat{F}_L}{M_v} - \dfrac{B_v}{M_v} \hat{v} + G_v \cdot \varepsilon_x, \\[3mm] \dfrac{d\hat{x}}{dt} = \hat{v} + G_x \cdot \varepsilon_x \end{cases}$$

<div align="right">Eq. 3.47</div>
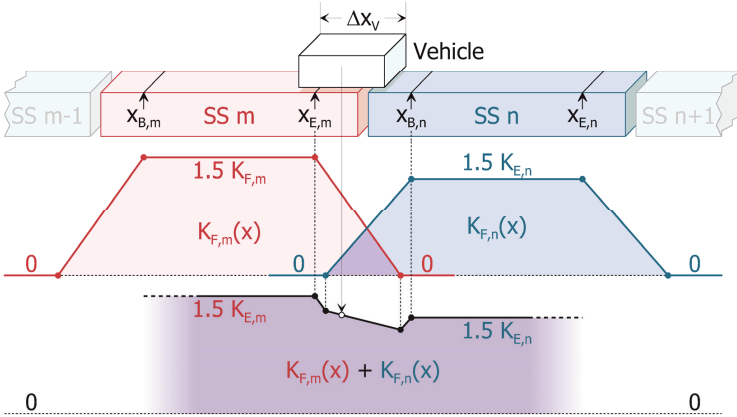
Figure 3.36: Current-force coefficients

where $\hat{F}_L$, $\hat{v}$ and $\hat{x}$ are the estimated load force, velocity and position, respectively. The reference value of the electrical force F_E^* is calculated based on the reference current i_q^*. M is the mass the vehicle and B is the friction coefficient. G_F, G_v, and G_x are the gains of the mechanical observer. The correction term ε_x is discussed below.

For only one stator segment, the correction ε_x is calculated:

$$\varepsilon_x = \text{sgn}\left(v^*\right) \cdot \left[\cos\frac{\pi\hat{x}}{\tau_P} \quad \sin\frac{\pi\hat{x}}{\tau_P} \right] \cdot \begin{bmatrix} \hat{e}_\alpha \\ \hat{e}_\beta \end{bmatrix}$$

Eq. 3.48

In order to analyse the action of the correction term, Eq. 3.14 will be introduced in Eq. 3.48, yielding:

$$\varepsilon_x = \text{sgn}\left(v^*\right) \cdot v \cdot K_E \cdot \left[\cos\frac{\pi\hat{x}}{\tau_P} \quad \sin\frac{\pi\hat{x}}{\tau_P} \right] \cdot \begin{bmatrix} -\sin\frac{\pi x}{\tau_P} \\ \cos\frac{\pi x}{\tau_P} \end{bmatrix}.$$

Eq. 3.49

By assuming that sgn(v) = sgn(v^*), ε_x can be reduced to:

$$\varepsilon_x = K_E \left| v \right| \sin\left(\frac{\pi}{\tau_P}\hat{x} - \frac{\pi}{\tau_P}x \right).$$

Eq. 3.50

If x and $\hat{x}$ are close to each other, then ε_x can be liniarised:

$$\varepsilon_x \simeq K_E \left| v \right| \frac{\pi}{\tau_P}\left(\hat{x} - x\right)$$

Eq. 3.51

For analysing the dynamics of the mechanical observer, the same method of the error dynamics will be used, as in the case of the EMF observers. Defining the estimation errors:

$$\tilde{F}_L = F_L - \hat{F}_L, \quad \tilde{v} = v - \hat{v}, \quad \tilde{x} = x - \hat{x},$$

Eq. 3.52

the equation describing the dynamics of the estimation errors can be derived from Eq. 3.43 and Eq. 3.47:

$$\frac{d}{dt}\begin{bmatrix} \tilde{F}_L \\ \tilde{v} \\ \tilde{x} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & G_F\left(K_E|v|\pi/\tau_P\right) \\ -\dfrac{1}{M_v} & -\dfrac{B_v}{M_v} & G_v\left(K_E|v|\pi/\tau_P\right) \\ 0 & 1 & G_x\left(K_E|v|\pi/\tau_P\right) \end{bmatrix}}_{\mathbf{A}}\begin{bmatrix} \tilde{F}_L \\ \tilde{v} \\ \tilde{x} \end{bmatrix}.$$

Eq. 3.53

Matching the coefficients of the characteristic polynomial

$$\det\left(s\mathbf{I}_3 - \mathbf{A}\right) = 0$$

Eq. 3.54

with the ones of a 3rd order polynomial with the poles p_1, p_2 and p_3, the following expressions are obtained for the mechanical observer's gains:

$$\begin{cases} G_F = \dfrac{M_v \cdot p_1 \cdot p_2 \cdot p_3}{K_E \cdot |v| \cdot \pi/\tau_P} \\[3mm] G_v = -\dfrac{\left(B_v/M_v\right)^2 + B_v/M_v \cdot \left(p_1 + p_2 + p_3\right) + p_1 \cdot p_2 + p_2 \cdot p_3 + p_3 \cdot p_1}{K_E \cdot |v| \cdot \pi/\tau_P} \\[3mm] G_x = \dfrac{B_v/M_v + p_1 + p_2 + p_3}{K_E \cdot |v| \cdot \pi/\tau_P} \end{cases}.$$

Eq. 3.55

From Eq. 3.55 it can be seen that the eigenvalues of the observer will move as the speed changes (if fixed values are considered for the gains).

The gains of the observer will be calculated in such a manner, as to achieve a desired dynamic for a certain minimal speed v_0, and then the locus of the observer's poles will be analysed for the entire speed range.

Let us assume, that the dynamic of a 3rd order Butterworth filter, with cut-off frequency at 20 Hz, is desired for a minimal speed $v_0 = 0.5$ m/s. Following values result for the mechanical observer's poles:

$$\begin{cases} p_1 = -125.66 \\ p_{2,3} = -62.83 \pm j \cdot 108.83. \end{cases}$$

Eq. 3.56

Introducing these values in Eq. 3.55, together with the vehicle's mass $M_v = 13.2$ Kg and the pole pitch $\tau_P = 24$ mm, and considering the EMF constant of stator segment 1, $K_E = 17.72$ Vs/m and a friction coefficient $B_v = 50$ Kg/s, the following gains result at $v_0 = 0.5$ m/s:

$$\begin{cases} G_F = -23.542 \cdot 10^3 \\ G_v = -27.543 \\ G_x = -0.2225 \end{cases}.$$

Eq. 3.57

Figure 3.37 shows the root locus of the obtained mechanical observer, over the entire speed range $v = 0\ldots10$ m/s. It can be seen, that for speeds higher than v_0, the dynamic of the dominant poles p_2 and p_3 changes relatively little.

With increasing speed, the pole p_1 moves further away from the origin of the s-plane, and, for speeds close to the maximal value of 10 m/s, it approaches the poles of the EMF
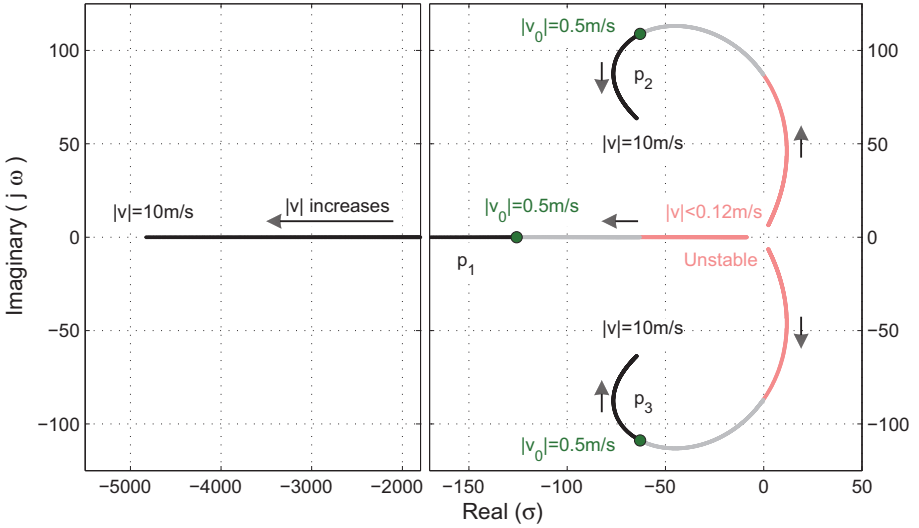
Figure 3.37: Root locus of the mechanical observer

observer. This, however, should have no influence on the estimation dynamic if the decoupled EMF observer (as described by Eq. 3.22) is used.

For speeds lower than v_0, the mechanical observer is still stable, but the dynamic depreciates with decreasing speed. For speeds lower than approx. 0.12 m/s, the dominant poles of the observer move to right half-plane, leading to instability.

It can be seen from Figure 3.37, that, as the speed v approaches zero, the poles of the mechanical observer converge to the origin of the s-plane.

One must however note that, in the right half-plane, the dominant poles remain close to the imaginary axis (slow dynamic), which makes possible to reverse the vehicle's moving direction using EMF-based, sensorless speed control (this implying a passing through zero of the speed), so long the reversal occurs fast enough (see also the experimental results in the next subsection).

Up to this point, only one stator segment was considered in the analysis of the mechanical observer.

If two consecutive stator segments, m and n, with **no relative phase displacement** are considered, Eq. 3.48 can still be used to calculate the correction term [62], replacing $\hat{e}_\alpha$ with $\hat{e}_{\alpha,m} + \hat{e}_{\alpha,n}$ and $\hat{e}_\beta$ with $\hat{e}_{\beta,m} + \hat{e}_{\beta,n}$, respectively:

$$\varepsilon_x = \text{sgn}(\hat{v}) \cdot \left[\cos\frac{\pi\hat{x}}{\tau_P} \quad \sin\frac{\pi\hat{x}}{\tau_P} \right] \cdot \begin{bmatrix} \hat{e}_{\alpha,m} + \hat{e}_{\alpha,n} \\ \hat{e}_{\beta,m} + \hat{e}_{\beta,n} \end{bmatrix}.$$

Eq. 3.58

In case there is a **phase difference** between the two observed EMF vectors, the correction term ε_x will be calculated as a sum of two partial corrections, one for each controlled segment:

$$\varepsilon_x = \varepsilon_{x,m} + \varepsilon_{x,n}$$

Eq. 3.59

The two partial corrections $\varepsilon_{x,m}$ and $\varepsilon_{x,n}$ are calculated based on the estimated EMF vectors corresponding to the two controlled segments, and on the two estimated electrical angles $\hat{\theta}_m$ and $\hat{\theta}_n$:

$$\begin{cases} \varepsilon_{x,m} = \text{sgn}(\hat{v}) \cdot \begin{bmatrix} \cos(\hat{\theta}_m) & \sin(\hat{\theta}_m) \end{bmatrix} \cdot \begin{bmatrix} \hat{e}_{\alpha,m} \\ \hat{e}_{\beta,m} \end{bmatrix} \\ \varepsilon_{x,n} = \text{sgn}(\hat{v}) \cdot \begin{bmatrix} \cos(\hat{\theta}_n) & \sin(\hat{\theta}_n) \end{bmatrix} \cdot \begin{bmatrix} \hat{e}_{\alpha,n} \\ \hat{e}_{\beta,n} \end{bmatrix} \end{cases}.$$

Eq. 3.60

The estimated electrical angles of the two controlled segments are determined from the estimated position $\hat{x}$, taking into account the different initial phases $\theta_{0,m}$ and $\theta_{0,n}$ of the segments:

$$\hat{\theta}_m = \frac{\pi \hat{x}}{\tau_P} + \theta_{0,m} \quad \hat{\theta}_n = \frac{\pi \hat{x}}{\tau_P} + \theta_{0,n},$$

Eq. 3.61

the initial phase of each segment being considered with respect to the first segment of the track (SS1).

A block diagram representation of the mechanical observer described by Eq. 3.47, Eq. 3.59, Eq. 3.60, and Eq. 3.61 is given in Figure 3.38.

3.3.3 Implementation at the Straight Section of the Machine (Rigid Vehicle)

The three-observers structure described in the previous subsections was first implemented at section 1 of the linear machine (straight, double-sided section, consisting of stator segments 1 and 2). The rigid vehicle was used in this implementation. With this vehicle it was possible to set the air-gap at ca. 2.5 mm (smaller than the 3 mm air-gap set for the articulated vehicle), so a slightly higher EMF constant was attained in this implementation (about 20 Vs/m).

Figure 3.39 shows an offline measurement of the phase "a" component of the EMF at the transition between stator segments 1 and 2. The induced voltages in the two segments are in phase, so the correction term of the mechanical observer will be
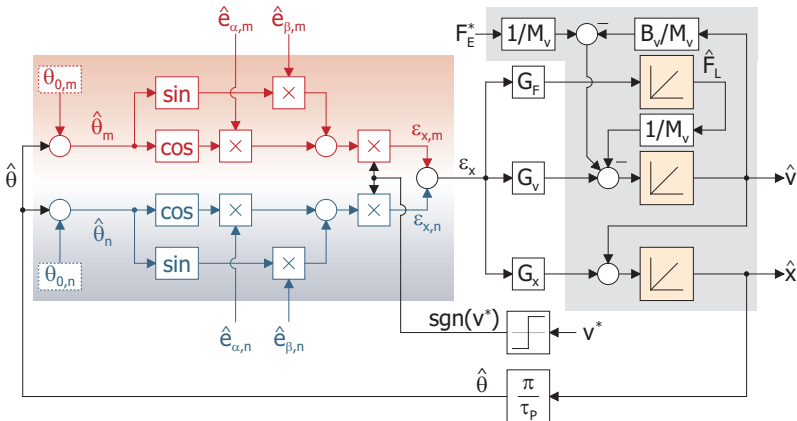
Figure 3.38: Block diagram of the mechanical observer
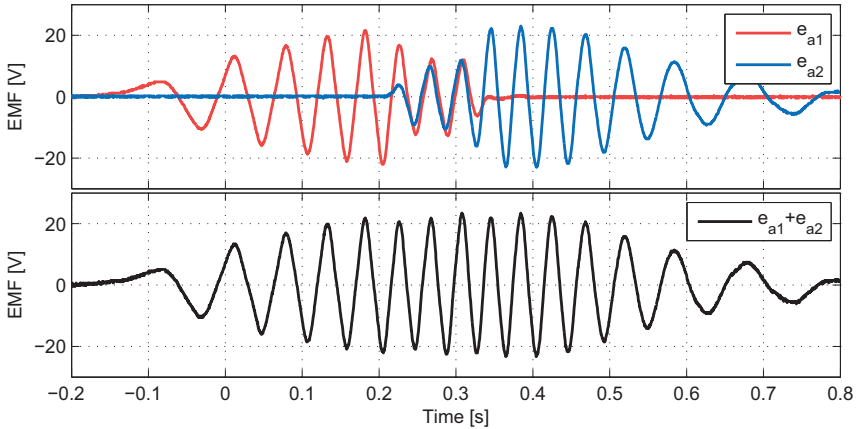
Figure 3.39: Offline measurement of the EMF at the transition between stator segments 1 and 2 (Rigid Vehicle)

calculated as described in Eq. 3.58 (m = 1, n = 2). In Figure 3.39, it can also be seen that there is no noticeable drop in the sum of the two EMF components, which was to be expected, as there is no gap between the two segments.

Figure 3.40 shows a measurement in which the observed position and speed were used for field orienting, as well as for feedback to the position and speed controllers. The sensed position and speed were used for starting from standstill, at the beginning of the control algorithm (not shown in Figure 3.40 – see section 3.4 for a description of the synchronisation procedure between the sensed and the estimated position and speed); after that the sensed position and speed were only used for evaluating the estimation errors. In this measurement position control was used (with the reference position step-changing between 5 and 595 mm – close to the limits of the sensors' region); the speed was limited to 2 m/s, and the force-producing currents i_{q1} and i_{q2} were limited at 10 A (about the half of the maximal current).

In order to reverse the direction of motion (taking into account that positioning is not possible under EMF-based, sensorless control), the following procedure was used: the proportional gain of the position controller was set high enough, so that the controlled position would overshoot the reference position. This means that when the vehicle reaches the reference position its speed is decreasing, but it still has a high enough value for the mechanical observer to remain stable. When the reference position is first reached by the controlled position, a step-change occurs in the reference position, determining the vehicle to accelerate in the opposite direction.

It can be seen in Figure 3.40 (a) and (b) that, when the estimated speed is close to zero, the estimation errors increase. The maximal position error is about 4 mm (corresponding to an error in the electrical angle of ca. 30°, and to a reduction of ca. 15% in the electrical force). The reduced electrical force is however still high enough for the vehicle to accelerate in the opposite direction. During this acceleration, as the speed increases, the estimation errors decrease again.

In Figure 3.40 (c) the force-producing currents of the two segments, i_{q1} and i_{q2}, are shown. When the vehicle is inside a stator segment (completely or partially), the current of that segment is controlled to the reference value obtained at the output of the speed controller. When the vehicle is at the transition between the two segments, the currents of both segments are controlled.
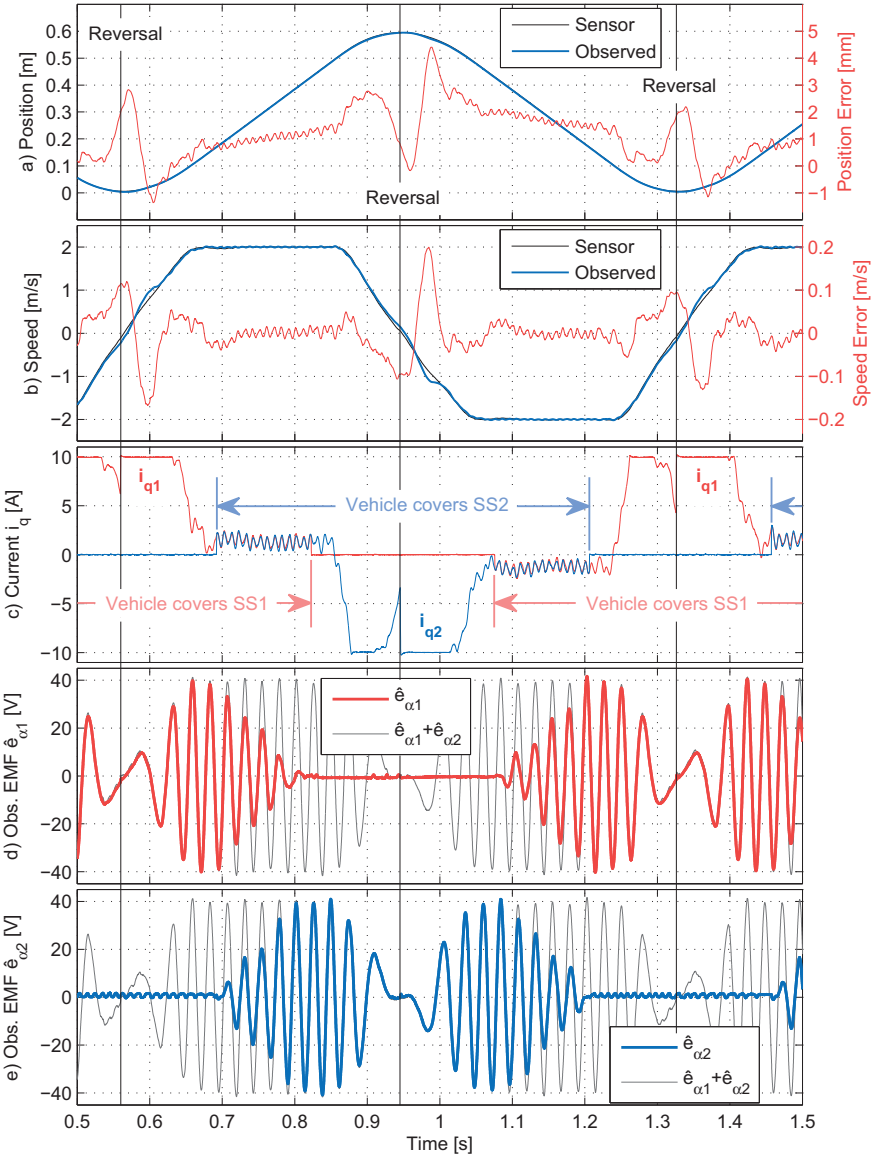
Figure 3.40: Field oriented control, using the estimated position and speed: a) Sensed and estimated position, position error; b) Sensed and estimated speed, speed error; c) Force-producing current in the two stator segments (i_{q1}, i_{q2}); d) Obs. EMF (α) in segment 1; e) Obs. EMF (α) in segment 2.

In Figure 3.40 (d) and (e) the observed "α" components of the EMF vectors corresponding to the two segments are shown. The thin grey curve in these two figures represents the sum of the "α" components of the two EMF vectors.

3.3.4 Implementation on the Entire Length of the Machine (Articulated Vehicle)

Due to the modular construction of the linear machine, there are gaps (spacings) in the stator winding between consecutive stator segments – see Figure 3.41 (a). These gaps are generally not equal between each other, and not a multiple of $2\tau_P$ (τ_P being the pole pitch of the linear machine), so they will generate phase displacements between the estimated EMF vectors of two consecutive stator segments, which displacements must be taken into consideration in the implementation of the mechanical observer.

The phase differences between each two stator segments were determined by offline EMF measurements at the transitions between segments, as illustrated in Figure 3.41 (b). The phase differences between all 18 stator segments are listed in Table 3.5.

Figure 3.42 shows the α-component of the observed EMF vectors, as well as the observed speed during motion of the vehicle from segment 4 to segment 6. At the transitions between segments, the phase differences of the observed EMFs can be seen.

It can also be noticed that the amplitude of $\hat{e}_{\alpha,5}$ varies significantly, while its frequency remains approx. constant, implying that the amplitude variation is caused by changes of the air-gap.
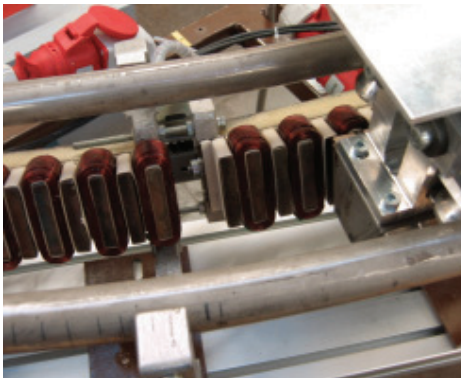
a) Spacing in the machine's winding   b) Resulting phase difference in the EMF
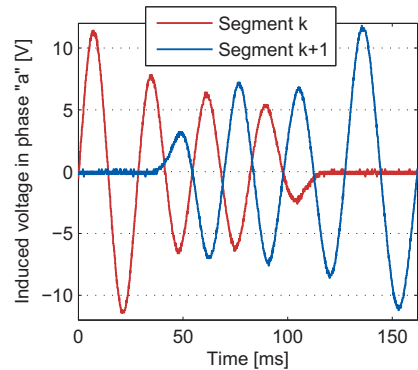
Figure 3.41: Phase difference in the EMF at the transition between stator segments

Segm. (k)	$\Delta\theta_{k,k+1}$ [°]	Segm. (k)	$\Delta\theta_{k,k+1}$ [°]	Segm. (k)	$\Delta\theta_{k,k+1}$ [°]
1	0,00	7	167,36	13	96,70
2	317,35	8	360,00	14	166,55
3	80,96	9	321,81	15	104,87
4	127,35	10	0,00	16	164,97
5	157,27	11	43,75	17	84,01
6	194,71	12	133,86	18	305,20

Table 3.5: Measured phase differences between all stator segments
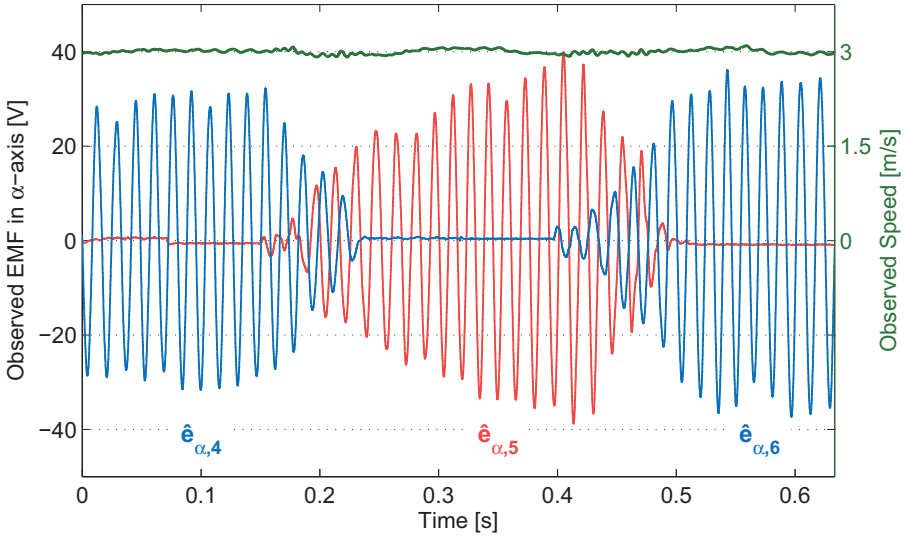
Figure 3.42: The α-components of the observed EMF and the observed speed (SS4, SS5 and SS6)

Because the mechanical observer described in Section 3.3.2 only uses the phase information contained in the observed EMF vectors, these air-gap changes are not reflected in the estimated speed.

3.4 Sensor/Sensorless Transition

As previously described, there are two sources for position and speed information:

- When the vehicle is inside the <u>processing station</u>, the sensor-provided position and speed (x_S, v_S) are used in control;
- When the vehicle travels through the <u>transport region</u>, the estimated position and speed ($\hat{x}$, $\hat{v}$) are used as feedback by the control algorithm.

The position and speed synchronisation function (as depicted in Figure 3.20; see also the block diagram in Figure 3.34) determines which position and velocity will be used in control (x_C, v_C).

Based on the position x_C and on the linear machine's parameters (length of each stator segment, length of the vehicle, spacing between each two segments), the synchronisation function also determines the addresses m and n of the Power Stacks, which correspond to the stator segment(s), where the vehicle is located.

Due to the closed shape of the track of the machine used in our experimental setup, a wrapping of the position x_C must also be implemented. Because the estimated position is not absolute, and only one processing station exists in our setup, the zero position of the processing station (i.e. the zero position of the optical sensors system) was chosen as zero position of the entire track. More details about the position wrapping are given in subsection 3.4.2.

Figure 3.43 shows, schematically, the implemented synchronisation procedure between the sensed and the estimated position and speed.

Let us assume, that initially the vehicle is moving inside the processing station, using feedback from the position sensors, and approaching one of the processing station's
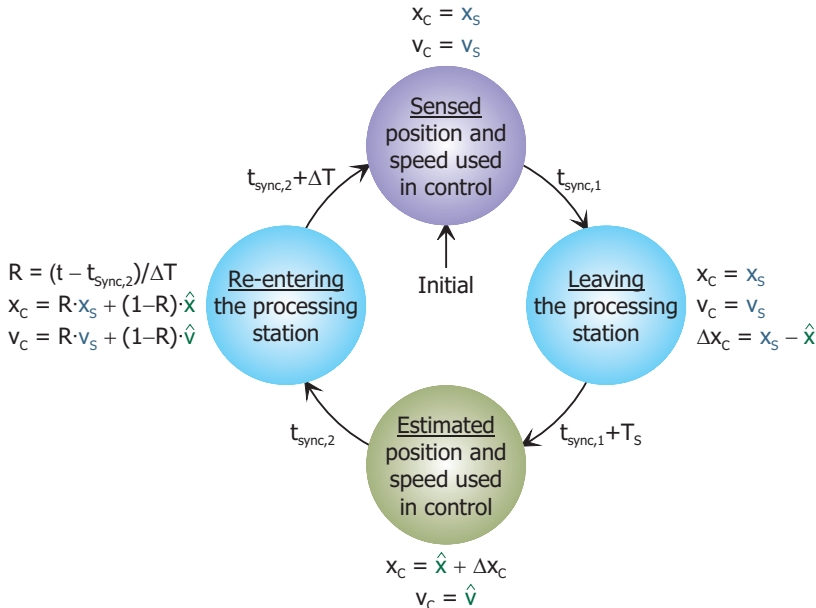
Figure 3.43: Synchronisation between sensed and the estimated position and speed

extremities (i.e., the point where the sensors will provide no more valid position information). Following equation is valid in this situation:

$$x_C(t) = x_S(t), \qquad \text{for } t \leq t_{sync,1}.$$ **Eq. 3.62**

The time, when the sensor based position x_S approaches its maximal or minimal value is called $t_{sync,1}$. After this time, the control algorithm will start using the observed position and velocity. Taking into consideration that the observed position $\hat{x}$ is not absolute (as opposed to the one given by the implemented sensors system), the difference Δx_C between the two positions will be calculated at $t_{sync,1}$:

$$\Delta x_C = x_S(t_{sync,1}) - \hat{x}(t_{sync,1}), \quad \text{for } t = t_{sync,1}.$$ **Eq. 3.63**

The difference Δx_C will be used to correct the estimated position. After $t_{sync,1}$, the position used in control will be calculated as:

$$x_C(t) = \hat{x}(t) + \Delta x_C, \qquad \text{for } t > t_{sync,1}.$$ **Eq. 3.64**

This method of synchronization also ensures a smooth transition of the electrical angle used for field orientation. For more details, see the discussion of the experimental results in subsection 3.4.1.

When the vehicle enters the processing station, it must be taken into consideration that the position given by sensors is more accurate than the estimated one, so, in this case, using a position correction similar to the one defined by Eq. 3.63 will only introduce an error in the sensed position x_S.

The straightforward way to do the synchronisation in this case would be simply to use x_S as soon as it is available (as soon as the vehicle enters the processing station). But, if the estimation error in the observed position is large, a step will be introduced in the electrical angle and, subsequently, in the generated electrical force.

In order to avoid this step in the electrical angle, a smooth synchronisation can be realised over a time interval ΔT, starting at $t_{sync,2}$. Using a ramp function defined as:

$$R(t) = \begin{cases} 0, & t - t_{sync,2} < 0 \\ \dfrac{t - t_{sync,2}}{\Delta T}, & 0 \leq t - t_{sync,2} \leq \Delta T \\ 1, & t - t_{sync,2} > \Delta T \end{cases}$$ **Eq. 3.65**

the position used in control can be determined, during the time interval ΔT, as follows:

$$x_C(t) = R(t) \cdot x_S(t) + \left[1 - R(t)\right] \cdot \hat{x}(t).$$ **Eq. 3.66**

A similar expression, using the same ramp function, can be used for the calculation of the speed v_C during ΔT, in order to avoid a step in the reference current (due to speed estimation error).

3.4.1 Leaving the Processing Station

In this subsection, experimental results acquired at the vehicle's leaving of the processing station are discussed.

Figure 3.44 shows the sensed and estimated position and speed, as well as the respective estimation errors, at the transition between the processing station and the subsequent transport section. All the quantities were acquired in the same test run.

The control algorithm starts at time t_0. At this time the vehicle is inside the processing station ($x_S \approx 0.013$ m), so the sensors provide valid position and speed information. When the control starts, the vehicle begins to move towards a commanded

position, outside the processing station, using initially the feedback information given by the position sensors.

The three-observers structure is disabled until the vehicle gains some speed – time t_1 in Figure 3.44. After being enabled, the observers need the time interval $t_1 - t_2$ (ca. 100 ms) to converge.

In the interval $t_2 - t_4$ valid position and speed information is available from the optical sensors, as well as from the observers, so the estimation errors can be analysed.
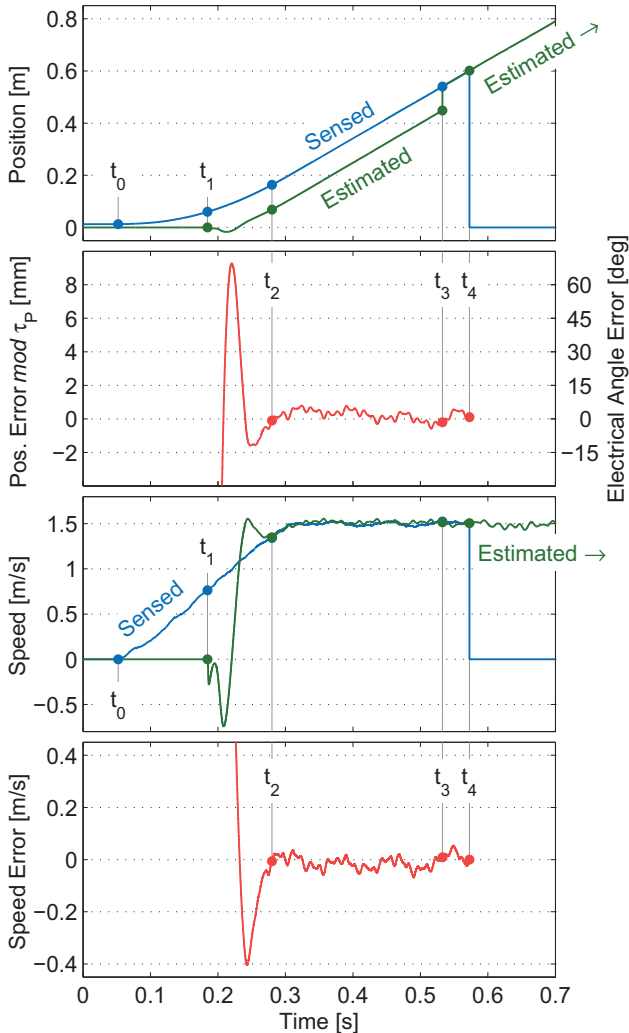
Figure 3.44: Measurement of the position and speed when leaving the processing station. From top: Sensed and estimated position; Position / electrical angle estimation error; Sensed and estimated speed; Speed estimation error.

Before synchronisation, which occurs at time $t_3 = t_{sync,1}$, according to Eq. 3.63, the mean position estimation error amounts to 96 mm, i.e. 4 τ_P (due to the fact that the observers were not started close to the zero of the sensed position).

Starting with time t_3 the estimated position and speed are used as feedback in control. In Figure 3.44, the position estimation error is represented modulo $2\tau_P$, so the synchronisation at time t_3 does not affect the error curve.

After the estimated position converges, the estimation error remains less than ±1 mm (corresponding to an orientation error of ±7.5 electrical degrees). The speed estimation error is less than ±0.05 m/s, in steady state.

At time t_4 the range of the sensors is exceeded and the sensor-based signals are set to zero. The vehicle has exited the processing station and moves, sensorless, through the transport section, towards the next processing station.

3.4.2 Re-entering the Processing Station

In Figure 3.45, the position from the sensors x_S, the observed position $\hat{x}$ and the position used in control x_C are shown, when the vehicle enters the processing station.

At the beginning, the vehicle moves through the transport region, and the observed position is used in control ($x_C = \hat{x}$). The vehicle enters the processing station at time t_0 (starting from this time the optical sensors provide valid position information x_S), with the observed position $\hat{x}$ amounting to ca. 12.477 m (which represents the total length of the track, abstraction being made of the estimation error).

The synchronization process starts at $t_{sync,2}$ (one control interrupt after t_0, so that the sensed speed v_S is also available), and it will be completed in two steps: In the first step, at
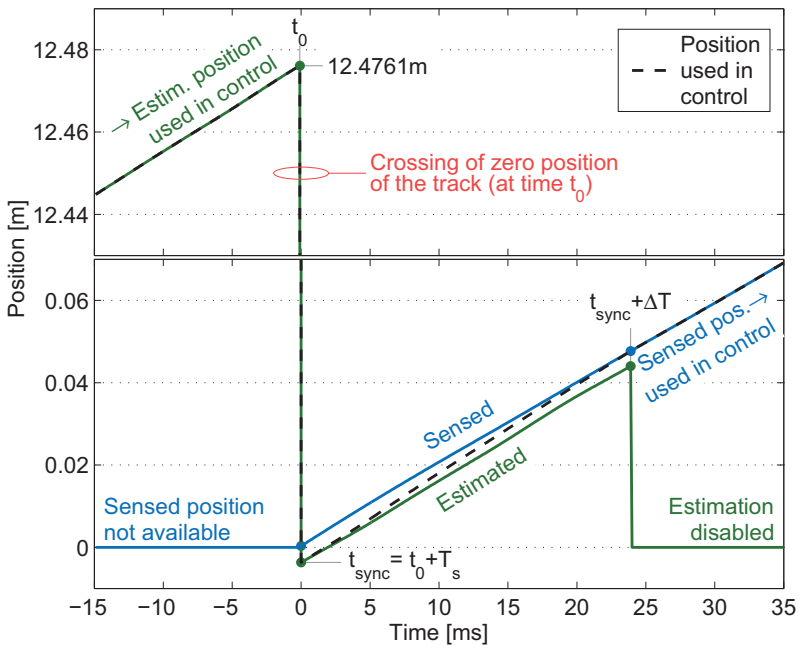
Figure 3.45: Measurement of the position given by sensors, of the observed position, and of the position used in control, when the vehicle re-enters the processing station

time $t_{sync,2}$, the observed position $\hat{x}$ is wrapped around (this is necessary due to the closed shape of the track), and brought as close as possible to sensors' position x_S, but without affecting the electrical angle used in control:

$$\hat{x}'(t_{sync,2}) = \hat{x}(t_{sync,2}) - 2k\,\tau_P, \qquad \text{with } k \in \mathbb{Z}\,, \qquad\qquad \textbf{Eq. 3.67}$$

so that

$$x_S(t_{sync,2}) - \tau_P < \hat{x}'(t_{sync,2}) \le x_S(t_{sync,2}) + \tau_P\,. \qquad\qquad \textbf{Eq. 3.68}$$

After the wrapping of the observed position, a small difference still remains between $\hat{x}$ and x_S (ca. 3 mm – due mainly to inaccuracies in determining the phase displacements between consecutive stator segments). This corresponds to a phase difference, which will be eliminated in the second step, during the time ΔT, using the ramp function defined in Eq. 3.65.

If the position control loop is active when the vehicle passes through the zero of the track, the reference position x^* must also be wrapped. In this case no ramping is necessary; the wrapping will occur as soon as the sensed position is available (i.e. at time t_0 in Figure 3.45):

$$x^{*\,\prime}(t) = x^*(t) - \left[\hat{x}(t_0) - x_S(t_0)\right], \qquad \text{for } t \ge t_0\,. \qquad\qquad \textbf{Eq. 3.69}$$

In the above equation, the length of linear machine is subtracted from the reference position, simultaneously compensating for any position estimation error.

One must note that the resulting reference position must not be too close to the zero of the sensors – no positioning is possible until the synchronisation between the estimated and the sensed speed is completed (a speed close to zero would lead to instability of the observers structure).

At the time $t_{sync,2} + \Delta T$ the position used in control x_C equals the sensed position x_S and the synchronisation is completed. The observers can now be disabled – the sensed position and speed will be used in control.

4 Conclusions

In this work, aspects concerning the position acquisition and control for linear drives with passive vehicles were investigated. For industrial material handling, a combined transportation and processing system based on a segmented, permanent magnet, linear drive is developed. In order to attain high accelerations, passive, lightweight vehicles are used. The track of the drive contains processing stations, which must be equipped with position sensors, for high accuracy of positioning and high dynamics.

Passive vehicles present a restriction for the position acquisition system, through the fact that neither energy nor information must be transmitted to the moving parts. In Chapter 2, the realisation and testing of two position sensing systems, which comply with this requirement, were examined: the first one is based on high resolution optical sensors, while the second one uses a comparatively lower resolution capacitive sensor. For the capacitive sensor two evaluation methods were tested: the first one uses square-wave excitation and instantaneous demodulation, and the second is based on sinusoidal excitation and position determination by phase measurement. Table 4.1 gives a comparative overview of the evaluated position sensing systems.

Evaluated System / Evaluation criterion		Optical Sensors System	Capacitive Encoder	
			Square-Wave Excitation	Sinusoidal Excitation
Pitch of the sensor		40 µm	2000 µm	
Position error without any corrections (peak to peak)	Integral	20 µm	120 µm	
	Local	10 µm	100 µm	
Position error with parametric corrections (peak to peak)	Integral	10 µm	120 µm	
	Local	1 µm	30 µm (see note)	
Complexity of the signal processing hardware		Very High (four ADCs plus synchronisation firmware)	Moderate (only one fast ADC is required)	Low (only digital processing)
Computational resources		High (arctangent, divisions, etc.)	High (arctangent, divisions, etc.)	None
Cost of the system (incl. signal processing)		High (Large number of read-heads required)	Low	Very Low

Note: For evaluation of the capacitive encoder with sinusoidal excitation, only a mean correction was used. Using individual period corrections (as in the other two cases) would certainly improve this result.

Table 4.1: Comparative overview of the evaluated position sensing systems

For both optical and capacitive sensors, correction tables were used in order to improve the accuracy; these corrections tables were generated based solely on the information provided by the evaluated sensors (i.e., without the use of the reference sensor, which is typically not available in a practical application).

Compared to optical sensors, the capacitive sensors achieve a lower accuracy, but the hardware and the signal processing for capacitive sensors are extremely simple (especially when the evaluation method based on phase measurement is used), and can be produced at very low cost, so the capacitive sensors can be an attractive alternative to the optical ones for applications where their accuracy can be accepted.

The optical sensors system was implemented at the experimental linear drive described in Chapter 3 of this dissertation. For the power electronics and control of the drive, a centralised architecture is used. A dedicated Power Stack feeds each of the 18 segments of the drive, and a controller is assigned to each vehicle (for the time being, only one vehicle is available at the linear drive). All Power Stacks, as well as the Control PC (where the Vehicle Controller is implemented) are located in a central cabinet. Control information is exchanged between the Vehicle Controller and the Power Stacks via an Inverter Bus.

In the range outside of the processing station (transport section), EMF-based, sensorless position control was implemented. The construction of the linear machine must allow for gaps between consecutive stator segments, which bring phase differences in the EMFs of the segments. These phase differences were addressed in the implementation of the observer structure used in sensorless control.

At the transition between the processing station and transport section, the synchronisation between the sensor-based, and the sensorless position feedback was also implemented in the control algorithm, and was experimentally tested.

4.1 Future work

Future work regarding the control of the linear drive for material handling described in this thesis will be oriented towards expanding the functionality of the drive. Concerning the position acquisition and motion control, the project will be continued in two main directions:

1) The implementation at the existing linear drive of a sensorless method based on the evaluation of the machine's anisotropies, by means of injecting of a test signal. This will allow for positioning of the vehicles in the transport region, too. The two sensorless methods – the anisotropies-based method, and the one described in Section 3.3 (EMF-based) – complement each other: the first method yields good results at standstill and low speeds, while the second one is more advantageous at higher speeds. Both methods should be implemented in the control algorithm, and a synchronisation (switching) procedure between them should also be investigated.

2) The operation of the linear drive with more than one vehicle must be tested. For this, at least a second vehicle must be added to the existing drive. Each vehicle will have its own Vehicle Controller, and in order to coordinate the movement of the vehicles, a Motion Coordination PC will be used (see Figure 3.1). This Motion Coordination PC must generate the position reference values for each vehicle. For this, a slower cycle time is sufficient (typical values are around $1 - 2$ ms). In the same cycle time, the Vehicle Controllers should send back their measured positions, as well as their status, for monitoring. Then, in each Vehicle Controller an interpolator should be integrated, to generate the intermediate position reference values for the 100 µs position control cycle. Because the real-time communication demand between the Control PCs and the Motion Coordination PC is less stringent (compared, e.g. with the demand on the Inverter Bus), the Ethernet interface, which is nowadays present in every PC, could be used for this purpose.

5 Appendix

5.1 Imaging Scanning Principle

"The LIDA linear encoders operate according to the imaging scanning principle. […] To put it simply, the imaging scanning principle functions by means of projected light signal generation: two scale gratings with equal grating periods are moved relative to each other — the scale and the scanning reticle. The carrier material of the scanning reticle is transparent, whereas the graduation on the measuring standard may be applied to a transparent or reflective surface. When parallel light passes through a grating, light and dark surfaces are projected at a certain distance. An index grating with the same grating period is located here. When the two gratings move in relation to each other, the incident light is modulated: if the gaps are aligned, light passes through. If the lines of one grating coincide with the gaps of the other, no light passes through. Photocells convert these variations in light intensity into electrical signals. The specially structured grating of the scanning reticle filters the light current to generate nearly sinusoidal output signals. The smaller the period of the grating structure is, the closer and more tightly toleranced the gap must be between the scanning reticle and scale. Practical mounting tolerances for encoders with the imaging scanning principle are achieved with grating periods of 10 µm and larger." (Source: [26])
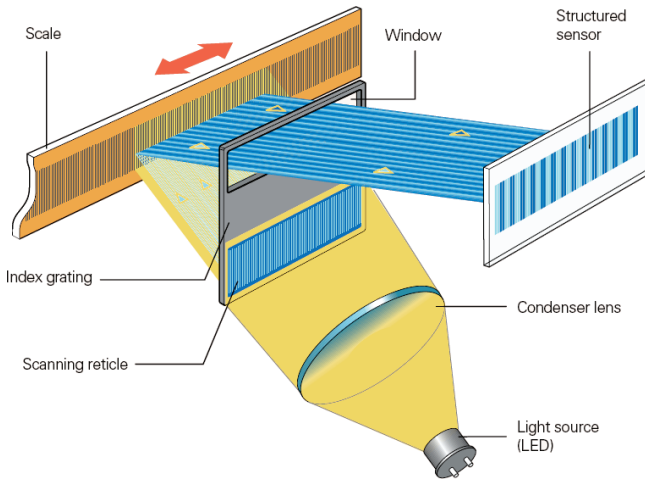
Figure 5.1: Imaging scanning principle. Source: [26]

5.2 Dimensions of the Capacitive Sensor

<u>Geometrical dimensions:</u>

- Pitch of the sensor:
  $P = 2\,mm$
- Width of the transmitting electrodes:
  $w = 0.35\,mm$ (70% of P/4)
- Height of the upper / lower sinusoidal copper pattern:
  $A = 10\,mm$
- Height of middle (rectangular) copper pattern:
  $h = 2\,mm$
- Total length of the sensor:
  $L = 560\,mm$
- Total number of periods (pitches):
  $N_p = L / P = 280$
- Length of the slider (mobile plate):
  $L_m = 60\,mm$
- Number of periods on the slider:
  $N_{mp} = L_m / P = 30$
- Spacing between transmitting and modulating plates:
  $d_1 = 0.4\,mm$
- Spacing between modulating and receiving plates:
  $d_2 = 0.4\,mm$
- Thickness of each of the three plates:
  $d_3 = 1.5\,mm$

  Electrical permittivities:
- Absolute permittivity of air (approximated to that of vacuum):
  $\varepsilon_0 = 8.854187817 \cdot 10^{-12}\,F/m$
- Relative permittivity of epoxy (FR4) [35]:
  $\varepsilon_3 = 4.7$

Figure 5.2:
Dimensions of the position-dependent overlapping areas

Based on the above values, the different <u>capacitances</u> can be calculated, using the homogeneous field distribution approximation:

- Average value of the position dependent capacitances:

$$C_{avg} = N_{mp} \cdot \varepsilon_0 \cdot w \cdot \left(\frac{h + A}{d_1} + \frac{A}{d_1 + \dfrac{d_3}{\varepsilon_3}} \right) = \mathbf{4.082\,pF}$$

- Variation amplitude of the position dependent capacitances:

$$C_{var} = N_{mp} \cdot \varepsilon_0 \cdot \frac{PA}{\pi} \cdot \left(\frac{1}{d_1} - \frac{1}{d_1 + \frac{d_3}{\varepsilon_3}} \right) \cdot \sin\left(\frac{w\pi}{P} \right) = \mathbf{0.980\,pF}$$

This less than 1 pF variation of the capacitances encodes the position information.

- Coupling capacitance (between modulating and receiving electrodes):

$$C_k = \frac{N_{mp} \cdot \varepsilon_0 \cdot P(2A + h)}{d_2} = \mathbf{29.219\,pF}$$

- Position independent capacitance between transmitting and receiving electrodes:

$$C_0 = \frac{(N_p - N_{mp}) \cdot \varepsilon_0 \cdot w(2A + h)}{d_1 + d_2 + d_3} = \mathbf{7.411\,pF}$$

- Input capacitances (in parallel with the voltage sources):

$$C_{j,in} = \frac{N_p \cdot \varepsilon_0 \cdot \varepsilon_3 \cdot w(2A + h)}{d_3} = \mathbf{59.814\,pF} \text{ (j = a, b, c, d)}$$

- Output capacitance (short-circuited by the charge amplifier):

$$C_{out} = \frac{N_p \cdot \varepsilon_0 \cdot \varepsilon_3 \cdot P(2A + h)}{d_3} = \mathbf{341.795\,pF}$$

Additionally, for the <u>charge amplifier feedback</u>, following components were used:

- Feedback capacitance:

$$C_R = \mathbf{20\,pF}$$

- Feedback resistance:

$$R_R = \mathbf{20\,M\Omega}$$

5.3 Supply Control Board
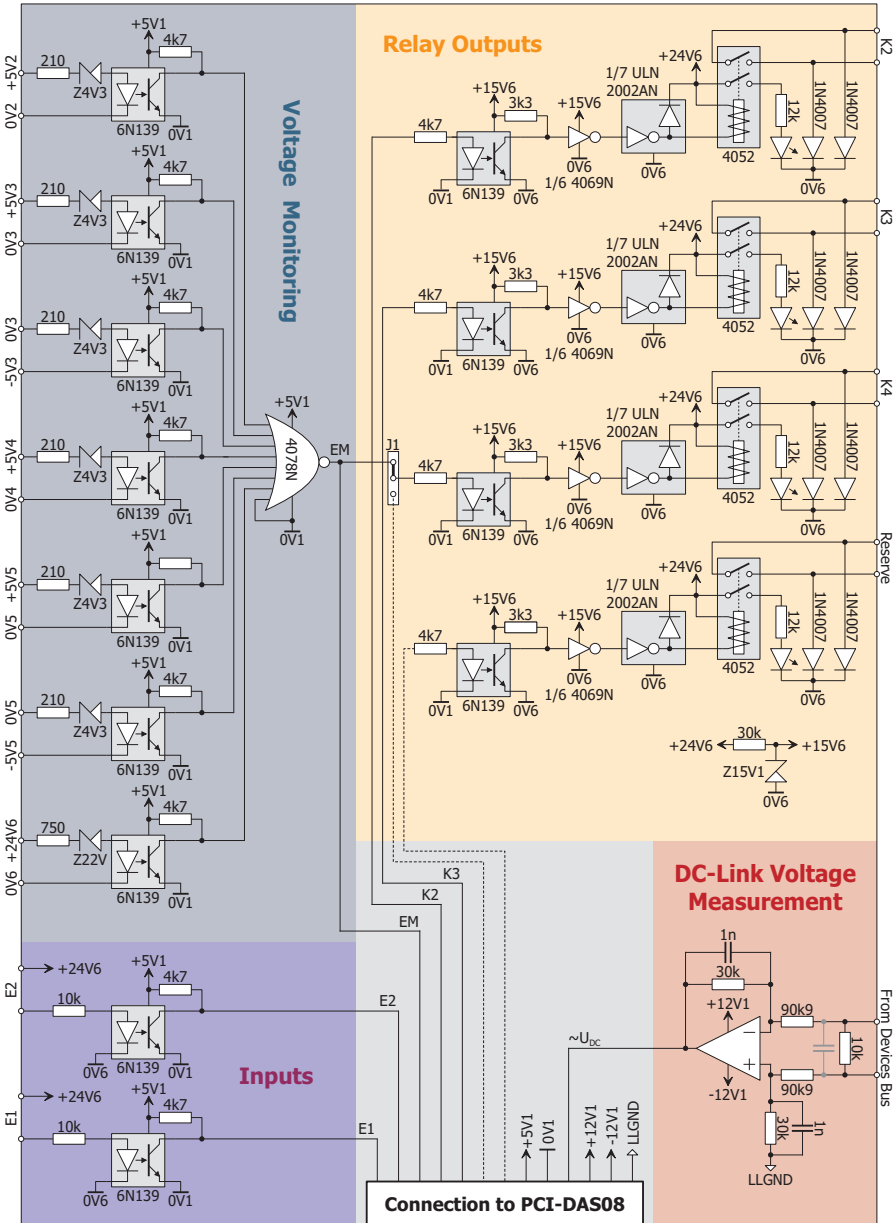
Figure 5.3: Diagram of the Suppy Control Board

5.4 Real-Time Module Structure

The real-time module is compiled from the eleven source files listed below. Their detailed structure, including the defined functions and the main data structures, is given in Figure 5.4, Figure 5.5 and Figure 5.6.

- **rtmodule.c**   Main file; initialisation / cleanup and the ISR
- **monitoring.h**   Implements the Monitoring State Machine
- **capture.h**   Implements the data capture functionality
- **control.h**   Here is implemented the machine control algorithm

rtmodule.c

Headers for Linux kernel functions
```
#include <linux/kernel.h>
#include <linux/module.h>
#include <linux/pci.h>
```

Headers for RTAI functions
```
#include <rtai.h>
#include <rtai_sched.h>
#include <rtai_fifos.h>
#include <rtai_math.h>
```

Vendor IDs and Device IDs of the three PCI Interface Boards - used for their identification
```
#define VCI_BRD_VID    0x1172
#define VCI_BRD_DID    0x0004
#define POS_BRD_VID    0x1172
#define POS_BRD_DID    0x0005
#define DAS08_BRD_VID  0x1307
#define DAS08_BRD_DID  0x0029
```

The two FIFOs used (command and data)
```
#define FIFO_CMD   0
#define FIFO_DATA  1
```

Definition of the commands of the Command FIFO
```
#define CMD_SET_PARAMS      1
#define CMD_SET_CORRS       2
#define CMD_DC_LINK_POW     3
#define CMD_MACHINE_ON_OFF  4
#define CMD_CAPTURE         5
```

DC Link state definition
```
static enum {
    DC_LINK_DISABLE,
    DC_LINK_ENABLE
} dc_link_state;
```

Machine state definition
```
static enum {
    MACHINE_OFF, MACHINE_ON
} machine_state;
```

Communication state definition
```
static enum {
    COMM_OK, COMM_ERR_UI,
    COMM_ERR_DAS08, COMM_ERR_S2D
} comm_state;
```

```
#include "monitoring.h"

#include "control.h"

#include "capture.h"
```

Module initialisation
```
int init_module(void) {
    ...
}
```

Command FIFO read-handler
```
void get_ui_command (void) {
    ...
}
```

Interrupt Service Routine
```
static void isr (void) {
    ...
    monitoring ();
    machine_control ();
    data_capture ();
    ...
}
```

Module clean-up
```
void module_cleanup(void) {
    ...
}
```

monitoring.h
```
static int E1, E2, EM;
static int K2, K3;
```
Monitoring in- / outputs

```
static enum {
    M0, M1, ... M9
} mon_state;
```
Monitoring states definition

```
void monitoring (void) {
    ...
}
```
Monitoring state machine

control.h
```
#include "parameters.h"       ①

#include "transforms.h"       ②

#include "pi_control.h"       ③

#include "modulator.h"        ④

#include "position.h"         ⑤

static double u_dc;
```

```
void ctrl_init(void) {
    ...
}
```
Initialisation of the control parameters

```
void machine_control (void) {
    ...
}
```
Machine control function: implements the control algorithm

capture.h
```
#define DATA_LEN 30
static double capture_data[DATA_LEN];
```
Capture buffer def.

```
static unsigned long capture_length;
static unsigned capture_decimation;
```
Capture parameters

```
void data_capture (void) {
    ...
    rtf_put (FIFO_DATA,
             &capture_data,
             sizeof(capture_data));
    ...
}
```
Data capture function: the capt. buffer is written in Data FIFO

Figure 5.4: Real-time module detailed structure (1/3)

- **`parameters.h`** — Data structures containing the machine parameters
- **`transforms.h`** — Axis transformations functions
- **`pi_control.h`** — PI controllers (currents, speed)
- **`modulator.h`** — Calculation of the switching states and times (SV-PWM)

Figure 5.5: Real-time module detailed structure (2/3)

- **position.h**   Synchronisation function; includes the two headers below
- **s2d.h**   Position and speed given by the optical sensors system
- **sls_emf.h**   EMF-based, sensorless position and speed estimation

⑤

position.h

```
#include "parameters.h"                    ⑥
#include "s2d.h"
#include "slsemf.h"
```

Position and speed used in control
```
static double x_c;
static double v_c;
```

Synchronisation function (between optical sensors' and observed position and speed)
```
void synchronisation (
    double x_s2d, double v_s2d
    double x_emf, double v_emf
    double *x_c,  double *v_c) {
    ...
}
```

s2d.h

Position Interface Board addresses definitions
```
#define POS_BRD_RESET 0x00
#define POS_BRD_D0    0x04
#define POS_BRD_STAT  0x20
#define POS_BRD_D1D2  0x24
#define POS_BRD_D3D4  0x28
#define POS_BRD_D5D6  0x2C
```

Sensors' period
```
#define S2D_PITCH (40e-6)
```

```
#define S2D_OFFS_3B (0.500135356)
```

Period correction tables
```
static double CORR_1A[5001][4];
static double CORR_2B[5001][4];
static double CORR_3A[2501][4];
static double CORR_3B[2505][4];
```

Head numbers definition (the states of the Reconstruction S.M.)
```
typedef enum {
    S2D_H0,
    S2D_H1A, S2D_H2B,
    S2D_H3A, S2D_H3B
} t_head_nr;
```

Definition of one read-head's data
```
typedef struct {
    int use, muxcode, counter;
    double sine, cosine;
    double x_inc, x_abs;
} t_head_data;
```

Definition of the Sensor to Digital data structure, used to determine sensors' position and speed (including synchronisation between read-heads)
```
typedef struct {
    t_head_data HA_data;
    t_head_data HB_data;
    t_head_nr current_head;
    short x_is_valid;
    double x0;
    double x;
    double v;
    void (*update)(void *);
} t_s2d_data;
static t_s2d_data s2d = {
    ...
    (void*) s2d_update
};
```
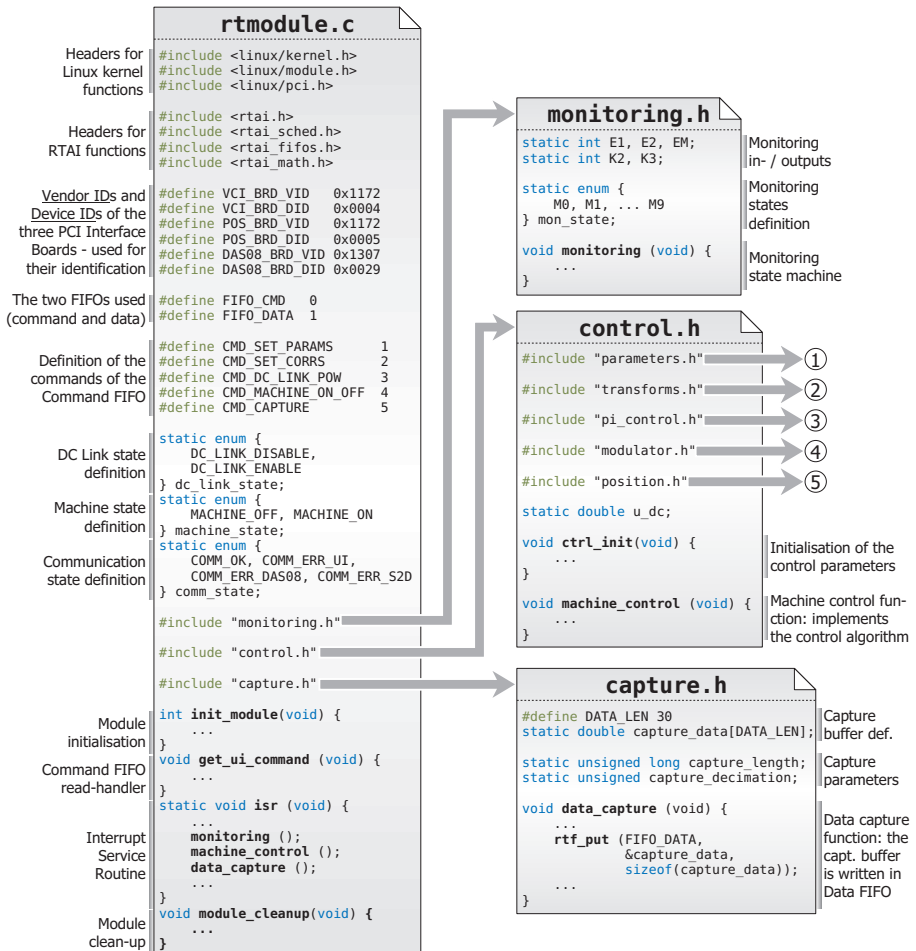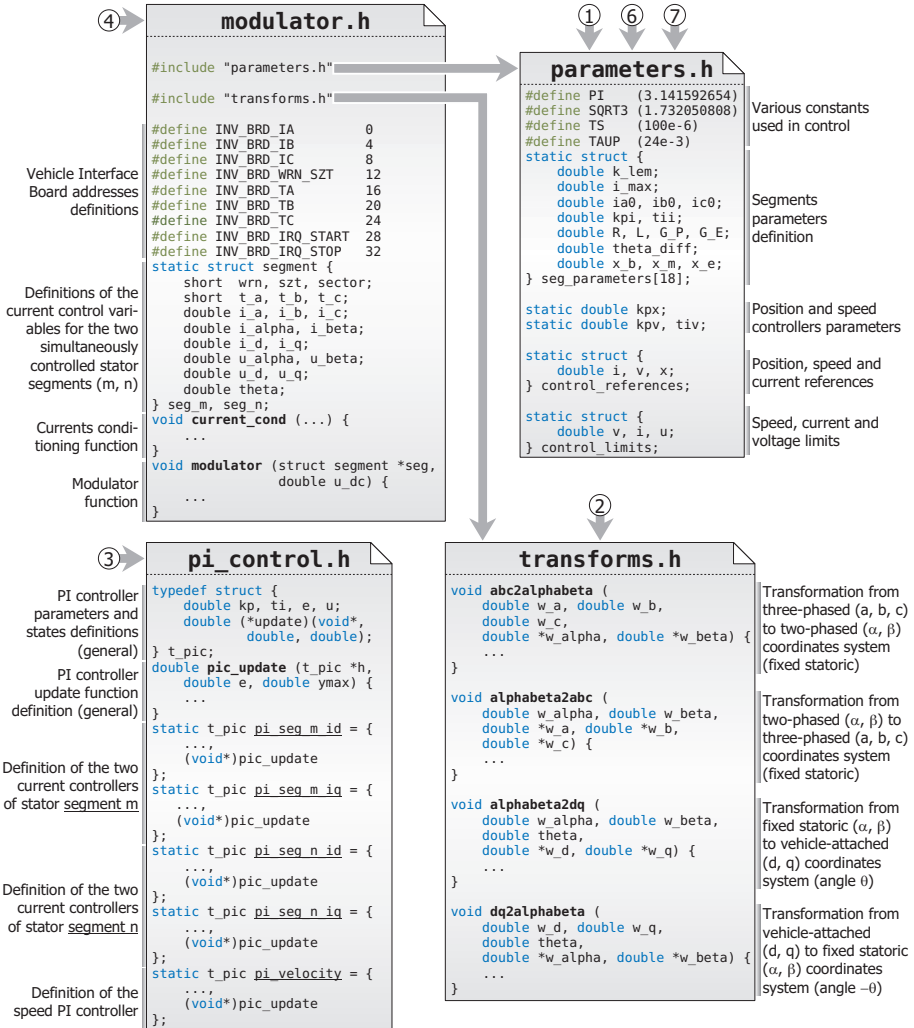
Function used to calc. a read-head's local position
```
void s2d_calc_head_x (...) {
    ...
}
```

Sensors' position and speed update function
```
void s2d_update (...) {
    ...
}
```

sls_emf.h

```
#include "parameters.h"                    ⑦
```

EMF observer definition (general)
```
typedef struct {
    double psi_alpha, psi_beta;
    double emf_alpha, emf_beta;
    void (*update)(void *, ...);
} t_emfobs;
```

EMF observer update function
```
void emfobs_update (t_emfobs *h, ...) {
    ...
}
```

Definition of the two EMF observers for the two controlled stator segments (m and n)
```
static t_emfobs emfobs_m = {
    ... ,
    (void*) emfobs_update
};
static t_emfobs emfobs_n = {
    ... ,
    (void*) emfobs_update
};
```

Mechanical observer definition (general)
```
typedef struct {
    double M, B;
    double G_F, G_v, G_x;
    double eps_x, FL, v, x;
    void (*update)(void *, ...);
} t_mecobs;
```

Mechanical observer update function
```
void mecobs_update (t_mecobs *h, ...) {
    ...
}
```

Mechanical observer definition
```
static t_mecobs mecobs = {
    ... ,
    (void*) mecobs_update
};
```
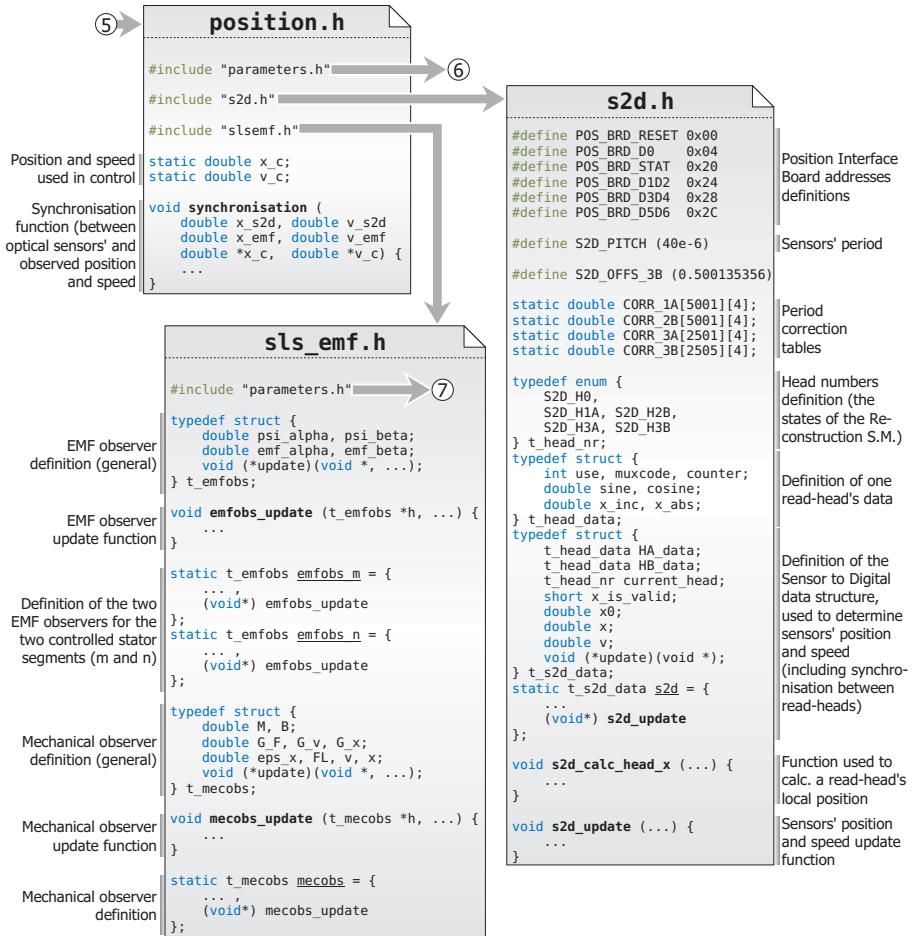
Figure 5.6: Real-time module detailed structure (3/3)

5.5 Mounting of the Optical Sensors at the Linear Drive for Material Handling

The distance between the vehicle-attached scale and the stationary read-head is not drawn to scale.

Figure 5.7: Read-heads mounting (only one read-head shown)

Adjustable distance between the collision protection plate and the load carrying plate on the vehicle
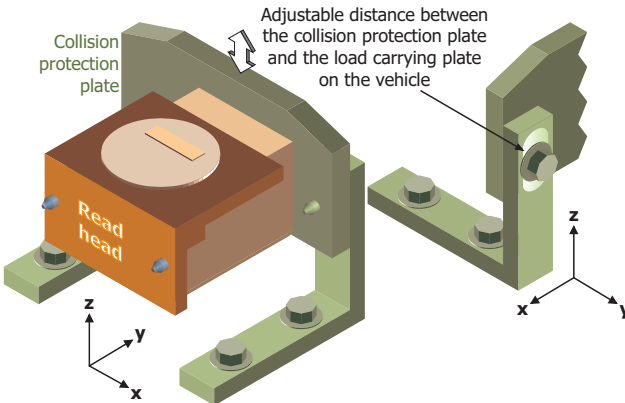
Figure 5.8: Collision protection

5.6 Measurement of the EMF constants

In order to determine the EMF constants, the following measurement was conducted for each segment of the linear machine: with the segment disconnected from its power stack, the vehicle was moved, by hand, inside the segment, while the voltages induced in the three phases were acquired using a storage oscilloscope.

Figure 5.9 (a) shows the result of such a measurement (the induced phase voltages e_a, e_b, and e_c), for stator segment 3.
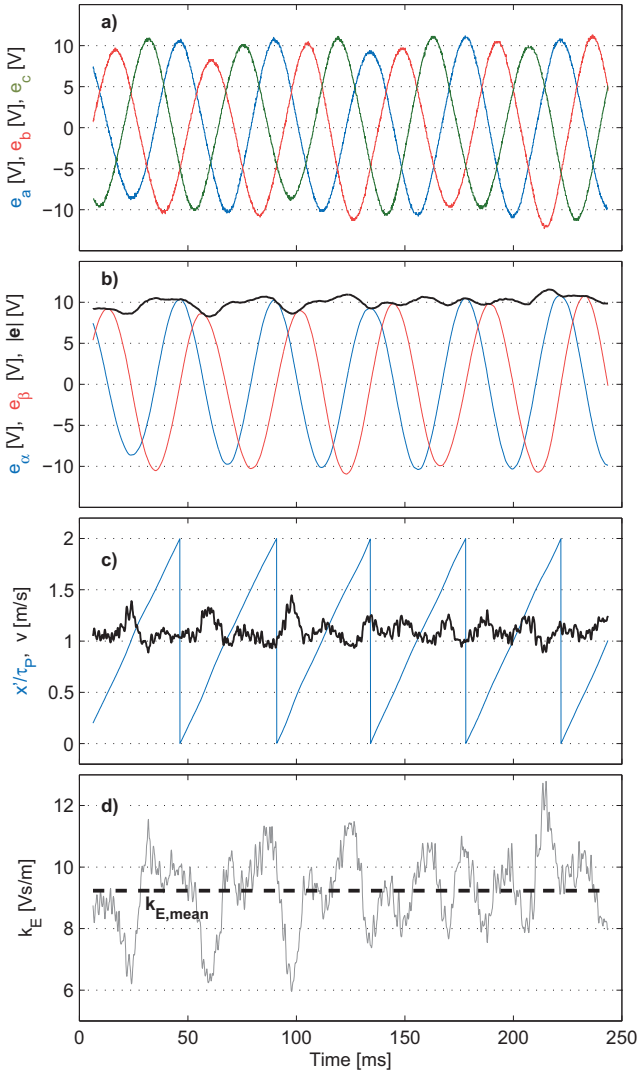

Figure 5.9: Determination of the EMF constant: a) Measured EMF in the three phases of Segment 3; b) The α–β components of the EMF, and the EMF module; c) Position and speed; d) EMF constant

After filtering the high-frequency noise, the e_α and e_β components of the EMF were calculated using the Clarke transform:

$$\begin{cases} e_\alpha = \dfrac{1}{3}(2 \cdot e_a - e_b - e_c) \\ e_\beta = \dfrac{1}{\sqrt{3}}(e_b - e_c) \end{cases} .$$

The module of the EMF vector – see Figure 5.9 (b) – was then determined:

$$|e| = \sqrt{e_\alpha^2 + e_\beta^2} .$$

For calculating the EMF constant, the speed information is also required. First, the position x′ was determined, using the phase information contained in the measured EMF:

$$x' = \frac{\tau_P}{\pi} \operatorname{atan2}\left(\frac{e_\alpha}{e_\beta}\right)$$

Then, the speed v was calculated as the numerical derivative of the position:

$$v[i] = \frac{x'[i] - x'[i-1]}{T_S} ,$$

where T_S represents the sampling interval (the time between samples i – 1 and i). The resulting position (modulo τ_P) and speed are shown in Figure 5.9 (c).

Finally, the mean value of the ratio between the EMF module and speed, over the N acquired samples, was calculated – Figure 5.9 (d) – yielding the mean value of the EMF constant:

$$K_{E,mean} = \frac{1}{N} \sum_{i=1}^{N} \frac{|e[i]|}{v[i]} .$$

Table 5.1 summarises the measured EMF constants for all 18 stator segments.

Segm.	K_E [Vs/m]	Segm.	K_E [Vs/m]	Segm.	K_E [Vs/m]
1	17,72	7	6,26	13	9,31
2	16,75	8	7,52	14	10,36
3	9,21	9	7,02	15	8,12
4	7,60	10	12,50	16	8,63
5	8,01	11	12,00	17	9,28
6	9,41	12	10,22	18	9,54

Table 5.1: Measured EMF constant for all stator segments

Bibliography

[1]   Hellinger R., and Mnich P., "Linear Motor-Powered Transportation: History, Present Status, and Future Outlook," Proceedings of the IEEE, vol. 97, no. 11, pp. 1892-1900, Nov. 2009

[2]   Gustafsson J., "Vectus—Intelligent Transport," Proceedings of the IEEE, vol. 97, no. 11, pp. 1856-1863, Nov. 2009

[3]   Bosshard C., "MagneTrak© Paradigmenwechsel im Materialhandling," Proc. of 13th Deutscher Materialfluss-Kongress Innovative Techniken für die Logistik, vol. 1815, pp. 189-200, Garching, Germany, 2004

[4]   Stöppler G., "Segmentierte Langstator-Linearmotoren für die Schnellpositionierung von Werkstückträgern in längsverketteten Montagelinien," in SPS / IPC / DRIVES / Electric Automation Systems and Components, Nuremberg, Germany, 2008

[5]   Mutschler P., "Comparison of topologies for linear drives in industrial material handling and processing applications," in 7th International Conference on Power Electronics, 2007 ICPE '07, pp. 1027-1032, Oct. 2007

[6]   Nyce D.S., "Linear Position Sensors: Theory and Application," ISBN 978-0-471-23326-8, John Wiley & Sons, 2004

[7]   Reininger T., Welker F., and von Zeppelin M., "Sensors in position control applications for industrial automation," Sens. Actuators A, Phys., vol. 129, no. 1/2, pp. 270-274, May 2006

[8]   Bähr A., "Speed Acquisition Methods for High-Bandwidth Servo Drives," Dissertation, TU Darmstadt, 2004

[9]   Canders W.-R., Mutschler P., Mosebach H., Shi Z., Weigel J., and Lamsahel H., "Neue Funktionalitäten von Linearantrieben," Forschungsvereinigung Antriebstechnik e.V., Forschungsheft 692, Jan. 2003

[10]   Rashed M., MacConnell P. F. A., Stronach A. F., and Acarnley P., "Sensorless indirect-rotor-field-orientation speed control of a permanent-magnet synchronous motor with stator-resistance estimation," IEEE Trans. Ind. Electron., vol. 54, no. 3, pp. 1664-1675, Jun. 2007

[11]   Seok J.-K., Lee J.-K., and Lee D.-C., "Sensorless speed control of nonsalient permanent-magnet synchronous motor using rotor position-tracking PI controller," IEEE Trans. Ind. Electron., vol. 53, no. 2, pp. 399-405, Apr. 2006

[12]   De Angelo C., Bossio G., Solsona J., Garcia G., and Valla M. I., "A rotor position and speed observer for permanent magnet motor with non sinusoidal EMF waveform," IEEE Trans. Ind. Electron., vol. 52, no. 3, pp. 807-813, Jun. 2005

[13]   Robeischl E., and Schroedl M., "Optimized INFORM measurement sequence for sensorless PM synchronous motor drives with respect to minimum current distortion," IEEE Trans. Ind. Appl., vol. 40, no. 2, pp. 591-598, Mar./Apr. 2004

[14]   Briz F., Degner M. W., Garcia P., and Lorenz R. D., "Comparison of saliency-based sensorless control techniques for AC machines," IEEE Trans. Ind. Appl., vol. 40, no. 4, pp. 1107-1115, Jul./Aug. 2004

[15]   Consoli A., Scarcella G., and Testa A., "Industry application of zero-speed sensorless control techniques for PM synchronous motors," IEEE Trans. Ind. Appl., vol. 37, no. 2, pp. 513-521, Mar./Apr. 2001

[16]   K. Patel, S. D. D'Arco, A. Monti, D. Patterson, and R. Dougal, "Design and testing of a modular permanent magnets brushless linear drive," in Proc. 20th Annual IEEE Applied Power Electronics Conference and Exposition, 2005, APEC 2005, vol. 3, pp. 1883-1888, March 2005

[17]   Perreault B. M., "Optimizing Operation of Segmented Stator Linear Synchronous Motors," Proceedings of the IEEE, vol. 97, pp. 1777-1785, Nov. 2009

[18]   Yoshida K., Takami H., and Fujii A., "Smooth section crossing of controlled-repulsive PM LSM vehicle by DTC method based on new concept of fictitious section," IEEE Trans. Ind. Electron., vol. 51, pp. 821-826, Aug. 2004

[19]   Seki K., Watada M., Torii S., and Ebihara D., "Discontinuous arrangement of long stator linear synchronous motor for transportation system," in Proc. 1997 International Conf. on Power Electronics and Drive Systems, vol. 2, pp. 697-702, May 1997

[20]   Suzuki K., Kim Y.-J., and Dohmeki H., "Proposal of the section change method of the stator block of the discontinuous stator permanent magnet type linear synchronous motor aimed at long-distance transportation," in Proc. of 18th International Conference on Electrical Machines, 2008, ICEM 2008., pp. 1-6, Sept. 2008

[21]   MTS Sensors, "Temposonics® absolute, berührungslose Positionssensoren. R-Serie Katalog," http://www.mtssensor.de/fileadmin/medien/downloads/datasheets/rserie_katalog_d.pdf (Link retrieved in June 2010)

[22]   MTS Sensors, "Magnetostriktion physikalische Grundlagen," http://www.mtssensor.de/ fileadmin/medien/downloads/mts_messprinzip.pdf (Link retrieved in June 2010)

[23]   Rettenmaier T., "Positionserfassung und Kommunikation zwischen zwei DSPs in modularen Servoantriebssystemen," Diploma Thesis Nr. 1349, Institut für Stromrichtertechnik und Antriebsregelung, TU Darmstadt, 2009

[24]   Texas Instruments, "The RS-485 Design Guide," Application Report, Feb. 2008, http://focus.ti.com/lit/an/slla272b/slla272b.pdf (Link retrieved in June 2010)

[25]   PCI Special Interest Group (PCI SIG), "PCI Local Bus Specification, Revision 2.3," March 2002, http://www.pcisig.com/specifications/conventional/ (Link retrieved in June 2010)

[26]   Dr. Johannes Heidenhain GmbH, "Exposed Linear Encoders," Specifications, July 2003

[27]   Linear Technology, "LTC1410 – 12-Bit, 1.25Msps, Sampling A/D Converter with Shutdown," Datasheet, 1995, http://cds.linear.com/docs/Datasheet/1410fa.pdf (Link retrieved in June 2010)

[28]   Bähr A. and Mutschler P., "Systematic Error Correction Methods for Sinusoidal Encoders and their Application in Servo Control," in Proc. of 10th European Conference on Power Electronics and Applications, EPE 2003, Sept. 2003

[29]   Höscheler B., "Erhöhung der Genauigkeit bei Wegmeßsystemen durch selbstlernende Kompensation systematischer Fehler", in Proc. of SPS / IPC / DRIVES 1999, pp. 617-626, Nov. 1999

[30] Bünte A. and Beineke S., "High-Performance Speed Measurement by Suppression of Systematic Resolver and Encoder Errors," IEEE Trans. on Ind. Electron., vol. 51, no. 1, pp. 49-53, Feb. 2004

[31] Netzer Precision Motion Sensors, "LE^{2}C Linear Encoder With Cable-free Read Head," Specifications, March 2004

[32] Netzer Y., "Linear, Absolute-position Output, Cable-free Electric Encoder," in Proc. of Power Conversion Intelligent Motion Conf., PCIM 2003, pp. 135-139, May 2003

[33] Mihalachi, M., "Characterization of a Capacitive Incremental Encoder and Design of an Improved Signal Handling Algorithm," Diploma Thesis Nr.1316, Institut für Stromrichtertechnik und Antriebsregelung, TU Darmstadt, 2004

[34] Li X., De Jong G., and Meijer G. C. M., "The influence of electric-field bending on the nonlinearity of capacitive sensors". IEEE Transactions on Instrumentation and Measurement, vol. 49, no. 2, pp. 256-259, April 2000

[35] VDE/VDI – Gesellschaft Mikroelektronik, Mikro- und Feinwerktechnik (GMM) "VDE/ VDI – Schulungsblätter für die Leiterplattenfertigung," 3711, Blatt 2, Mai 1999

[36] Netzer Y., "Capacitive Displacement Encoder," U.S. Patent No. 6,492,911, Dec. 2002, http://www.freepatentsonline.com/6492911.html (Link retrieved in June 2010)

[37] Texas Instruments, "OPA627 / OPA637 Precision High-Speed Difet® Operational Amplifiers," Datasheet, Literature Number SBOS165, March 1998, http://focus.ti.com/docs/prod/folders/print/opa627.html (Link retrieved in June 2010)

[38] Baxter L. K., "Capacitive Sensors," IEEE Press, New York, ISBN: 0-7803-1130-2, 1997, pp. 190-191

[39] Texas Instruments, "[...], TMS320F2812, [...] Digital Signal Processors," Data Manual, Literature Number SPRS174R, April 2001, http://focus.ti.com/docs/prod/folders/print/tms320f2812.html (Link retrieved in June 2010)

[40] Texas Instruments, "UAF42 Universal Active Filter," Datasheet, Literature Number SBFS002A, Nov. 2007, http://focus.ti.com/docs/prod/folders/print/uaf42.html (Link retrieved in June 2010)

[41] Mathworks Inc., "filtfilt – Zero-phase digital filtering," MATLAB online function reference, http://www.mathworks.com/access/helpdesk/help/toolbox/signal/filtfilt.html (Link retrieved in June 2010)

[42] Benavides R., and Mutschler P., "Controlling a System of Linear Drives," in IEEE 36th Power Electronics Specialists Conference, 2005, IEEE-PESC 2005, pp 1587-1593, June 2005

[43] Canders W.-R., Hoffmann J., Maurus Q., and Mosebach H., "Modular linear drive system with soft magnetic composite elements for 3D tracks," in 7th International Symposium on Linear Drives for Industry Applications, LDIA '09, 2009

[44] Siemens AG, "Simodrive Umrichter Projektierungsanleitung," Manufacturer/Service Documentation, Aug. 2002

[45] Benavides R., "Entwurf und Realisierung eines RS-485-basierten Multimasterbussystems," Master Thesis Nr. 1307, Institut für Stromrichtertechnik und Antriebsregelung, TU Darmstadt, 2003

[46]   Measurement Computing Corporation, "PCI-DAS08 Analog input and Digital I/O User's Guide," http://www.mccdaq.com/PDFs/manuals/pci-das08.pdf (Link retrieved in June 2010)

[47]   "Vector Linux 5.8 documentation," http://vectorlinux.osuosl.org/docs/vl58/manuals/index.html (Link retrieved in June 2010)

[48]   "Linux Kernel Documentation Index," http://kernel.org/doc/ (Link retrieved in June 2010)

[49]   Kroah-Hartman G., "Linux Kernel in a nutshell," ISBN 978-0-596-10079-7, O'Reilly Media Inc., 2007, pp. 127

[50]   Racciu G., and Mantegazza P., "RTAI 3.4 User Manual, rev 0.3," Oct. 2006

[51]   Dozio L., and Mantegazza P., "Real Time Distributed Control Systems using RTAI," in Proceedings of the 6th IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, ISORC '03, pp. 11-18, May 2003

[52]   Schwebel R., "Echtzeit unter Linux mit RTAI," Elektronik, No. 7/2002, pp. 72-77

[53]   "Glade – A User Interface Designer," http://glade.gnome.org/ (Link retrieved in June 2010)

[54]   "The GTK+ Project," http://www.gtk.org/index.php (Link retrieved in June 2010)

[55]   Intel Corporation, "Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3A: System Programming Guide, Part 1," pp. 221-241, March 2010 http://www.intel.com/Assets/PDF/manual/253668.pdf (Link retrieved in June 2010)

[56]   Cupertino F., Giangrande P., Salvatore L., and Pellegrino, G., "Sensorless position control of permanent magnet motors with pulsating current injection considering end-effect," Energy Conversion Congress and Exposition, 2009, ECCE 2009, pp. 1954-1961, Sept. 2009

[57]   Leidhold R., and Mutschler P., "Specific features of Position-Sensorless Methods in Synchronous Linear Motors," in Proc. of the 2010 International Exhibition & Conf. for Power Electronics, Intelligent Motion, Power Quality, PCIM 2010, pp. 124-129, May 2010

[58]   Tomita M., Senjyu T., Doki S., and Okuma S., "New sensorless control for brushless DC motors using disturbance observers and adaptive velocity estimations," IEEE Trans. Ind. Electron., vol. 45, no. 2, pp. 274-282, Apr 1998

[59]   Leidhold R., and Mutschler P., "Speed sensorless control of a long-stator linear synchronous motor arranged in multiple segments," IEEE Trans. Ind. Electron., vol. 54, no. 6, pp. 3246-3254, Dec. 2007

[60]   Kim J.-S., and Sul S.-K., "New approach for the low-speed operation of PMSM drives without rotational position sensors," IEEE Trans. Power Electron., vol. 11, no. 3, pp. 512-519, May 1996

[61]   De Angelo C., Bossio G., Solsona J., Garcia G. O., and Valla M. I., "A rotor position and speed observer for permanent-magnet motors with nonsinusoidal EMF waveform," IEEE Trans. Ind. Electron., vol. 52, pp. 807-813, June 2005

[62]   Leidhold R., and Mutschler P., "Speed sensorless control of a long-stator linear synchronous-motor arranged by multiple sections," Proc. of 31st Annual Conference of IEEE Industrial Electronics Society, 2005, IECON 2005, Nov. 2005

Papers published in peer-reviewed conference proceedings:

[63]   Mihalachi M., Leidhold R., and Mutschler P., "Long Primary Linear Drive for Material Handling," accepted for publication in the IEEE Industry Applications Magazine.

[64]   Mihalachi M., Leidhold R., and Mutschler P., "Motion Control for Long Primary Linear Drives used in Material Handling," in Proc. of the 14th International Power Electronics and Motion Control Conference, EPE-PEMC 2010, 6-8 Sept. 2010

[65]   Mihalachi M., and Mutschler, P., "Capacitive Sensors for Position Acquisition of Linear Drives with Passive Vehicles," in Proc. of the 12th International Conference on Optimization of Electrical and Electronic Equipment, OPTIM 2010, pp. 673-680, 20-22 May 2010

[66]   Mihalachi M., Leidhold R., and Mutschler P., "Long Primary Linear Drive for Material Handling," in Proc. of the 12th International Conference on Electrical Machines and Systems, ICEMS 2009, 15-18 Nov. 2009

[67]   Mihalachi M., Leidhold R., and Mutschler P., "Linear Drive System for Combined Transportation and Processing of Materials," in Proc. of the 35th Annual Conference of the IEEE Industrial Electronics Society, IECON 2009, 3-5 Nov. 2009

[68]   Mihalachi M., Leidhold R., and Mutschler P., "Control of Segmented Long Primary Linear drives," in Proc. of the 7th International Symposium on Linear Drives for Industry Applications, LDIA 2009, 20-23 Sept. 2009

[69]   Mihalachi M., and Mutschler P., "Position Acquisition for Linear Drives: A Comparison of Optical and Capacitive Sensors," in Proc. of the 34th Annual Conference of IEEE Industrial Electronics Society, IECON 2008, pp. 2998-3005, 10-13 Nov. 2008

[70]   Mihalachi M., and Mutschler P., "Position Acquisition for Long Primary Linear Drives with Passive Vehicles," in Proc. of the 2008 IEEE Industry Applications Society Annual Meeting, IAS 2008, pp 1-8, 5-9 Oct. 2008

[71]   Mihalachi M., and Mutschler P., "Evaluation of Two Position Acquisition Systems for Permanent Magnet Linear Motors with Passive Movers," in Proc. of the 11th International Conference on Optimization of Electrical and Electronic Equipment, OPTIM 2008, pp. 37-44, 22-24 May 2008

Academic Profile

Marius Alexandru Mihalachi

Born in Câmpina, Romania, on August 29th, 1981.

2005 – 2010   Working as assistant at the Department of Power Electronics and Control of Drives, Technische Universität Darmstadt, Germany.

1999 – 2005   Study of Electrical Engineering and Computer Science at "Transilvania" University of Braşov, Romania, with specialisation in Automatics and Industrial Informatics.

1995 – 1999   Technical High School "Energetic" Câmpina, Romania, with specialisation in Informatics.