

UNIVERSITY OF GENOA



Laurea Specialistica in Ingegneria Elettronica

---

**Position Estimation for the LHC Collimation  
System Actuators**

---

Master's Degree Thesis of  
**Andrea Tacchetti**

Tutor

**Prof. Marco Storace**

Supervisors

**Dr. Alessandro Masi**

**Dr. Mark Butcher**

2011

**Alla Commissione di Laurea e Diploma**  
**Alla Commissione Tirocini e Tesi**

Sottopongo la tesi redatta dallo studente Andrea Tacchetti dal titolo *Position Estimation for the LHC Collimation System Actuators*. Ho esaminato, nella forma e nel contenuto, la versione finale di questo elaborato scritto, e propongo che la tesi sia valutata positivamente assegnando i corrispondenti crediti formativi.

**Il Relatore Accademico**  
Prof. Marco Storage

## Abstract

In this thesis the tuning and implementation of an Extended Kalman Filter (EKF) for a sensorless drive working with arbitrary hybrid stepper motors and long cables is considered. A method to tune the filter using a set of data acquired from the real system is proposed. From this data set, the system parameters and the EKF's covariance matrices are estimated.

The hardware and software implementation of the EKF in the drive is also described, with specific emphasis on the code optimization steps that are necessary to execute the filter at the desired sampling rate. Moreover the developed drive's data acquisition capabilities and the experimental testbench used for tuning and validating of the filter are discussed. Experimental results prove the effectiveness of the tuning method and the efficacy of its implementation.

## Acknowledgments

This thesis is the result of over a year of investigation conducted at CERN in the EN-STI-ECE section. The work presented here is only a tiny fraction of what my experience at CERN has been and what it has meant to me.

First of all, I would like to thank Alessandro Masi for believing in my potential and bringing me to CERN, where he has been a great boss, an outstanding advisor and from time to time, a most needed psychiatrist. I will never be able to thank him enough for the regard and esteem he showed in my respect. His encouragement and faith in my potential as a researcher has been constant and fundamental for the development of my thesis and the decisions that followed it.

Michele Martino introduced me to CERN and carefully lead me in my first steps in the project, it would not have been possible for me to write this thesis had it not been for Michele's patience and encyclopedic knowledge. I yet have to meet an engineer as eclectic as Michele and I wish to thank him for the time he dedicated me and the quality of the explanations.

I have to thank Sergio Batuca and Christophe Mitifiot for their remarkable work on the board's hardware. Their experience and dedication has been invaluable. This project could not be started, let alone lead to any result, if not thanks to their tireless work in the backstage.

I wish to thank Alessandro Danisi, Carlo Zannini, Giovanni De Michele, Antonio Pierno and Giovanni Spiezia for their enjoyable company and cherishing humor which has kept me going through the hard times (and winters) in Geneva.

Above all in the CERN community I want to thank Mark Butcher. His contribution to the work presented here is enormous and his knowledge of control systems left me stunned a few times a day on average. I am much in debt with Mark from an academic point of view, he has been available beyond any imaginable limit to explain me every bit and piece of the project I was struggling with. Yet this is not his biggest influence on my experience at CERN. I would like to thank Mark for making the office a place I always

looked forward to go to as well as cheering me up when things were not going the way I planned. I had some great time cycling, skiing, roaming in the snow and even rowing on a pair with him. Together with Joana, Mark has welcomed me in Switzerland in the best of ways and I want to thank him for his friendship and cheerfulness. He is one of the best team-mates I ever had, and I am sure he will be hard to beat for years to come.

Professor Marco Storace has time and again proven to be the academic advisor anyone would hope for. His insight and support have been fundamental to complete this work. I cannot thank him enough for his availability (in his academic role and beyond) as well as the appraise and respect he has shown for me many a time and which has been fundamental in my career.

Coming from Genova, a year in the Pays de Gex can feel very long; yet it has been a most enjoyable time for me. Good laughs were never missing mostly thanks to my flat-mate Iago Gonzales Tabares. Adventures were endless as we tried to make our ways in the French communities. Coffees and beers at the Charlie's pub will be greatly missed. I have to thank Iago for his friendship, table-tennis coaching and moral support. He has literally shed new light on many aspects of life and I am much in debt with him.

A MEng program is a long journey; mine has been a most fun one thanks to Vito, Edo, Francesco, Michele, Paolo, Ilir and Luca. They literally dragged me through long studying sessions and their friendship is among the most valuable things I got out of my experience at the University of Genoa.

Lastly my biggest thanks go to my family, for supporting me in every possible way in these years. They taught me to carefully listen to good advice to then turn around and do the exact opposite. They instilled in me the curiosity and eagerness for knowledge that have brought me around the world having so much fun.

# Contents

List of Figures	viii
List of Tables	x
Acronyms	xi
Introduction	1
<b>1 The LHC Collimation Project</b>	<b>3</b>
1.1 Circular Particle Accelerators . . . . .	5
1.1.1 Principles of operation . . . . .	5
1.2 Beam cleaning and collimation . . . . .	6
1.2.1 Beam features . . . . .	6
1.2.2 Design goals . . . . .	7
1.3 Mechanical design and actuation system . . . . .	8
1.3.1 Hybrid stepping motors . . . . .	9
<b>2 Problem Formulation</b>	<b>10</b>
2.1 Hybrid Stepping Motor Model . . . . .	11
2.1.1 Qualitative description and torque expression . . . . .	11
2.1.2 The motor electrical part description . . . . .	13
2.1.3 The complete model for a hybrid stepping motor . . . . .	14
2.1.4 Stepping modes and third harmonic correction . . . . .	15
2.2 The long cable model . . . . .	17
2.2.1 Long cable model . . . . .	17

---

2.3	The hardware prototype and the implemented control scheme	20
2.3.1	The TMS320F28335 DSP . . . . .	21
2.3.2	The Feedback Loop and Actuator implementation . .	22
2.3.3	The implemented control scheme . . . . .	23
<b>3</b>	<b>Open-loop observer based solution</b>	<b>25</b>
3.1	Discretization of the open-loop observer . . . . .	25
3.2	Digital controller design . . . . .	27
3.3	Software architecture . . . . .	29
3.4	Graphical User Interfaces . . . . .	32
3.4.1	CollimationDriverCONFIG . . . . .	32
3.4.2	CollimationDriverGUI . . . . .	32
<b>4</b>	<b>The Extended Kalman Filter based solution</b>	<b>36</b>
4.1	The Extended Kalman Filter . . . . .	36
4.1.1	The asymptotic observer . . . . .	37
4.1.2	The optimal transient observer . . . . .	38
4.1.3	The Kalman Filter algorithm . . . . .	40
4.1.4	The Extended Kalman Filter . . . . .	40
4.2	State space model of a hybrid stepping motor . . . . .	41
4.3	The proposed implementation scheme . . . . .	43
4.3.1	Motor-side current estimators . . . . .	45
4.3.2	Motor-side voltage estimator . . . . .	46
4.3.3	Integration in the existing software architecture . . .	49
4.4	Implementation details . . . . .	50
<b>5</b>	<b>Identification and Tuning procedure</b>	<b>52</b>
5.1	Input and measurement estimation . . . . .	54
5.2	Automatic parameter estimation and tuning procedure . . .	55
5.2.1	Details . . . . .	55
<b>6</b>	<b>Experimental results</b>	<b>62</b>
6.1	Open-loop observer based solution . . . . .	63

---

6.1.1	Position repeatability . . . . .	63
6.1.2	Electromagnetic emissions . . . . .	67
6.2	EKF-based solution results . . . . .	69
6.2.1	Numerical stability . . . . .	69
6.2.2	Execution time performances . . . . .	69
6.2.3	State tracking performances . . . . .	71
6.2.4	Identification and tuning procedure results . . . . .	77
<b>7</b>	<b>Conclusions and future directions</b>	<b>78</b>
7.1	Contributions . . . . .	78
7.2	Conclusions on the developed methods . . . . .	79
7.3	Conclusions on the added value for the considered application	80
7.4	Outlook . . . . .	81
	<b>Bibliography</b>	<b>82</b>

# List of Figures

1.1	Aerial view of the CERN area with SPS (inner circle) and LHC (outer circle) accelerators overdrawn. . . . .	4
1.2	An LHC dipole junction. . . . .	6
1.3	Section of a Collimator Assembly. . . . .	8
1.4	A Hybrid stepping motor. . . . .	9
2.1	Driver-Motor set up. . . . .	11
2.2	A cross section of a two phases hybrid stepping motor. . . . .	12
2.3	Equivalent circuit of the motor electrical part. . . . .	14
2.4	Phasor diagram for the half-step mode. . . . .	16
2.5	An example of the superposed ringing effect with a cable length of 360 m. Both $i_{mot_A}$ and $i_{drv_A}$ are shown. . . . .	18
2.6	Equivalent circuit of a transmission line. . . . .	19
2.7	Equivalent circuit of the motor's electrical part including iron losses. . . . .	20
2.8	The prototype board. . . . .	21
2.9	MOSFET H-bridge. . . . .	23
2.10	Control Scheme. . . . .	24
3.1	Regulator with anti-windup scheme. . . . .	28
3.2	Flowchart for the chosen implementation scheme. . . . .	30
3.3	Snapshot image of the <i>CollimationDriverCONFIG</i> GUI. . . . .	33
3.4	Snapshot image of the <i>CollimationDriverGUI</i> GUI. . . . .	34

4.1	Magnitude Bode diagram of $G_{cm}(s)$ for two extremes of cable length . . . . .	44
4.2	Magnitude of $G_{est}(z)$ 's poles over a range of cable lengths . . . . .	45
4.3	Distribution of the open-loop estimator reconstruction error. Errors are evaluated in [A]. . . . .	47
4.4	Low frequency cable model connected to motor phase . . . . .	47
4.5	Comparison of the frequency responses of $H_v(s)$ using the complete impedance expressions and the approximation, for two cable length extremes . . . . .	48
4.6	Block diagram for the complete system. . . . .	50
6.1	Qualitative evaluation of the third harmonic correction effects on the position repeatability. . . . .	65
6.2	Emission measurement test-bench. . . . .	67
6.3	Comparative plot for the emitted H-field spectrum. . . . .	68
6.4	Time-lines as measured experimentally programming the DSP to toggle digital lines at specific events. . . . .	70
6.5	Sample data set. . . . .	72
6.6	Measured (blue) and estimated (green) states. Training data set: 2. Validation data set: 1 (see table 6.4). . . . .	73
6.7	Measured (blue) and estimated (green) states. Zoom on external torque and position. . . . .	74

# List of Tables

2.1	Full step phase excitation sequence. . . . .	13
2.2	Used symbols for the motor model. . . . .	14
2.3	Used symbols for the transmission line model. . . . .	19
2.4	Texas Instruments <sup>TM</sup> TMS320F28335 Features . . . . .	22
3.1	Values and expressions for the controller constants, $l$ , $h$ , and $r$ were defined in table 2.3. . . . .	28
6.1	Static position repeatability benchmark without a cable. . . . .	66
6.2	Controller parameters. . . . .	66
6.3	Static position repeatability with the cable. . . . .	66
6.4	Position tracking performance <i>Montecarlo</i> results. . . . .	75
6.5	Position tracking performance data-set related results. . . . .	76
6.6	EKF optimal parameters for each data-set. . . . .	77

# Acronyms

---

RMSE	Root mean square error
RMS	Root mean square
DSP	Digital Signal Processor
PCB	Printed Circuit Board
FPU	Floating Point Unit
PWM	Pulse Width Modulation
RAM	Random Access Memory
EEPROM	Electrically-Erasable Programmable Read Only Memory
QEI	Quadrature Encoder Interface
I2C	Inter-Integrated Circuit
UART	Universal Asynchronous Receiver/Transmitter
GPIO	General Purpose Input/Output
IIR	Infinite Impulse Response
MMACS	Million Multiply Accumulate Cycles per Second
ADC	Analog to Digital Converter
RMSE	Root Mean Square Error
DAQ	Data Acquisition
IDE	Integrated Development Environment
VI	Virtual Instrument
LHC	Large Hadron Collider

# Introduction

The European Organization for Nuclear Research (CERN) is an international organization whose mission is to provide for collaboration among European States in nuclear research of a pure scientific and fundamental character. It was established in 1954, as one of the first European joint ventures. It has since been the scene of important scientific breakthroughs and discoveries.

Nuclear researchers nowadays are faced with exciting and peculiar engineering challenges to build the highly sophisticated machines and infrastructures needed for their experiments. Most of the technological return to society produced at CERN comes from these challenges. The World-Wide Web as well as the capacitive touch screen both first saw the light in the CERN facilities astride the French-Swiss border.

In 2000 the operation of the Large Electron-Positron collider (LEP) came to an end after more than 10 years. The construction of the Large Hadron Collider (LHC), the largest, most powerful and most expensive scientific instrument ever built, started in the very same year.

The LHC is a circular particle accelerator and, as for all machines of this kind, needs a collimation system to block particles flying off their trajectories. These particles are highly energetic and if they are not properly collected they can seriously damage the accelerator. A collimation system is the module that collects these potentially harmful particles. It comprises some moving parts that need to be actuated with high precision inside the tunnel. Stepper motors are used for this purpose.

It is impossible to locate electronic sensors inside the tunnel due to the very high radioactivity of the environment. This results in the unavailability of direct measurements of motor state components. Furthermore, indirect information of the motor state is available only in a very corrupted form.

In its 50 years history the Kalman filter has proven to be a workhorse of linear estimation and an overwhelming number of research papers have addressed its many properties and domains of applications. Nevertheless only a few of these papers cover its implementation and the tuning of its parameters.

This thesis concerns the application of an Extended Kalman filter to the position estimation of the LHC collimation system actuators. Some novel requirements and techniques will be presented that allow a remarkable state tracking performance to be achieved even in harsh environments such as particle accelerators. A particular attention will be devoted to an automatic procedure to automatically tune all the filter parameters and a number of implementation details. These results, despite being developed to solve a very specific problem, can find application in a variety of domains where it is impossible to locate sensors in the proximity of the actuators; these include high temperature environments, underwater and radioactive settings.

The thesis is organized as follows, the first chapter introduces the LHC collimation project and gives some details on the mechanics of the collimation system. Chapter 2 presents the details of the particular problem we wish to solve, in the next chapter the main system architecture is laid out and detailed and a possible solution that does not use an Extended Kalman filter is introduced. Chapter 4 presents a solution based on the Extended Kalman filter; after a brief summary of the filter's equations, the architecture presented in Chapter 3 is extended to include the closed loop observer. Chapter 5 presents the of the automatic parameter tuning procedure, the following chapter presents some experimental results and finally, Chapter 7 draws the conclusions of this thesis and highlights the main contributions.

# 1 The LHC Collimation Project

*We've learned from experience  
that the truth will come out.*

---

Richard P. Feynman

**The CERN accelerating complex is described along with its basic principle of operation. The LHC collimators and their actuation system are introduced.**

CERN is the European Organization for Nuclear Research, the world's largest particle physics centre. It sits astride the French-Swiss border near Geneva. CERN is a laboratory where scientists from all over the world, gathered in various international collaborations, study the building blocks of matter and the forces that hold them together.

Founded in 1954, the laboratory was one of Europe's first joint ventures and now includes 20 member states. The mission for this organization is to provide for collaboration among European countries in research of a pure and fundamental character.

Various machines enhanced the accelerating complex after the initial Proton Synchrotron (PS) accelerator. The SPS (Super Proton Synchrotron) provided enough energy to allow the discovery of the weak force bosons  $W_+$ ,  $W_-$ ,  $Z_0$  for which Prof. Carlo Rubbia and Eng. Simon Van der Meer were attributed the Nobel Prize in 1984. The LEP (Large Electron Positron), whose energy in the center of mass topped 209  $GeV$  in 2000, provided

high precision measurements for the aforementioned particles. The Large Hadron Collider (LHC), where the first collisions at  $7\text{ TeV}$  in the center of mass took place on 30th March 2010, is currently the most powerful particle accelerator ever built. It is a circular accelerator installed in the  $27\text{ km}$  long tunnel that hosted LEP until late 2000.



Figure 1.1: Aerial view of the CERN area with SPS (inner circle) and LHC (outer circle) accelerators overdrawn.

The LHC accelerates two proton beams travelling in opposite directions at almost the speed of light. Each beam is made of proton bunches, essentially long cylinders of a proton gas with a section of a few microns. The LHC is foreseen to be fully operational by 2012 when it will collide 2808 bunches per beam each of  $1.15 \times 10^{11}$  protons, reaching a nominal luminosity<sup>1</sup> of  $10^{34}\text{ cm}^{-2}\text{ s}^{-1}$  with a  $\beta^*$  of <sup>2</sup>  $0.5\text{ m}$  and an energy of  $14\text{ TeV}$  in the center of mass [1]. The two counter rotating beams are accelerated in Radio Frequency (RF) cavities, then steered and squeezed to collide at the four interaction points, where four detectors (ATLAS, CMS, ALICE and LHCb) are installed. The experiments are designed to study a wide range of

<sup>1</sup>The luminosity of a particle accelerator is the number of interactions per unit surface per unit time.

<sup>2</sup> $\beta^*$  is the distance from the interaction point where the beam section has twice the radius it will have at the interaction point. It is an important figure of performance for particle accelerators for a number of reasons.

fundamental physics: proof for the Higgs boson existence, Super Symmetry, the reason why in our universe we see more matter than anti-matter, and the physics of the Bottom-Quark are only a few of the questions the LHC may answer.

## 1.1 Circular Particle Accelerators

### 1.1.1 Principles of operation

The LHC, as all particle circular accelerators, uses electromagnetic fields to give a charged particle high kinetic energy (i.e. a big momentum in the direction of motion) and to keep it from drifting away from its closed orbit. A particle of charge  $q$  and momentum  $\mathbf{p}$  through an electromagnetic field is subject to the Lorentz's force:

$$\mathbf{F} = \frac{d\mathbf{p}}{dt} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B}) \quad (1.1)$$

The electric field  $\mathbf{E}$  changes the particle trajectory and speed so it can be used to accelerate and decelerate particles. A constant magnetic field  $\mathbf{B}$ , normal to the particle velocity, alters the trajectory leaving the modulus of the momentum unchanged. In the LHC, particles are accelerated in RF cavities with an electric field applied in the direction of motion, 1232 magnetic dipoles (see figure 1.2) guide the particles on the reference circular orbit applying a magnetic field that counteracts the centrifugal forces to ensure beam stability, 360 quadrupoles focus the beams to keep a proper intensity and size.

The bending dipole field intensity is determined by the desired orbit's curvature and particles' energy  $E$ . At the equilibrium, Lorentz's force must balance the centrifugal forces that are driving the particles away from their reference path. In the LHC, particle beams circulate practically at the speed of light  $c$  so the equilibrium condition results in:

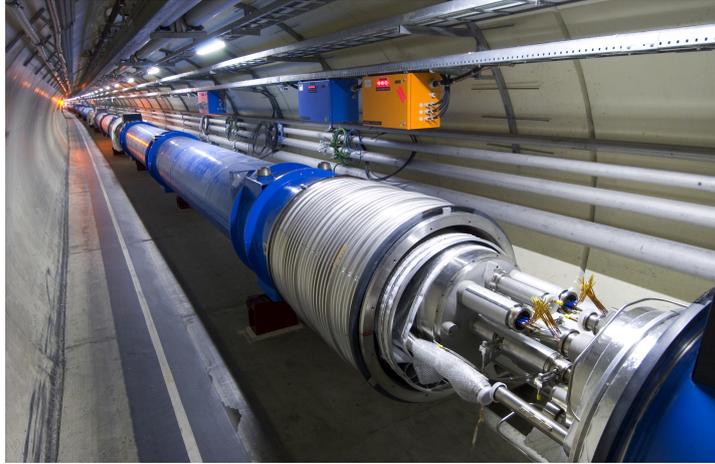


Figure 1.2: An LHC dipole junction.

$$|\mathbf{B}| = \frac{Ev}{\rho c^2 q} = \frac{E}{\rho c q} \quad (1.2)$$

An  $8.33 \text{ T}$  magnetic field is needed in order to have a beam with a curvature radius  $\rho$  of  $2803 \text{ m}$  an energy of  $7 \text{ TeV}$  per beam (the nominal desired energy for the LHC) in a machine that accelerates protons, with a charge  $q$  of  $1.6 \times 10^{-19} \text{ C}$ . A similar reasoning can be used to calculate the  $233 \text{ T/m}$  quadrupoles gradient necessary to achieve the equilibrium. These high field intensities can only be achieved with superconducting magnets.

## 1.2 Beam cleaning and collimation

### 1.2.1 Beam features

Particles traveling in a circular accelerator drift from their reference orbit. The magnets used to correct these errors present “field errors” and thus, despite this effort, particle deviation from their desired path is not negligible. This orbit excursion has to be limited to a RMSE of  $4 \text{ mm}$  for the LHC. A larger error could lead to serious damages to the accelerator itself as highly energetic particles would crash into a part of the machine called

the injection ring. A natural spread in momentum results in a different interaction of each particle with the dipole field yielding an energy dispersion  $\Delta E$  whose ratio to the nominal energy must not exceed  $10^{-4}$ .

### 1.2.2 Design goals

Due to their unprecedented energy content, LHC beams are highly destructive. The superconducting magnets used for acceleration and bending could quench, at 7 *TeV*, if as little energy density as  $30 \text{ mJ/cm}^{-3}$  was laid on their coils. Nevertheless beam losses cannot be completely avoided. A “primary beam halo” is continuously filled by various processes and the beam lifetime is in fact finite.

An effective beam collimation system is therefore needed to guarantee the following features:

1. Effective cleaning of the beam halo during the whole LHC beam cycle.
2. Suppression of halo-induced backgrounds for the physics experiments.
3. Generation of an abort trigger if abnormal beam loss at the collimation point is detected.
4. Beam tails scraping and halo population diagnostics.
5. Cleaning of abort gap to avoid spurious quenches after normal beam dump operation.
6. Reliability and robustness both in normal and abnormal operation.

To fulfill these specifications, innovative technologies and materials such as carbon-carbon composites are used.

## 1.3 Mechanical design and actuation system

Collimators in the LHC need to be mechanically robust but at the same time long, and allow a very precise positioning. The basic idea used for the design is rather simple: two long carbon-carbon composite bars are moved, as a jaw, to tighten or widen the gap between them (the collimation gap), where the beam will pass. The minimum gap size of  $0.5\text{ mm}$  and the beam size of  $200\text{ }\mu\text{m}$  RMS at the collimators imply tight mechanical tolerances. The absolute opening of the collimation gap is safety critical and must be known with good precision at all times. Settings reproducibility is also necessary to avoid lengthy re-optimizations that cannot be afforded.

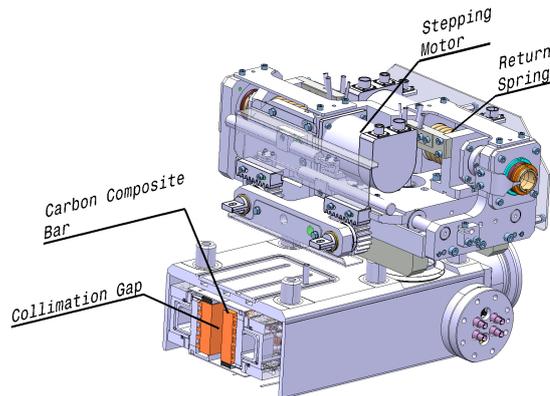


Figure 1.3: Section of a Collimator Assembly.

Each bar in the jaw is independently actuated by two stepping motors to allow lateral displacement (with a nominal stroke of  $30\text{ mm}$ ) and tilt adjustment. Each motor directly drives, via a roller screw/nut set, a table that provides the precise positioning of the jaw supporting axle. A return spring ensures a semi-automatic back driving in case of power supply failure.

### 1.3.1 Hybrid stepping motors

Hybrid stepping motors are used as actuators for the LHC collimators [2]. These are electrical motors without commutators. Windings are placed on the stator poles and a permanent magnet is mounted on the rotor. Teeth on the rotor provide a preferred path for the magnetic flux.

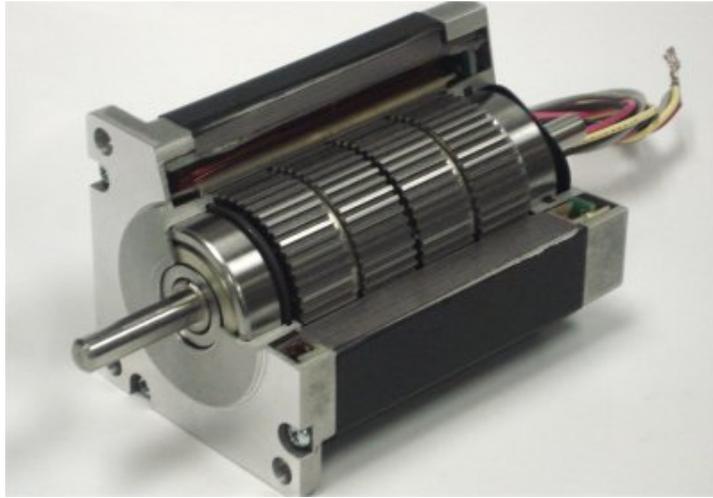


Figure 1.4: A Hybrid stepping motor.

Torque is produced in hybrid stepping motors exploiting the interaction between the magnetic field induced by the stator windings and the permanent magnet on the rotor.

## 2 Problem Formulation

*Engineers like to solve problems. If there are no problems handily available they will create their own problems.*

---

Scott Adams

**The system and its settings are presented. The motor model is introduced and the cable is described with its effects. The designed and realized driver is introduced along with an open-loop observer based solution.**

The LHC tunnel, where the collimators stepping motors are installed, is a highly radioactive environment; it is thus impossible to install the driver in situ. A vast literature exists on the damages caused by radiation to electronics [3], [4], [5], [6].

In general, to avoid phenomena such as *single event upsets*, *single transient upsets* or the undesired generation of low impedance paths between the power supply potential and ground, it is desirable to avoid the employment of non radiation-hard electronics in a radioactive environment. For these reasons it has been decided to locate the driver and its sensors in radioactivity-safe zones, at a distance that varies between 100 *m* and 1 *km* from the motor. Driver and motor phases are connected by long cables.

These cables act as a good source for electromagnetic radiation. The power spectrum of these emissions has to be negligible at low frequencies

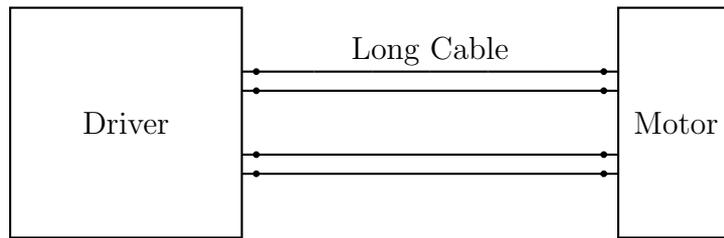


Figure 2.1: Driver-Motor set up.

to avoid causing major problems for neighbouring electronics. To achieve this requirement, the control action is delivered to the motor via a PWM signal, a standard technique for motor drives as explained in [7], [8] and [9], with a relatively high frequency for the rising edge:  $50\text{ kHz}$ .

This last solution leads to a new serious issue: the long cables act as transmission lines at these high PWM frequency, thus provoking an effect called *ringing* explained in section 2.2.

Referring to figure 2.1 our system is naturally split into three subsystems: the stepping motor, the cable and the driver.

## 2.1 Hybrid Stepping Motor Model

### 2.1.1 Qualitative description and torque expression

As mentioned before, the torque in hybrid stepping motors is produced by the interaction between the magnetic field induced in the stator windings with an exciting current and the permanent magnet in the rotor.

The controlled variables are the two currents that flow in the stator windings as shown in figure 2.2. By changing their amplitude and direction we can vary polarity and intensity of the induced magnetic flux. Let us focus exclusively on Pole 1 and 2 in the figure for the sake of simplicity.

When, as shown in the figure, a current is flowing counterclockwise in the winding of Pole 1, while Pole 2 winding is left open, Pole 1 becomes an S pole and the tooth pitches connected to the N pole of the rotor will tend

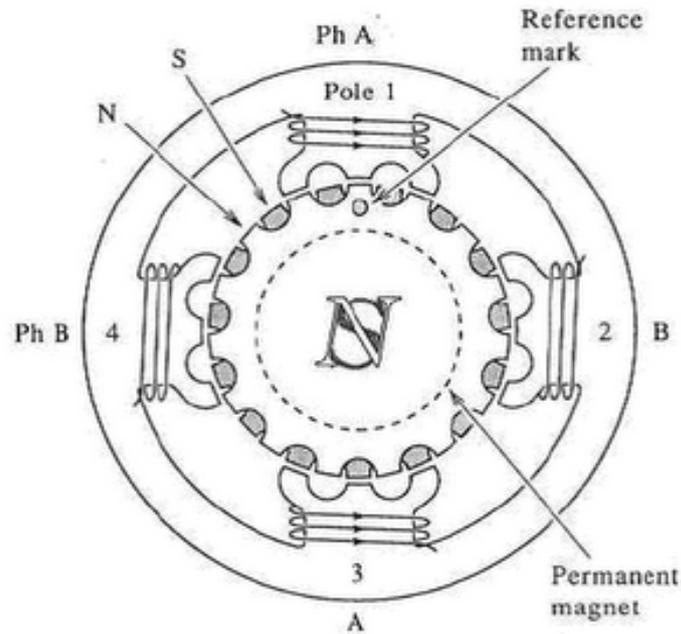


Figure 2.2: A cross section of a two phases hybrid stepping motor.

to align to it. Conversely, if from this latter configuration, we open Pole 1 winding and excite with a clockwise current Pole 2, the S pole rotor tooth pitches will align to Pole 2.

If we keep forcing this pattern for the motor phase currents we obtain the full-step excitation sequence reported in table 2.1. In case of no load torque, the shaft will move by one step (defined as the angular displacement between two subsequent tooth pitches) at each change in the phases' currents. A precise positioning is guaranteed, at least at steady state, by the presence of small tooth pitches on the rotor.

The intensity of the induced magnetic field is proportional to the amplitude of the current flowing through the phase and, therefore, so is the magnitude of the produced torque. The instantaneous value of the torque produced by each pole will also depend on the angular displacement between the stator poles and the rotor tooth pitches. More precisely, the produced electromagnetic torque is:

Position	$i_A$	$i_B$
1	$I_{max}$	0
2	0	$-I_{max}$
3	$-I_{max}$	0
4	0	$I_{max}$

Table 2.1: Full step phase excitation sequence.

$$\tau_{EM} = -K_m i_{mot_A} \sin(p\theta) + K_m i_{mot_B} \cos(p\theta) \quad (2.1)$$

where the constant  $K_m$  relates the phase current to the produced moment of force and is one of the motor parameters,  $\theta$  is the shaft angular position,  $p$  is the number of rotor pitches per quarter turn and  $i_{mot_A}$  and  $i_{mot_B}$  are the currents flowing in the stator poles.

The so called Cogging torque, that takes into account the detent moment, due to the interaction between the rotor magnet and the stator poles, when no current is flowing through the latter, has the following expression (see [10] and [11]):

$$\tau_{detent} = -T_{dm} \sin(2p\theta + \phi). \quad (2.2)$$

being  $T_{dm}$ , the detent torque constant, another motor characteristic figure.

To complete the expression for the torque, the shaft is subject to, we shall consider, a viscous friction term  $\tau_{vf} = -B\omega$  and a static friction term:  $\tau_{sf} = -C \text{sign}(\omega)$  where  $\omega$ . is the shaft angular velocity.

### 2.1.2 The motor electrical part description

The motor electrical part can be modelled as the  $RL$  circuit shown in figure 2.3, with input  $u_{mot_j}$  and a voltage generator  $e_j$  to model the Back Electromagnetic Force (Back EMF).

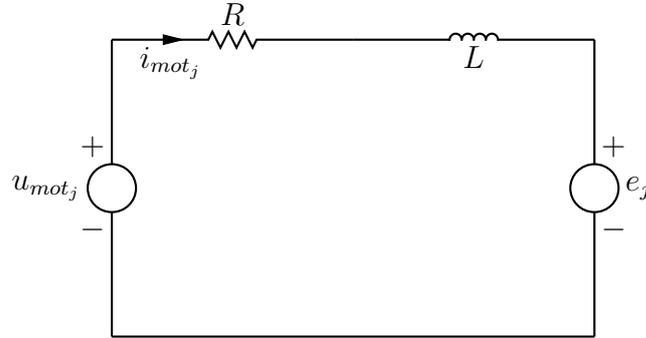


Figure 2.3: Equivalent circuit of the motor electrical part.

The Back EMF is due to the fact that our rotor is a magnet and therefore, from Maxwell's third equation, a sinusoidal voltage is induced in the motor phases when the shaft is moving.

### 2.1.3 The complete model for a hybrid stepping motor

Symbol	Description	Units
$i_{mot_j}$	Phase current	$A$
$R$	Phase winding resistance	$\Omega$
$L$	Phase winding inductance	$H$
$e_j$	Back EMF	$V$
$u_{mot_j}$	Control action	$V$
$J$	Rotor inertia	$Nms^2$
$B$	Viscous friction coefficient	$Nms$
$\omega$	Rotor angular velocity	$rad/s$
$T_{dm}$	Detent torque constant	$Nm$
$\theta$	Rotor angular position	$rad$
$\tau_L$	Load torque	$Nm$
$Km$	Electromagnetic torque constant	$Nm/A$
$C$	Static friction coefficient	$Nm$

Table 2.2: Used symbols for the motor model.

If we synthesize the expressions we have derived so far in this section

and solve the circuit shown in figure 2.3 for the motor phase current, we can derive the complete model for a hybrid stepping motor described by equations 2.3 and 2.4. All parameters and variables are summarized in table 2.2 with their units of measurement.

$$L \frac{di_{mot_j}}{dt} = -Ri_{mot_j} - e_j + u_{mot_j} \quad \text{for } j = A, B \quad (2.3)$$

$$\begin{aligned} J \frac{d\omega}{dt} &= -B\omega - T_{dm} \sin(2p\theta + \phi) - \tau_L \\ &\quad - K_m i_{mot_A} \sin(p\theta) + K_m i_{mot_B} \cos(p\theta) \\ &\quad - C \text{sign}(\omega) \\ \frac{d\theta}{dt} &= \omega \end{aligned} \quad (2.4)$$

with

$$\begin{aligned} e_A &= -K_m \omega \sin(p\theta) \\ e_B &= K_m \omega \cos(p\theta) \end{aligned}$$

The mechanical behaviour is described as the rotation of a rigid body of inertia  $J$  driven by a combination of the torques described above and subject to an unknown load torque  $\tau_L$ .

A more formal derivation of this description can be found in [11] and [10]. For a more exhaustive qualitative explanation of the principle of operation see [12] and [13].

#### 2.1.4 Stepping modes and third harmonic correction

Carefully adjusting the two phase currents it is possible to move a rotor tooth between two poles. An intuitive representation of the values needed for the two phase currents to achieve this configuration is the phasor diagram.

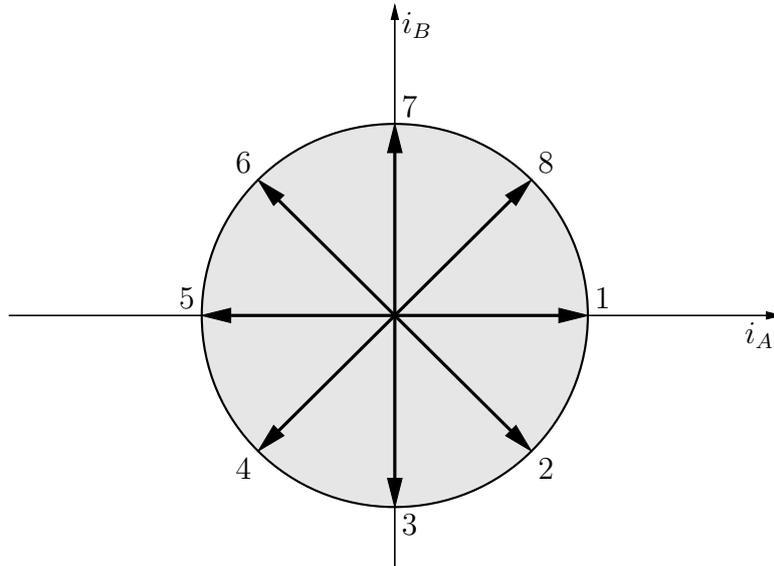


Figure 2.4: Phasor diagram for the half-step mode.

An example is shown in figure 2.4. From table 2.1 we can see that selecting only positions 1,3,5, and 7 and defining the circle radius as  $I_{max}$  we obtain the phasor diagram for the full step mode. Considering also the even positions, instead, we obtain a diagram for the so-called half step mode, where we are aligning the rotor teeth both to the stator poles and halfway between them. Obviously, adding more and more available configurations to the excitation scheme we can obtain the 1/4 step mode, the 1/8 step mode and so on.

Defining  $f_s$  as the stepping frequency, (i.e., the frequency at which a step request is generated), we can write the law to generate a reference for the two currents:

$$i_a(t) = I_{max}[\sin(2\pi m \frac{f_s}{4} t)] \quad (2.5)$$

$$i_b(t) = I_{max}[\cos(2\pi m \frac{f_s}{4} t)] \quad (2.6)$$

where  $m = 1/2, 1/4 \dots$  according to the stepping mode.

It has been experimentally shown that a better performance in terms of

positioning capabilities is achieved if this reference generating law is chosen instead:

$$i_a(t) = I_{max}[(1 - \alpha) \sin(2\pi m \frac{f_s}{4} t) + \alpha \sin(2\pi m 3 \frac{f_s}{4} t)] \quad (2.7)$$

$$i_b(t) = I_{max}[(1 - \alpha) \cos(2\pi m \frac{f_s}{4} t) + \alpha \cos(2\pi m 3 \frac{f_s}{4} t)] \quad (2.8)$$

Henceforth, we will refer to  $\alpha$  as the third harmonic correction.

## 2.2 The long cable model

As mentioned above, our driver delivers its control action through a PWM signal in a long cable. It also reads the feedback current on the driver side.

Although we are wishing to control the motor phase current  $i_{mot_j}$  we only have access to the driver side current  $i_{drv_j}$ . This current is the result of the superposition of two components: the motor side current and the ringing effect due to the PWM signal in the long cable. An example of this effect is presented in figure 2.5 where data acquired with two LEM-ClipOn Current Transducer and a LeCroy WaveRunner oscilloscope are presented.

### 2.2.1 Long cable model

To be able to control the motor side current, some kind of observer is needed for the controller to operate properly. In other words, we need an estimator for the motor side current  $i_{mot}$ , given observations of the driver side current  $i_{drv}$ .

$$[\hat{i}_{mot_j}(k) | i_{drv_j}(k-1) \dots i_{drv_j}(k-N)] \quad (2.9)$$

An open-loop estimator has been chosen for this specific task. More specifically, a linear digital IIR filter (so that  $N$  in equation 2.9 is the order of a linear filter) is used to approximate the transfer function  $G_{est}(z)$  between  $i_{drv_j}$  and  $i_{mot_j}$ , more explicitly:

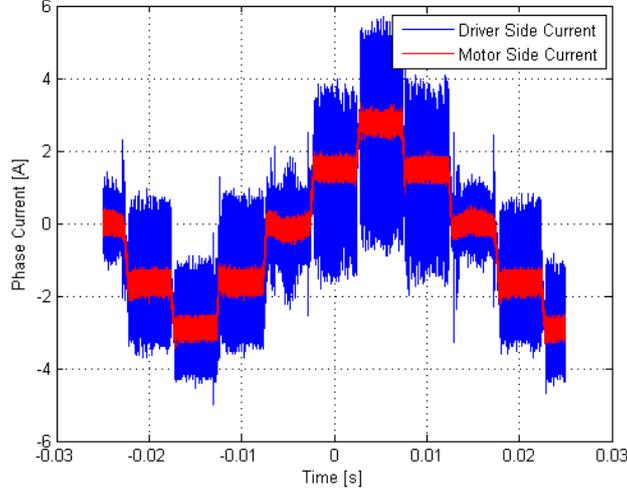


Figure 2.5: An example of the superposed ringing effect with a cable length of 360 m. Both  $i_{mot_A}$  and  $i_{drv_A}$  are shown.

$$Z\{\hat{i}_{mot_j}(k)|i_{drv_j}(k-1)\dots i_{drv_j}(k-N)\} = G_{est}(z)Z\{i_{drv_j}(k)\} \quad (2.10)$$

Where the quantity  $Z\{s(k)\}$  denotes the Z-transform of the discrete signal  $s(k)$ .

To design such a filter, an accurate model of the motor-cable system in the frequency domain of interest has been derived, following standard transmission lines theory (see among others [14]). The expression for  $G_{est}$  will now be derived in continuous time.

A transmission line can be modeled with the equivalent circuit shown in figure 2.6, where:

$$\begin{aligned} Z_{t1} &= Z_{t2} = Z_0[\coth(\gamma_0 h) - \operatorname{csch}(\gamma_0 h)] \\ Z_{t3} &= Z_0 \operatorname{csch}(\gamma_0 h) \\ \gamma_0 &= \sqrt{(r + j2\pi fl)(g + j2\pi fc)} \\ Z_0 &= \sqrt{\frac{r + j2\pi fl}{g + j2\pi fc}} \end{aligned} \quad (2.11)$$

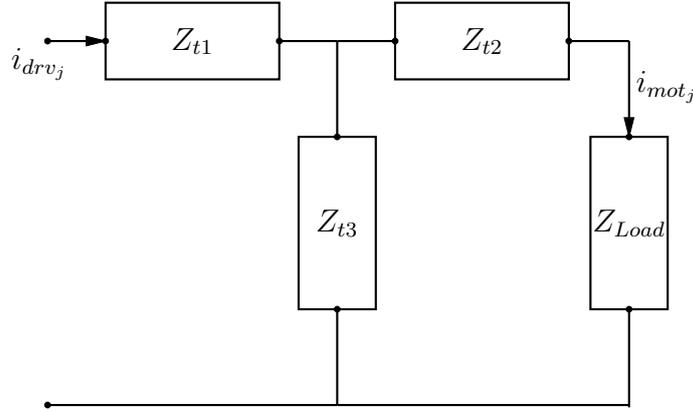


Figure 2.6: Equivalent circuit of a transmission line.

Where the symbols summarized in table 2.3 are used.

Symbol	Description	Measured value	Units
$h$	Cable Length	0.1-1.0	$km$
$r$	Nominal cable resistance	23	$\Omega/km$
$l$	Nominal cable inductance	0.6	$mH/km$
$c$	Nominal cable capacitance	48.7	$nF/km$
$g$	Nominal cable conductance	Negligible ( $2\pi fc \gg g$ )	$S/km$

Table 2.3: Used symbols for the transmission line model.

Referring to figure 2.6 we are looking for the transfer function between the current  $\hat{i}_{mot_j}$  entering in  $Z_{Load}$  and the current  $i_{drv_j}$  entering  $Z_{t1}$ . Using the current divider rule, and neglecting a certain component from  $e_j$ , which are negligible at standard stepping speeds, the transfer function results in:

$$G_{est} = \frac{i_{mot_j}}{i_{drv_j}} = \frac{Z_{t3}}{Z_{t2} + Z_{t3} + Z_{Load}} \quad (2.12)$$

After some manipulations and simplifications of common terms (see [15]), an expression for the motor side current estimator can be obtained as follows:

$$G_{est} = \frac{Z_0}{Z_0 \cosh(\gamma_0 h) + Z_{Load} \sinh(\gamma_0 h)} \quad (2.13)$$

In our case  $Z_{Load}$  is the equivalent impedance of the motor. A satisfactory model for the stepping motor electrical part has been found to be more complex than the one shown in figure 2.3 and described by equation 2.3. In fact a pole has to be added to the equivalent impedance to take into account the iron losses (basically the effects of eddy currents in the motor stator poles) that play a non negligible role at high frequencies. An equivalent circuit for this model is shown in figure 2.7.

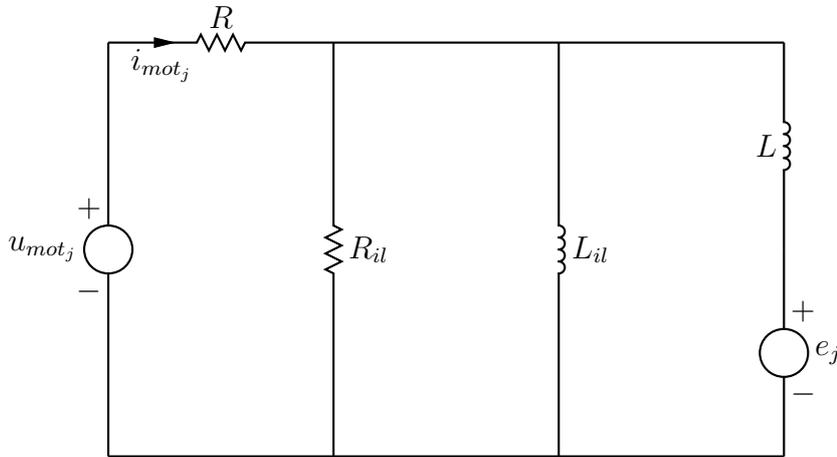


Figure 2.7: Equivalent circuit of the motor's electrical part including iron losses.

Values for  $L_{il}$  and  $R_{il}$  have been found fitting experimental data to this model as explained in [16]. Considering that  $R_{il} \gg R$  and defining  $L_{eq} \doteq LL_{il}/(L + L_{il})$  the following expression can be found:

$$Z_{Load}(s) \approx \frac{R + L_{eq}s}{1 + \frac{L_{eq}}{R_{il}}s} \quad (2.14)$$

## 2.3 The hardware prototype and the implemented control scheme

Because of the structure of hybrid stepping motors, it is common practice to close the control loop on the motor phase current (see [11] and [10]). In

the case of no external load torque, precise relative positioning of the shaft is guaranteed, at least at steady state, by the presence of teeth on the rotor.

Moreover this control scheme is the only directly applicable one in our case if we recall that no sensor (such as quadrature encoders) can be put on the motor.

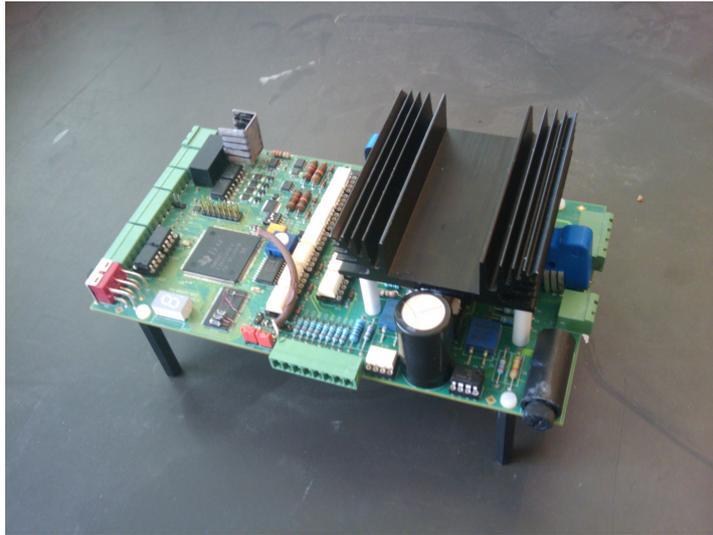


Figure 2.8: The prototype board.

The control scheme has been implemented on a PCB board based on a Texas Instruments<sup>TM</sup>DSP. The board layout design, prototyping and realization has been carried out completely using CERN facilities.

### 2.3.1 The TMS320F28335 DSP

The core of the board, where the actual control algorithm is implemented, is a Texas Instruments<sup>TM</sup>TMS320F28335 DSP. The choice of this component has been driven by its ease of programming, availability of on-chip peripherals (including an FPU) and its high clock frequency: its features are summarized in table 2.4.

Feature	Performance
Peak MMACS	150
FPU	Yes
Clock Frequency (MHz)	150
RAM	68 KB
Flash EEPROM	512 KB
PWM	18 Ch.
QEI	2 Ch.
ADC	16 Ch. 12 bits
ADC Conversion Time	80 ns
I2C	1 Ch.
UART	3 SCI
GPIO	88 pins
Timers	3 32-bit CPU Timers

Table 2.4: Texas Instruments<sup>TM</sup>TMS320F28335 Features

### 2.3.2 The Feedback Loop and Actuator implementation

Two LEM LTSR-6<sup>TM</sup> current transducers are mounted on the PCB. Each transducer provides an output that goes through an anti-aliasing filter and then to an ADC channel on the DSP. This circuit implements the feedback path.

The actuator delivers the desired control action via two full MOSFET H-bridges (one for each phase) as the one shown in figure 2.9. The *Motor Phase* block here represents a phase winding (see figure 2.2). The four MOSFET gates are controlled directly using the PWM channels available on the DSP.

The four fly-back diodes allow the easy implementation of a fast discharge scheme when needed. If, for example, a positive current is flowing through the phase (through  $Q_2$  and  $Q_3$ ) and we suddenly open all the MOSFETs the current can still flow through  $D_1$  and  $D_4$ . The voltage applied to the motor phase while the diodes are conducting is  $\approx -V_{cc}$  and the current is still positive. In short, we are forcing a positive current and a negative

voltage on the motor phase thus the power supply delivers negative power, i.e., it absorbs positive power. This will be the case until there is not enough energy stored in the phase winding (in the inductor of the equivalent circuit) and the diodes cannot conduct anymore.

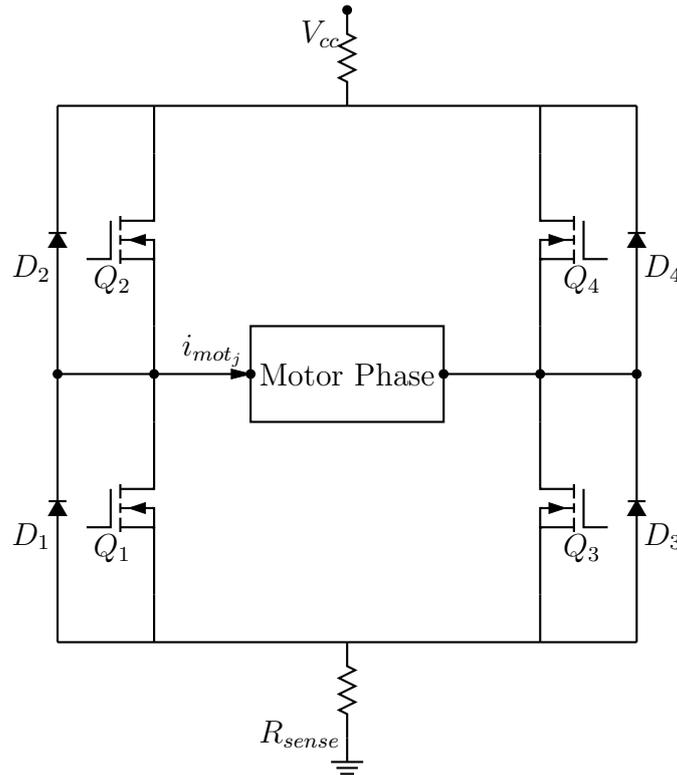


Figure 2.9: MOSFET H-bridge.

### 2.3.3 The implemented control scheme

The control scheme described so far is shown in figure 2.10

The *LEM* block represents the current transducer and the *AA* block the anti aliasing filter. The three blocks named  $R(z)$ ,  $G_{est}(z)$  and *ADC* are all implemented on the DSP.

Referring to figures 2.10 and 2.1 it is possible to link our description to that of a standard control loop. Our controlled variable is a current and  $i_{mot}^d$  represents the desired output. The regulator block  $R(z)$  calculates a

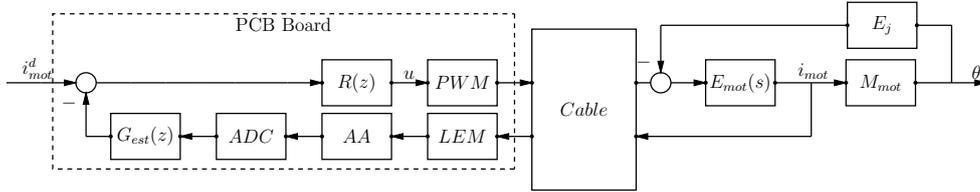


Figure 2.10: Control Scheme.

control action (a voltage to be applied to the motor phase) based on the actual and past errors. This control action is delivered to the plant via a non linear actuator: the PWM module. Our plant is the motor's electrical part. We have disturbances affecting the control action (due to the Back EMF) and the feedback loop (due to the ringing effect of the cable). The three blocks labelled  $E_{mot}(s)$ ,  $M_{mot}$ , and  $E_j$  represent the motor electrical and mechanical parts and the Back EMF respectively. The disturbance on the control action due to the effects of the cable can be safely neglected.

Our feedback loop is implemented as explained above with standard circuitry and an open loop observer whose implementation will be described in the next chapter.

# 3 Open-loop observer based solution

*Life is so unlike theory.*

---

Anthony Trollope

**The  $G_{est}$  discretization and regulator design are described. The software architecture for the open-loop observer based solution is introduced.**

Our control law ( $R(s)$  block) and open-loop observer ( $G_{est}(s)$  block) are implemented on the Texas Instruments<sup>TM</sup>TMS320F28335 DSP. In this chapter expressions for both  $G_{est}$  in discrete time and for the regulator will be derived.

The general software implementation architecture will also be described.

## 3.1 Discretization of the open-loop observer

In chapter 2 a continuous time domain expression for the transfer function between the motor phase current and the driver side current was found to be:

$$G_{est} = \frac{Z_0}{Z_0 \cosh(\gamma_0 h) + Z_{Load} \sinh(\gamma_0 h)} \quad (3.1)$$

where:

$$Z_{Load}(s) \approx \frac{R + L_{eq}s}{1 + \frac{L_{eq}}{R_{il}}s} \quad (3.2)$$

The derived expression for  $G_{est}$  has an infinite number of poles and zeros, moreover it is not a polynomial function of the frequency and thus cannot be directly discretized with the bilinear transformation.

It has been decided to calculate a Padé approximation of the filter in continuous time and then to discretize this expression with a standard bilinear transformation.

The Padé approximant of a function is its best representation as a rational function. It is normally superior to the Taylor expansion when a function contains poles [17].

The bilinear or Tustin transformation is a widely used method to calculate the discrete-time version of a transfer function in order to obtain a representation in the  $z$  domain. It relies on the fact that  $z = e^{sT}$  and thus, using the Taylor series:

$$s = \frac{1}{T} \ln(z) \approx \frac{2}{T} \left[ \frac{z-1}{z+1} \right] \quad (3.3)$$

It is readily used especially when a transfer function is a rational function as it is our case once we have utilized the Padé approximation. These two steps allow us to derive the IIR filter that best approximates  $G_{est}(s)$ .

The expression of  $G_{est}$  strongly depends on the cable length and so will the coefficients of the derived filter. Instead of calculating the coefficients for each cable length and hard-code them in the software, a more elegant solution has been found: the symbolic expressions for the IIR filter coefficients in terms of the cable length have been found using the Wolfram<sup>TM</sup>Mathematica<sup>TM</sup> environment and implemented on the driver, which is able to estimate the length of the cable to which it is attached.

The best trade off between the computational complexity of the resulting filter and the accuracy of the approximation has resulted (see [18]) in a second order IIR filter:

$$G_{est}(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \quad (3.4)$$

We recall here that we need to simultaneously control two motor phases so that we actually have two digital filters (one to act on  $i_{drv_A}$  and the other on  $i_{drv_B}$ ) running at the same time.

## 3.2 Digital controller design

The control law is calculated adding a proportional and an integral action to ensure an acceptable settling time and zero tracking error at steady state [19]. A far pole is added to this standard PI controller to avoid the amplification of high-frequency noise:

$$R(s) = \mu \frac{1 + s\tau_z}{s(1 + s\tau_p)} = \mu \frac{\tau_z - \tau_p}{1 + s\tau_p} + \frac{\mu}{s} = R_P(s) + R_I(s) \quad (3.5)$$

Actually, the control actuator (full MOSFET H-bridge) can only deliver control actions that fall in a certain interval since its voltage is limited between  $\pm V_{cc}$ , as should be evident from figure 2.9. The control action is in fact saturated: in the case of integrator windup, the regulator would fail to meet maximum overshoot and settling time requirements (see [20] and the figures therein for an explanation of integrator saturation and its consequences).

The last expression in 3.5 allows us to easily design an anti windup scheme for the controller integral part. The saturating effect of the non linear actuator is in fact modelled directly in the regulator as shown in figure 3.1, where the implemented control scheme is displayed.

The expressions or values for the constants appearing in 3.5 have been derived using the root-locus method [21] and are reported in table 3.1. The desaturation constant has been fixed to  $K_d = 1/\mu\tau_z$  following the results in [20].

The  $B_{cl}$  parameter, the desired closed-loop bandwidth, has to be set by the user and is actually the only *free* parameter for the controller. If we

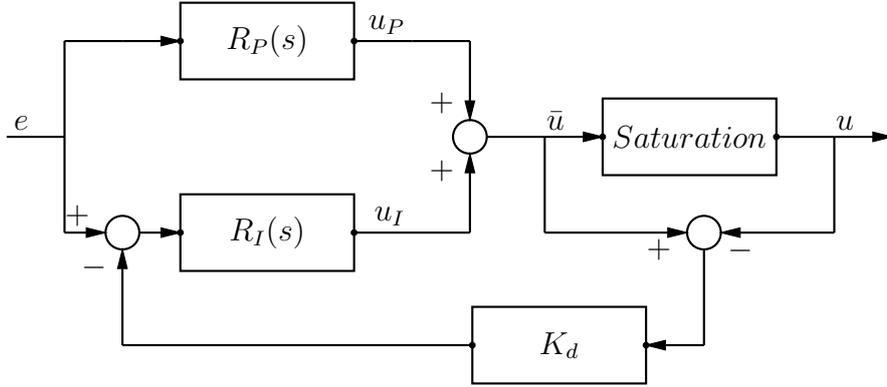


Figure 3.1: Regulator with anti-windup scheme.

Symbol	Expression	Units
$\tau_z$	$(lh + L_{eq})/(R + rh)$	$s$
$\tau_p$	$10.56 \times 10^{-6}$	$s$
$\mu$	$2\pi B_{cl}(R_w + rh)$	$\Omega/s$

Table 3.1: Values and expressions for the controller constants,  $l$ ,  $h$ , and  $r$  were defined in table 2.3.

use a Tustin transformation to discretize 3.5 and include the anti windup scheme, we obtain the following control law:

$$\begin{aligned}
 u_P(k) &= -\frac{T - 2\tau_p}{T + 2\tau_p}u_P(k-1) + \mu T \frac{\tau_z - \tau_p}{T + 2\tau_p}[e(k) + e(k-1)] \\
 u_I(k) &= u_I(k-1) + \mu \frac{T}{2}[e(k) + e(k+1)] \\
 &\quad + \mu T K_d[u(k-1) - \bar{u}(k-1)] \\
 \bar{u}(k) &= u_P(k) + u_I(k) \\
 u(k) &= \begin{cases} \bar{u}(k) & \text{when } -V_{cc} \leq \bar{u}(k) \leq V_{cc} \\ V_{cc} & \text{when } \bar{u}(k) > V_{cc} \\ -V_{cc} & \text{when } \bar{u}(k) < -V_{cc} \end{cases} \quad (3.6)
 \end{aligned}$$

where  $T$  is the controller sampling period in seconds<sup>1</sup>.

<sup>1</sup>We remark here that the  $G_{est}$ 's sampling time can be different from that of the

### 3.3 Software architecture

Figure 3.2 shows the software architecture chosen for implementation. A description of the task accomplished in each state follows.

**InitCommunication:** In this state the communication sub-module is initialized. Communication hardware is enabled and configured, software parameters such as peripheral address and communication speed are set.

**WaitForParameters:** The board is programmed to check at boot-time whether or not it has ever been configured before (a register in Flash memory is toggled after the configuration ended successfully). If it has never been configured it polls the communication module to allow for easy configuration.

*Details:* Physical parameters such as those reported in tables 2.3 and 3.1 as well as  $B_{cl}$  are sent through a ModBus over an RS-232 communication line. Additionally, rates for the filter and controller interrupts, PWM frequency, LEM transducer nominal gain, default stepping mode and third harmonic correction need to be set. These tasks are completed using a Graphical User Interface (GUI) developed ad-hoc. Once the configuration is completed, a RunStart signal is toggled via the GUI to allow the board to save the configuration in Flash memory.

**InitSystem:** Now the program can initialize all the used peripherals: ADC, GPIO pins, interrupt timers, PWM and safety measurements (H - bridge temperature and short circuit detection). The reading of the  $V_{cc}$  voltage supplying the H-bridge is also initialized to raise warnings in case of voltage drops and to update the *Saturation* block in figure 3.1 (see last line in equation 3.6).

---

controller. The filter will produce an estimation at a certain rate and this sequence will be down-sampled by the controller to produce a correcting action.

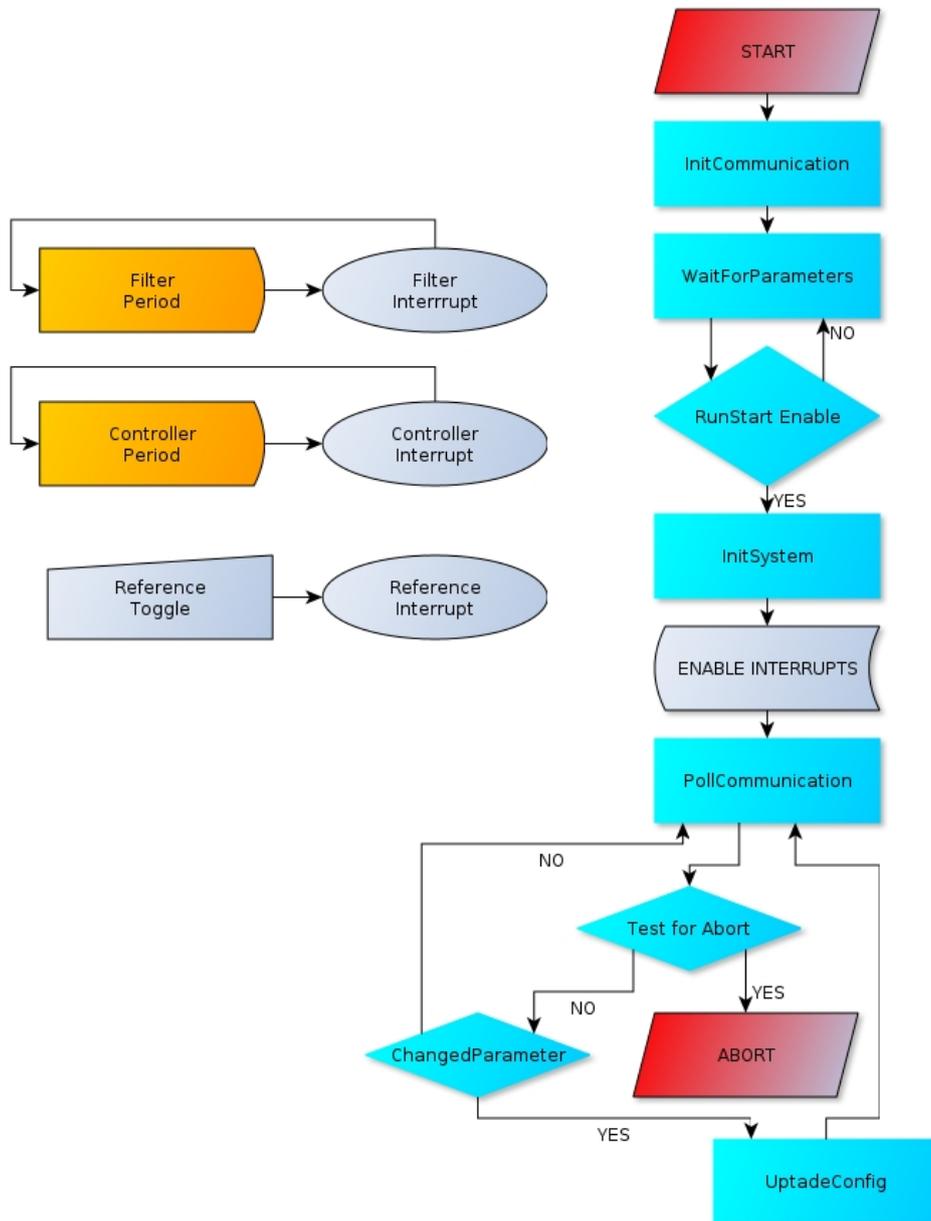


Figure 3.2: Flowchart for the chosen implementation scheme.

*Details:* The length of the cable connecting the driver to the motor phases is estimated by injecting a known constant voltage in the cable (at steady state the ringing effect of the cable will be negligible) and measuring the current induced. A resistance  $R_{est}$  is then estimated

for the motor-cable system. As  $R$  and  $r$  are known parameters, a value for  $h$  can be calculated as  $(R_{est} - R)/r$  (see table 2.3).

Now all data structures (filter and controller coefficients, etc.) can be populated.

Finally the interrupts are enabled so that the filter and controller start to operate. A motor step request reaches the board in the form of a change of state on a digital input line. An interrupt routine is linked to this event so that new references for the two phase currents can be calculated (see Reference Interrupt state description).

**Poll Communication:** Stepping mode and third harmonic correction parameters can be edited while the driver is operating, furthermore some safety critical information such as measured  $V_{cc}$ , H-bridge temperature, short circuit in the high-power path and detection for a missing motor-phase (open circuit) can be monitored on line with another ad-hoc GUI.

While being interrupted to execute high priority tasks, the main loop keeps polling the communication module. In the event of an abort condition the driver is programmed to open all the power MOSFETs in the H-bridge and halt.

**UpdateConfig:** If a request is made to change a parameter while the driver is in operation, the motor step request interrupt is disabled (to avoid possible inconsistencies), the change is applied and the step request interrupt is re-enabled.

**Filter Interrupt:** In this routine a new sample for both currents is read from the ADC and a new estimation for the motor side current is calculated and made available. This is the highest priority interrupt.

**Controller Interrupt** The controller, based on the latest available motor current estimation, calculates a control action as described in equa-

tions 3.6. If the motor is enabled, this control action is then delivered to the plant via the PWM module.

**Reference Interrupt:** When a change in logic state occurs on a dedicated input, a new reference for the two motor phase currents is calculated according to equation 2.7.

## 3.4 Graphical User Interfaces

In the previous section it has been described how the driver can be configured and monitored both during and before operation. Two ad-hoc GUI based applications have been developed for this purpose: *CollimationDriverCONFIG* and *CollimationDriverGUI*.

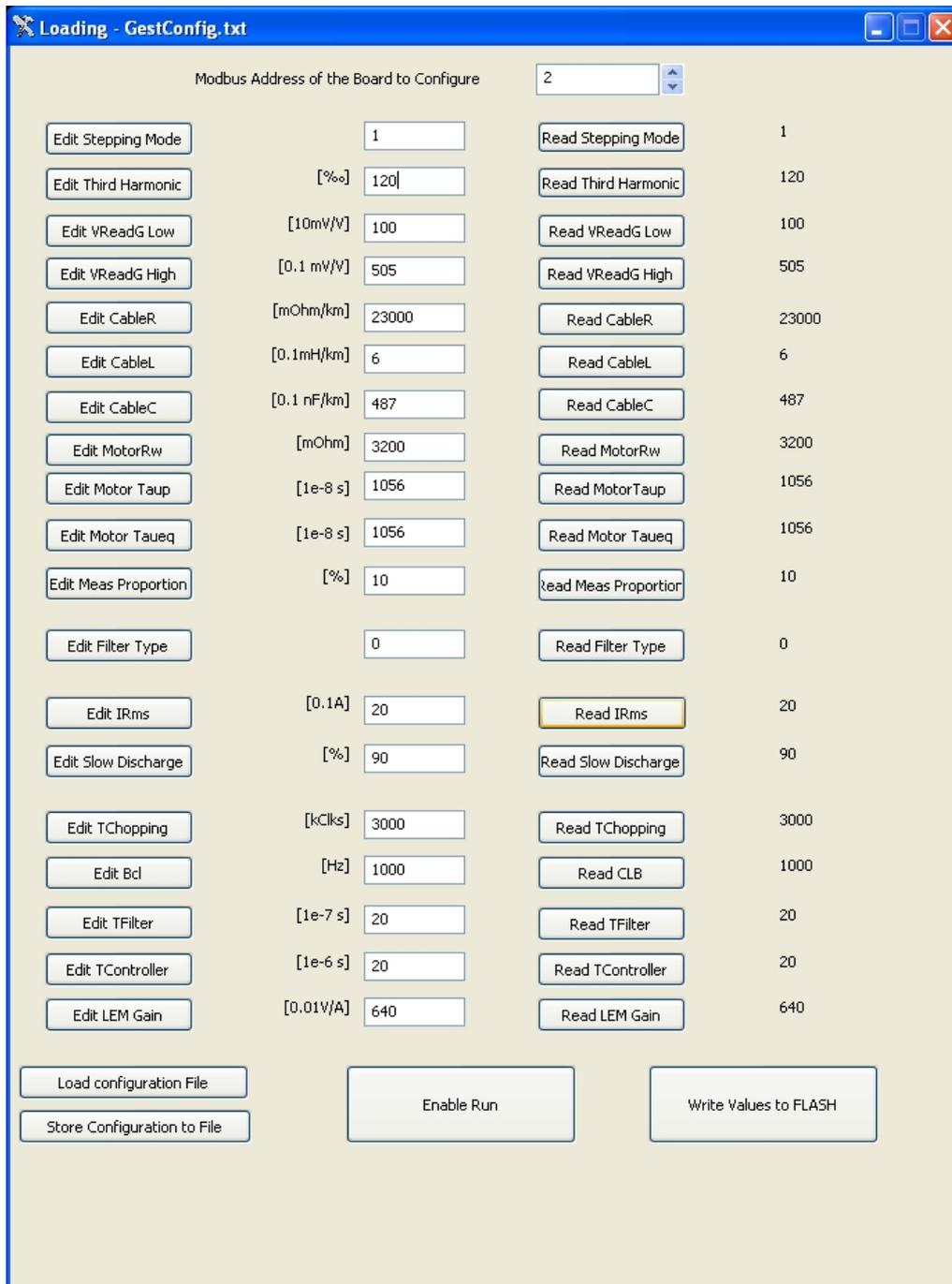
### 3.4.1 CollimationDriverCONFIG

This application provides a GUI to allow the user to configure the drive, ideally once and for all, at first boot. A snapshot image of the GUI for this application is shown in figure 3.3. As can be seen in the figure, this application allows for various parameters' configuration, including the two main controller interrupts' sampling times. The possibility to save and load configuration files is available together with the capability of configuring many boards connected to the same communication bus exploiting the address field of the ModBus protocol packet (see [22]).

### 3.4.2 CollimationDriverGUI

This GUI based application allows a user to easily monitor the driver safety-critical parameters. A snapshot image of the operating GUI is shown in figure 3.4.

This application can be set in polling mode, so that it automatically retrieves all time-varying parameters (stepping mode, third harmonic correction, short circuit detection status, phase missing detection, power supply

Figure 3.3: Snapshot image of the *CollimationDriverCONFIG* GUI.

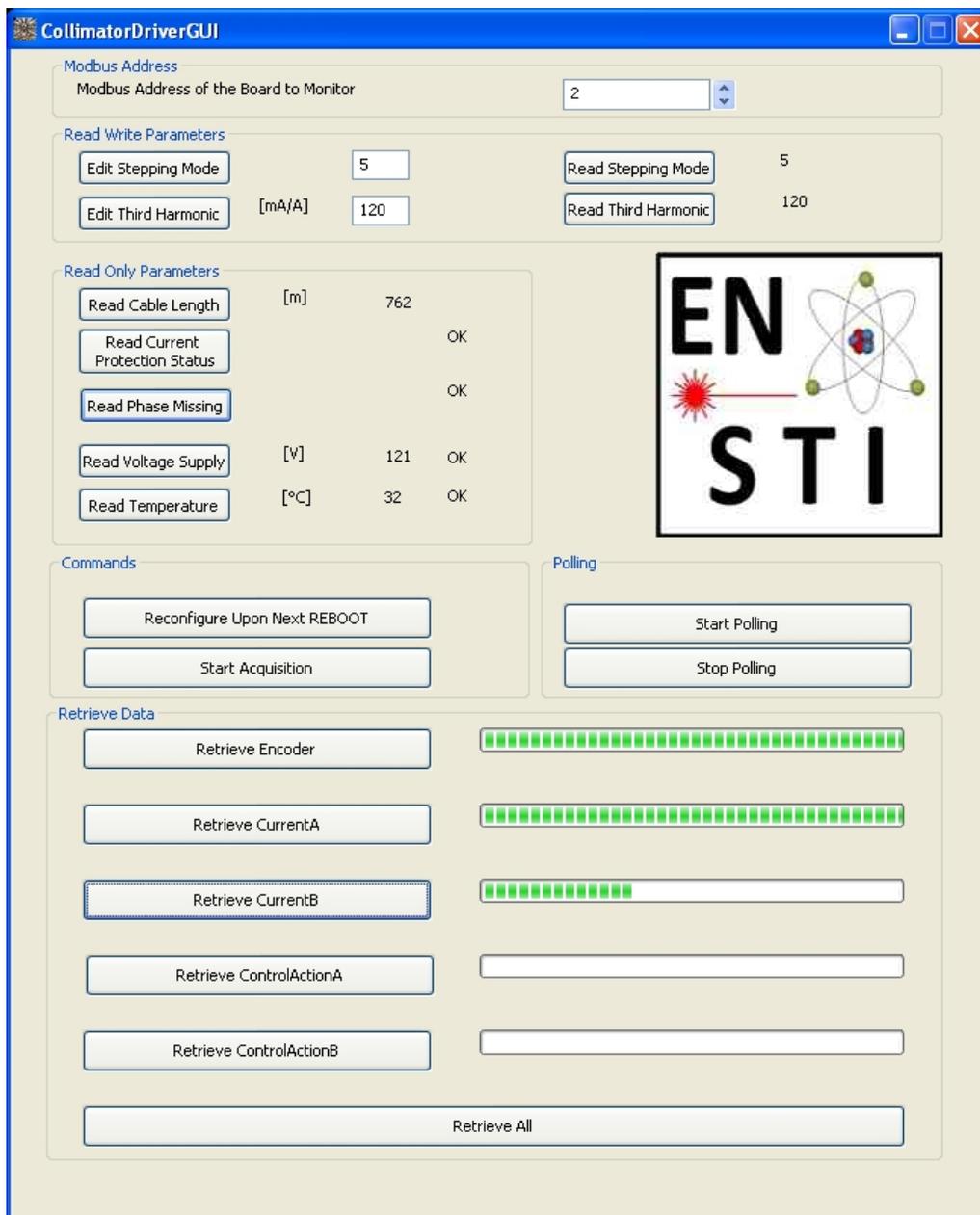


Figure 3.4: Snapshot image of the *CollimationDriverGUI* GUI.

voltage and h-bridge temperature) at regular intervals. The stepping mode as well as the third harmonic correction can be edited while the driver is working.

For testing, post-mortem analysis and parameter tuning, a data acquisition can be started directly on the DSP (simply by pressing the **Start Acquisition** button). A large (1 MByte) static RAM is installed on the PCB and exploiting the TMS320F28335 capabilities it is possible to include it directly in the DSP addressing space, thus allowing easy and fast read and write operations. When an acquisition is started the filter output and control action for the two currents, as well as the reading of a quadrature encoder<sup>2</sup> is saved within the controller interrupt routine until the memory is filled. These signals can then be retrieved by using the GUI via the RS-232 bus (this can be done by pressing the corresponding buttons in the bottom half of the GUI).

---

<sup>2</sup>Available only on the laboratory testbench set up, not in the final installation.

# 4 The Extended Kalman Filter based solution

*There ain't no rules around here.  
We're trying to accomplish something!*

---

Thomas Alva Edison

**The EKF algorithm steps are recalled in an informal way. A state space model is derived for a hybrid stepping motor. The implementation details for the EKF based solution are presented.**

The open-loop estimator solution has been expanded to include an Extended Kalman Filter (EKF) based observer. This allows the estimation of the two motor-phase currents, the shaft angular position and velocity as well as the unknown load torque, which can then be used in a state feedback controller.

## 4.1 The Extended Kalman Filter

The expression for the Kalman Filter will be now derived, in a somewhat intuitive way, for a linear system. A more formal procedure can be found in [23] and [24].

### 4.1.1 The asymptotic observer

In deterministic linear system theory, the so called *observer design* problem is that of developing a realistic solution to the problem of estimating the state of a linear system, given access to inputs and outputs.

Consider a linear, time invariant system of the form:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, & k \geq 0 \\ \mathbf{y}_k = \mathbf{H}\mathbf{x}_k \end{cases} \quad (4.1)$$

If we have a good estimation of the initial state  $\mathbf{x}_0$  (say  $\hat{\mathbf{x}}_0$ ) the problem would not seem hard: we just set up a *dummy* system using the known  $\mathbf{A}$ ,  $\mathbf{B}$  and  $\mathbf{u}_k$  as:

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k, \quad k \geq 0 \quad (4.2)$$

to calculate the system's state. Unfortunately, even if the real system is actually described by 4.1 (normally not the case), no matter how good the estimate  $\hat{\mathbf{x}}_0$  is, unless it's perfect, the above method will not work if the matrix  $\mathbf{A}$  has eigenvalues lying outside the unit circle. Indeed the error

$$\mathbf{x}_k - \hat{\mathbf{x}}_k \doteq \tilde{\mathbf{x}}_k \quad (4.3)$$

obeys the equation

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{A}\tilde{\mathbf{x}}_k, \quad \tilde{\mathbf{x}}_0 = \mathbf{x}_0 - \hat{\mathbf{x}}_0 \quad (4.4)$$

so that

$$\tilde{\mathbf{x}}_k = \mathbf{A}^k \tilde{\mathbf{x}}_0 \quad (4.5)$$

If  $\mathbf{A}$  has eigenvalues with a norm bigger than one,  $\|\tilde{\mathbf{x}}_k\|$  will grow exponentially, unless  $\tilde{\mathbf{x}}_0$  is completely in the subspace spanned by the eigenvectors and generalized eigenvectors corresponding to the stable eigenvalues of  $\mathbf{A}$ , a condition not normally met in real life.

For the sake of consistency, we remark that the both  $G_{est}(s)$  and  $G_{est}(z)$  are strictly stable, thus the results presented in the previous chapters still hold.

In making a post-mortem analysis of this unsuccessful attempt to solve the *observer design* problem for system 4.1, we may note that it makes no use of the available output sequence  $\mathbf{y}_k$ . Indeed, the term:

$$\tilde{\mathbf{y}}_k \doteq \mathbf{y}_k - \hat{\mathbf{y}}_k = \mathbf{H}(\mathbf{x}_k - \hat{\mathbf{x}}_k) = \mathbf{H}\tilde{\mathbf{x}}_k, \quad k \geq 0 \quad (4.6)$$

can give us the information of how  $\tilde{\mathbf{x}}_k$  is behaving, provided it does not always lie in the nullspace of  $\mathbf{H}$ , a quite unlikely event. Thus, suppose we also drive our *dummy* system 4.2 with a term  $\mathbf{K}\tilde{\mathbf{y}}_k$  where  $\mathbf{K}$  is a suitably chosen matrix so that:

$$\hat{\mathbf{x}}_{k+1} = \mathbf{A}\hat{\mathbf{x}}_k + \mathbf{B}\mathbf{u}_k + \mathbf{K}\tilde{\mathbf{y}}_k \quad (4.7)$$

The error will now evolve according to the following equation:

$$\tilde{\mathbf{x}}_{k+1} = (\mathbf{A} - \mathbf{K}\mathbf{H})\tilde{\mathbf{x}}_k \quad (4.8)$$

We can thus choose  $\mathbf{K}$  so that  $(\mathbf{A} - \mathbf{K}\mathbf{H})$  has any desired eigenvalue. Then the error can be driven to zero at any desired rate.

### 4.1.2 The optimal transient observer

It should suffice to say that a lot of experience and experiments suggest that the noiseless model described in 4.1 is not correct for most real world systems. A more realistic state-space model for a linear time invariant system is:

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}(\mathbf{u}_k + \mathbf{w}_k), & k \geq 0 \\ \mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \end{cases} \quad (4.9)$$

where the so-called process noise  $\mathbf{w}_k$ , measurement noise  $\mathbf{v}_k$ , and the initial state  $\mathbf{x}_0$  are assumed to be random. This primary randomness in turn makes the state  $\mathbf{x}_k$  and output  $\mathbf{y}_k$  themselves random. For the following results to hold it is crucial that the aforementioned random variables meet some requirements (with a little burden of notation):

$$E \left\{ \begin{bmatrix} \mathbf{w}_k \\ \mathbf{v}_k \\ \mathbf{x}_0 \end{bmatrix} \begin{bmatrix} \mathbf{w}_k^T & \mathbf{v}_k^T & \mathbf{x}_0^T & 1 \end{bmatrix} \right\} = \begin{bmatrix} \mathbf{Q} & 0 & & \\ 0 & \mathbf{R} & & \\ & & \delta_{i,j} & \\ 0 & 0 & & \mathbf{\Pi}_0 \end{bmatrix} \quad (4.10)$$

Where the  $E$  operator is the mathematical expectation and  $\delta_{i,j}$  is the Kronecker delta function. Thus defining  $\mathbf{Q}$  as the covariance matrix for the process noise,  $\mathbf{R}$  the covariance matrix for the measurement noise and  $\mathbf{\Pi}_0$  the covariance of the initial state estimation.

This matrix equality simply states that all random sequences are realizations of independent stationary processes characterized by zero-mean distributions and that each component of these random vector-valued variables is independent of all other components. It is also required that each sample in these sequences is independent of any (previous and future) other sample along with the fact that all the distributions these sequences are drawn from are Gaussian.

Once the structure for the observer has been fixed to that of 4.7, we shall find a value for  $\mathbf{K}$  suitable to take care of the fact that our model is now described by 4.9.

We may want, for example, to choose  $\mathbf{K}$  at each time step  $k$  such that:

$$E\{(\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T\} = E\{\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^T\} \quad \text{is a minimum} \quad (4.11)$$

Using 4.10 it can be shown that:

$$\begin{aligned} E\{\tilde{\mathbf{x}}_{k+1} \tilde{\mathbf{x}}_{k+1}^T\} &= (\mathbf{A} - \mathbf{K}_k)(E\{\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^T\})(\mathbf{A} - \mathbf{K}_k)^T + \\ &+ \begin{bmatrix} \mathbf{B} & -\mathbf{K}_k \end{bmatrix} \begin{bmatrix} \mathbf{Q} & 0 \\ 0 & \mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{B}^T \\ -\mathbf{K}_k^T \end{bmatrix} \end{aligned} \quad (4.12)$$

On the grounds of the discussion held so far and using basic matrix algebra results (such as the small matrix inversion lemma), the complete algorithm for the optimal observer in the least square sense 4.11 can be derived. The resulting algorithm is the well-known Kalman Filter.

### 4.1.3 The Kalman Filter algorithm

For a linear time invariant system described by 4.9 the optimal state estimation in the least square sense can be calculated as a prediction step:

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \mathbf{A}\hat{\mathbf{x}}_{k|k} + \mathbf{B}\mathbf{u}_k \\ \mathbf{P}_{k+1|k} &= \mathbf{A}\mathbf{P}_{k|k}\mathbf{A}^T + \mathbf{Q}\end{aligned}\quad (4.13)$$

followed by a correction step:

$$\begin{aligned}\mathbf{K}_{k+1} &= \mathbf{P}_{k+1|k}\mathbf{H}^T [\mathbf{R} + \mathbf{H}\mathbf{P}_{k+1|k}\mathbf{H}^T]^{-1} \\ \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1}[\mathbf{y}_{k+1} - \mathbf{H}\hat{\mathbf{x}}_{k+1|k}] \\ \mathbf{P}_{k+1|k+1} &= \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1}\mathbf{H}\mathbf{P}_{k+1|k}\end{aligned}\quad (4.14)$$

It is worth noting that, following the steps to minimize 4.12,  $\mathbf{P}_{k+1|k+1}$  turns out to be the covariance matrix of the estimation error:  $\mathbf{P}_{k+1|k+1} = E\{\tilde{\mathbf{x}}_{k+1|k+1}\tilde{\mathbf{x}}_{k+1|k+1}^T\}$ .

The Kalman Filter can be adapted to non-linear systems described by state space equations. One of the resulting algorithms is known under the name of Extended Kalman Filter.

### 4.1.4 The Extended Kalman Filter

For a nonlinear system described by the following state-space equations:

$$\begin{aligned}\mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k\end{aligned}\quad (4.15)$$

with hypothesis 4.10 holding, the steps for the EKF are as follows:

*Prediction:*

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} &= \mathbf{f}(\hat{\mathbf{x}}_{k|k}, \mathbf{u}_k) \\ \mathbf{P}_{k+1|k} &= \mathbf{A}_k\mathbf{P}_{k|k}\mathbf{A}_k^T + \mathbf{Q}\end{aligned}\quad (4.16)$$

where:

$$\mathbf{A}_k = \left. \frac{\partial \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \mathbf{x}_{k|k}} \quad (4.17)$$

*Correction:*

$$\begin{aligned} \mathbf{K}_{k+1} &= \mathbf{P}_{k+1|k} \mathbf{H}_k^T [\mathbf{R} + \mathbf{H}_k \mathbf{P}_{k+1|k} \mathbf{H}_k^T]^{-1} \\ \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1} [\mathbf{y}_{k+1} - \mathbf{h}(\hat{\mathbf{x}}_{k+1|k})] \\ \mathbf{P}_{k+1|k+1} &= \mathbf{P}_{k+1|k} - \mathbf{K}_{k+1} \mathbf{H}_k \mathbf{P}_{k+1|k} \end{aligned} \quad (4.18)$$

where:

$$\mathbf{H}_k = \left. \frac{d\mathbf{h}(\mathbf{x}_k)}{d\mathbf{x}_k} \right|_{\mathbf{x}_k = \mathbf{x}_{k+1|k}}$$

Despite the fact that the EKF estimation cannot be proven to converge over time [25] to the actual system state, it has been successfully applied to a considerable number and variety of problems [26–28].

## 4.2 State space model of a hybrid stepping motor

The state space description in discrete time for a hybrid stepping motor presented in 4.20, can be derived by making use of definitions in chapter 2, denoting the state, input and measurement vector respectively as:

$$\begin{aligned} \mathbf{x}_k &= [i_{mot_A}(k) \quad i_{mot_B}(k) \quad \omega(k) \quad \theta(k) \quad \tau_L(k)]^T \\ \mathbf{u}_k &= [u_{mot_A}(k) \quad u_{mot_B}(k)]^T \\ \mathbf{y}_k &= [i_{mot_A}(k) \quad i_{mot_B}(k)]^T \end{aligned} \quad (4.19)$$

and by applying a forward Euler transformation with sampling time  $T$  to the model described by 2.3 and 2.4.

$$\begin{aligned}
f_1(\mathbf{x}_k, \mathbf{u}_k) &= (1 + Ta_{11})x_{1_k} + Ta_{13}x_{3_k} \sin px_{4_k} \\
&\quad + Tb_{11}u_{1_k} \\
f_2(\mathbf{x}_k, \mathbf{u}_k) &= (1 + Ta_{22})x_{2_k} + Ta_{23}x_{3_k} \cos px_{4_k} \\
&\quad + Tb_{22}u_{2_k} \\
f_3(\mathbf{x}_k, \mathbf{u}_k) &= Ta_{31}x_{1_k} \sin px_{4_k} + Ta_{35}x_{5_k} \\
&\quad + Ta_{32}x_{2_k} \cos px_{4_k} \\
&\quad + (1 + Ta_{33})x_{3_k} + Ta_{34} \sin(2px_{4_k} + \phi) \\
f_4(\mathbf{x}_k, \mathbf{u}_k) &= x_{4_k} + Tx_{3_k} \\
f_5(\mathbf{x}_k, \mathbf{u}_k) &= x_{5_k}
\end{aligned} \tag{4.20}$$

Expressions for constants  $a_{m,n}$  and  $b_{q,r}$  are reported in 4.21 and 4.22 using symbols from table 2.2.

$$\begin{aligned}
a_{11} &= -\frac{R}{L} & a_{12} &= 0 & a_{13} &= \frac{K_m}{L} & a_{14} &= 0 & a_{15} &= 0 \\
a_{21} &= 0 & a_{22} &= -\frac{R}{L} & a_{23} &= -\frac{K_m}{L} & a_{24} &= 0 & a_{25} &= 0 \\
a_{31} &= -\frac{K_m}{J} & a_{32} &= \frac{K_m}{J} & a_{33} &= -\frac{B}{J} & a_{34} &= -\frac{T_{dm}}{J} & a_{35} &= -\frac{1}{J}
\end{aligned} \tag{4.21}$$

and

$$b_{11} = \frac{1}{L} \qquad b_{22} = \frac{1}{L}. \tag{4.22}$$

It should be noted that the unknown load torque has been added to the state-space description as  $x_{5_k}$  and modeled as a random walk process following results in [29] or [30].

### 4.3 The proposed implementation scheme

The EKF uses a discrete-time model. Since the driven system includes both the motor and cable, logically the EKF should use the combined model. The sampling period of the discrete-time model can be chosen such that the sampling frequency  $f_s = 1/T$  satisfies:

$$6f_{bw} \leq f_s \leq 25f_{bw}, \quad (4.23)$$

where  $f_{bw}$  is the bandwidth of the system producing the signals to be sampled [31]. In the case considered, the highest frequency signals are found in the electrical sub-system containing the cable and the motor phase. As previously mentioned, the driver's output voltage is modulated by a PWM. At a constant duty cycle, this PWM signal has a fundamental harmonic at its chopping frequency, which is typically chosen at 50 kHz due to EMI constraints [32]. Being similar to a square wave, it also has non-negligible higher harmonics. Depending on the cable's length, the electrical sub-system can have a gain at the frequencies of these harmonics that is even greater than at DC. This is evident from Fig. 4.1, which shows the magnitude Bode diagram of  $G_{cm}(s)$  (defined as the transfer function between the driver side voltage and the driver side current) for two extremes of cable length, when the typical cable and motor parameter values in Table 2.3 are used.

Even only considering the first 3 harmonics, a minimum sampling frequency of  $6 \times 3 \times 50kHz = 900kHz$  would be required. Running a multi-state EKF at this sampling frequency exceeds the computational power of standard DSPs and thus renders the model practically unusable for industrial applications. For this reason, the use of the motor model alone in the EKF will be considered hereafter. Its electrical closed-loop bandwidth is typically at 1-2 kHz [32]. This bandwidth requires a sampling frequency for the EKF of 6-50kHz, which is much more reasonable for implementation on a DSP.

As seen in the discretised model in the previous section, using exclusively the motor model in the EKF means that the inputs and measurements

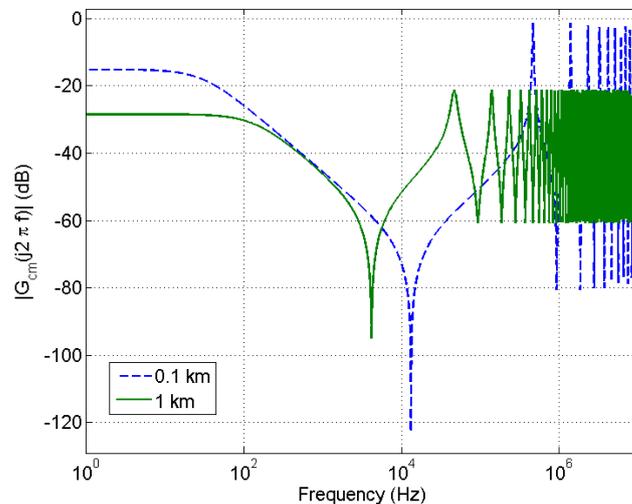


Figure 4.1: Magnitude Bode diagram of  $G_{cm}(s)$  for two extremes of cable length

used by the algorithm are the motor-side voltages and currents, respectively. However, as previously stated, these signals are not directly available in ordinary operation since measurements can only be made on the drive-side of the cable. The length of these cables, together with the relatively high chopping frequency of the PWM voltage signals, makes them act as transmission lines. These lines' principal characteristics are those of a series resistance with a shunt capacitance across the motor's polarities. The capacitive behaviour induces large oscillations in the current at the driver-end of the cable and, similarly, oscillations in the voltage at the motor end of the cable, both with frequency components close to those of the PWM's harmonics. These oscillations on the drive-side current and motor-side voltage can be viewed as measurement and process noise, respectively, however the properties of these noises do not satisfy assumptions outlined in Sec. 4.1.2, meaning an EKF applied on these signals is unlikely to produce reliable state estimates. Noise whitening filters are, therefore, required.

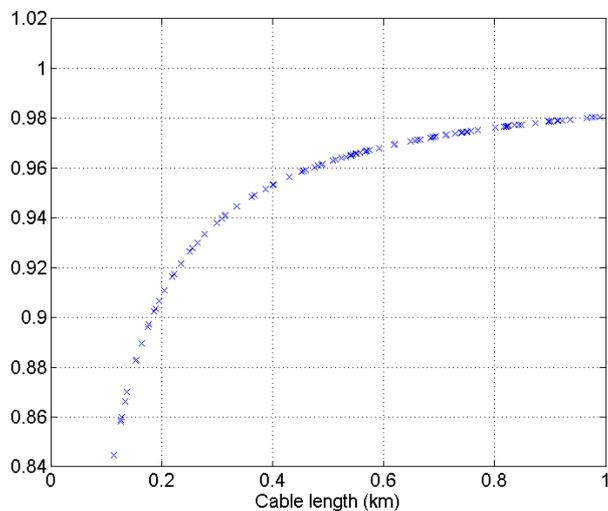


Figure 4.2: Magnitude of  $G_{est}(z)$ 's poles over a range of cable lengths

### 4.3.1 Motor-side current estimators

For this purpose, obviously, the discrete IIR filter  $G_{est}(z)$  is used, whose theoretical justification has been outlined in section 2.2. For the sake of consistency we add here two experimental results that support this choice.

The stability of the estimator  $G_{est}(z)$  is obviously of great importance. It is, however, difficult to analytically prove the filter's stability for all cable lengths due to the complexity of the coefficient expressions found via the Padé approximant procedure. Nonetheless, an empirical assessment has been made by evaluating  $G_{est}(z)$ 's poles over a grid of 100 randomly distributed cable lengths in the range of interest i.e. [0.1 km, 1 km].  $G_{est}(z)$ 's poles form a complex conjugate pair, their magnitudes are therefore the same. This value is plotted for each cable length in Fig. 4.2.

It can be seen that the poles are all inside the unit circle, ensuring, at least for the cable lengths considered, that  $G_{est}(z)$  is stable. Furthermore the values vary smoothly, despite the use of randomly distributed cable lengths, so it is unlikely that filters evaluated at inter-sample cable lengths would be unstable.

For the outlined application it is also fundamental that the estimation error of  $G_{est}$  falls in a Gaussian distribution, since this error can be seen as a *measurement noise* affecting our information about the motor phases' currents.

To evaluate this distribution (which also provides insights on the quantities needed in the diagonal elements of the EKF  $\mathbf{R}$  matrix, see 4.10 and next chapter for details and theoretical insights) an ad hoc experiment has been put forward. Practically this was realised by physically disconnecting one motor phase from the drive and using its feedback path to acquire the motor-side current for the phase under test. The estimated motor-side current instead was recorded whilst in normal operation, i.e. stepping the current, in DAQ mode. The estimation error can obviously be calculated as the difference between these two signals. This experiment was carried out once for each phase. Data histograms for both phases are shown in Fig. 4.3.

### 4.3.2 Motor-side voltage estimator

The motor-side voltage estimator is based on the idea that, within the relatively low frequency range of interest, the cable can be modelled such that the impedances (see Fig. 2.6)  $Z_{t1}$  and  $Z_{t2}$  are replaced by resistors and  $Z_{t3}$  is replaced by a capacitor, see Fig. 4.4. Neglecting the contribution from the Back-EMF, the transfer function between motor-side and driver-side voltage is given by:

$$\begin{aligned} H_v &= \frac{U_{mot_j}(s)}{U_{drv_j}(s)} \\ &= \frac{Z_{t3}Z_{load}}{Z_{t1}(Z_{t2} + Z_{t3} + Z_{mot}) + Z_{t3}(Z_{t2} + Z_{mot})}. \end{aligned} \quad (4.24)$$

Fig. 4.5 compares the frequency responses of this transfer function when the impedances are given by expressions (2.11) and when they are represented by 2 resistors and a capacitor, as in Fig. 4.4. The good correspondence between the two frequency responses at lower frequencies is evident.

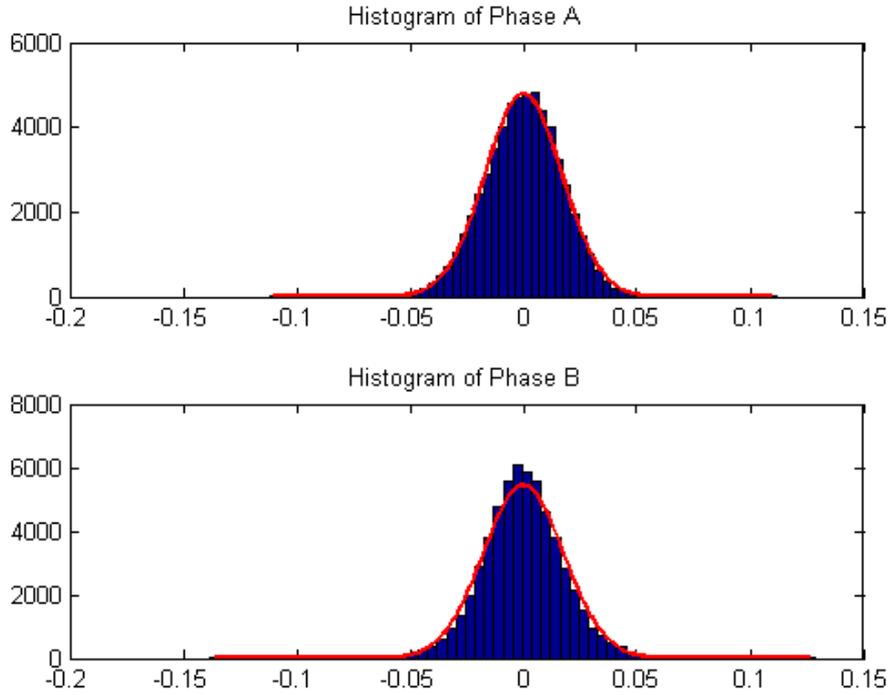


Figure 4.3: Distribution of the open-loop estimator reconstruction error. Errors are evaluated in [A].

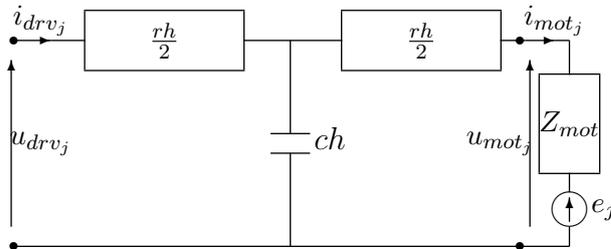


Figure 4.4: Low frequency cable model connected to motor phase

Unfortunately it is not possible to directly implement  $H_v(s)$  as the voltage estimator because it has a minimum bandwidth of over 3.3 MHz. This implies that a sampling rate in the order of 20 MHz would be necessary (according to inequalities in 4.23).

Nonetheless, using the circuit approximation of Fig. 4.4, it is still pos-

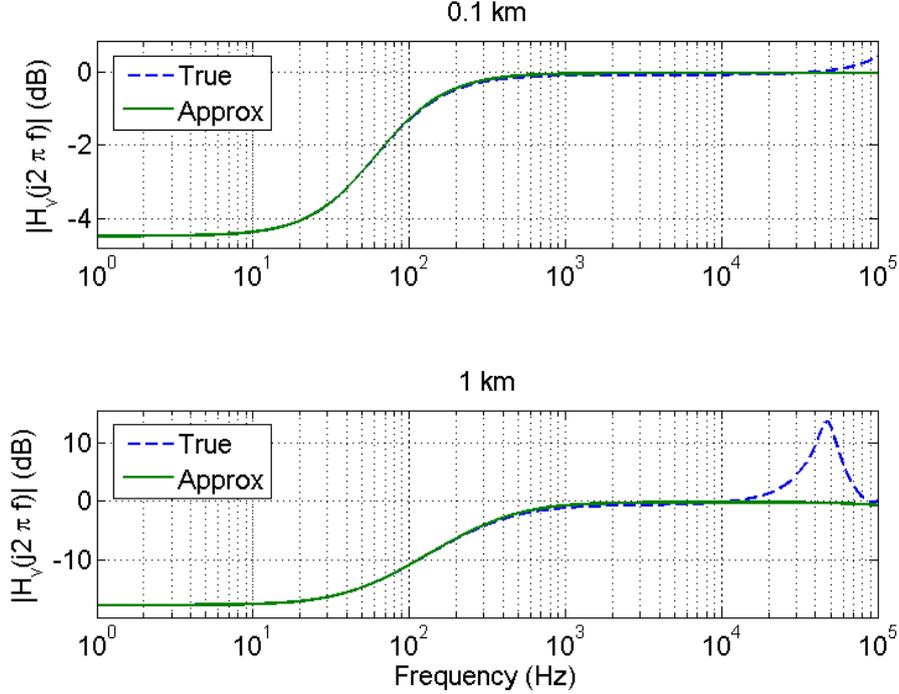


Figure 4.5: Comparison of the frequency responses of  $H_v(s)$  using the complete impedance expressions and the approximation, for two cable length extremes

sible to obtain an estimate of the motor-side voltage since the following relation holds:

$$u_{mot_j,k} = u_{drv_j,k} - \frac{rh}{2} (i_{mot_j,k} + i_{drv_j,k}). \quad (4.25)$$

It should be remembered that this expression holds for the low frequency model depicted in Fig. 4.4. The signals in it, therefore, are limited to the band of frequencies for which the model is valid. Thus low-pass filtered versions of  $u_{drv_j,k}$ ,  $i_{mot_j,k}$  and  $i_{drv_j,k}$  are, therefore, required. It is technically difficult to directly measure  $u_{drv_j,k}$ , since it is a high-voltage signal; nevertheless, a low-pass filtered version of  $u_{drv_j,k}$  is available since the computed control action is equal to the average voltage over two PWM chopping peri-

ods.  $i_{mot,j,k}$  is available in its estimated form from the previously presented motor-side current estimator, it will have a bandwidth equal to the closed-loop bandwidth of the current controller.  $i_{drv,j,k}$  can contain non-negligible frequency components at the PWM harmonic frequencies, as seen in Fig. 4.1, which violate the model's assumptions. Nonetheless these frequencies are greater than the frequency band of interest so they can be filtered out either analogically or digitally before the signal is used in the estimator expression (4.25). In our case, the anti-aliasing filter ensures this assumption is met, see Fig. 2.10.

**Remark:** The additional computing resources required for the above described estimators is less than would be needed for an EKF model incorporating even the low frequency cable dynamics. This is because the DSP is optimized to run Infinite Impulse Response (IIR) filters, such as  $G_{est}(z)$ , very efficiently, whereas the matrix manipulations required by the EKF are computationally very costly, especially for large matrices. The EKF incorporating these dynamics would require 7 states and a more complex measurement state equation making it practically unfeasible.

### 4.3.3 Integration in the existing software architecture

The time step  $T$  for the EKF has to be equal to the sampling time of the controller and thus the operations needed to complete an estimation update (4.16 and 4.18) can be included in the *Controller Interrupt* routine described in Chapter 3.

In Fig. 4.6 a block diagram for the implemented algorithm is presented.

With this configuration, an EKF based observer becomes feasible for implementation and on-line operation.

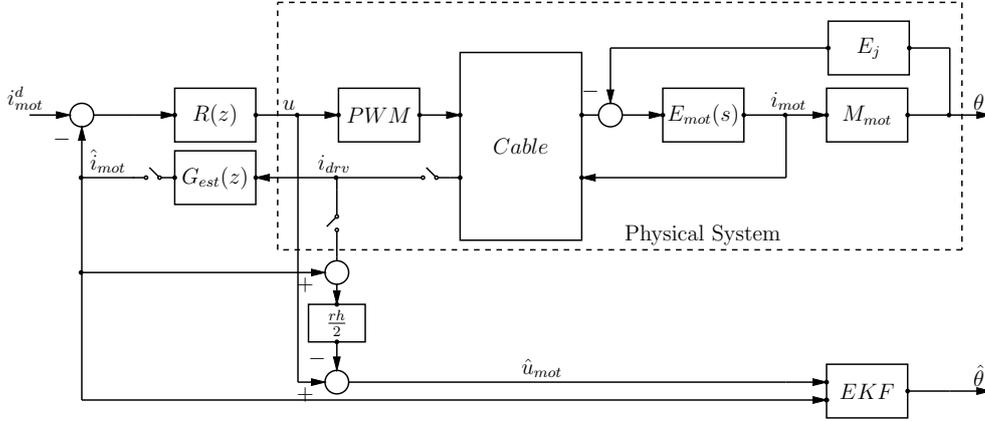


Figure 4.6: Block diagram for the complete system.

## 4.4 Implementation details

In this section the implementation details for the *EKF* block are reported.

The EKF algorithm is naturally divided into two tasks: prediction (4.16) and correction (4.18). Along with these the Jacobian matrix defined in 4.17 has to be updated at each time step.

Many properties of the matrices involved in these steps are preserved through time steps and can thus be exploited to simplify calculations and lighten the computational burden.

From the system equations reported in 4.20 and the definition of the Jacobian matrix in 4.17, it is evident that the values for  $\cos px_{4k}$  and  $\sin px_{4k}$  need to be calculated a total of 6 times each within the prediction step. It has been found more efficient to call the *sine* and *cosine* functions once and store the result in temporary variables to minimize the function call overhead.

From the definition of  $\mathbf{P}_{k+1|k+1}$  given in Sec. 4.1.3 it follows the non negativity and symmetry of this matrix. It is crucial for EKF stability that this properties are preserved from one step to the next; numerical errors could endanger this requirement and therefore a particular care should be taken. A common practice is to add a step to Eq. 4.18 to calculate the final value for  $\mathbf{P}_{k+1|k+1}$  as  $(\mathbf{P}_{k+1|k+1} + \mathbf{P}_{k+1|k+1}^T)/2$ . Nevertheless a better

solution [24] is to propagate only half the elements of  $\mathbf{P}_{k+1|k+1}$  - say the ones on and above the main diagonal. This also naturally shortens the execution time as (for our particular application) only 15 values need to be calculated as opposed to the 25 normally required.

As follows from its definition recalled in Sec. 4.1.4, the  $\mathbf{H}_k$  matrix for our application is:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

independent of the time step  $k$ . Thus left-multiplying by  $\mathbf{H}$  corresponds to the extraction the first two rows from the right operand; conversely right-multiplying by  $\mathbf{H}^T$  means copying to the result the first two columns of the left operand. This allows, for example, the reduction of the calculation of the quantity  $\mathbf{H}\mathbf{P}_{k+1|k}\mathbf{H}^T$  in Eq. 4.18 to copying to the result the upper-left  $2 \times 2$  submatrix of  $\mathbf{P}_{k+1|k}$  thus having to perform 4 store operation instead of 14 multilpy and add operations.

Along with the exploitation of the fact that both  $\mathbf{R}$  and  $\mathbf{Q}$  are diagonal and thus only few summations are needed in Eq. 4.16 and Eq. 4.18 to include their contribution, some more *brute force* techniques [33] have been employed to fulfill the required execution time constraints (an EKF time step has to fit in the *Controller Interrupt* routine). All matrices have been stored in single dimension arrays and the multiplications have been explicited elementwise using pre-compiler macros and inline functions.

# 5 Identification and Tuning procedure

*Computers make it easier to do a lot of things, but most of the things they make it easier to do don't need to be done.*

---

Andy Rooney

**The automatic procedure to identify system parameters and tune Kalman covariance matrices is presented.**

In its 50 years of existence the Kalman filter has been employed for state estimation and sensor fusion in numerous applications. Despite this widespread use, however, the selection of the filter's parameters, more specifically the noise covariance matrices, remains fairly undocumented. Many hundreds of publications discussing the theoretical aspects of the Kalman filter and its extensions have been written, but only a small fraction of these cover the important topic of filter tuning in detail.

Whilst, in certain cases the measurement noise's covariance matrix can be estimated relatively straightforwardly, the process noise's covariance matrix is notoriously difficult to obtain, since its unique contribution is not directly measurable. Often, therefore, the physical significance of this term is ignored and it is just considered as a tuning parameter to achieve the desired performance of the Kalman filter.

The publications that mention filter tuning propose methods that can be roughly separated into 3 groups:

1. By trial-and-error: The process noise covariance matrix can be interpreted as quantifying the amount of uncertainty that exists in the system model. It therefore influences how much weight the Kalman filter places on the state prediction from the model compared to the measurements. Using this concept, if the measurement noise covariance matrix is fixed, the elements of the process noise covariance matrix can be adjusted on a trial-and-error basis in simulation in order to achieve the desired state tracking. Being a trial-and-error method, this option is potentially very time-consuming and dependent on the expertise of the tuner.
2. By optimisation: Using a one off set of measured data of the key states to be estimated, an optimisation problem can be solved to minimise a cost function that penalises the state estimation error e.g. [34–38].
3. Tuning using heuristic rules e.g. [39].

The majority of these publications considering the Kalman filter tuning step assume that a model of the system, as also required by the Kalman filter, has already been identified. The estimation of the model's parameters, in itself, is also a nontrivial procedure [40].

A procedure for the simultaneous parameter estimation and covariance matrix tuning is presented here. This procedure renders the tuning of an EKF for a sensorless stepper motor drive almost automatic and greatly reduces the filter setup time since trial-and-error tuning is eliminated. It is also more likely to produce higher performance state estimation than tuning based on heuristic rules, since these must be robust enough to work for the general case.

## 5.1 Input and measurement estimation

The inputs and measurements considered in the model described by 4.20 are the motor-side voltages and currents, respectively. As mentioned in the previous chapters, these signals are not directly available in ordinary operation as they are separated from the drive electronics by long cables. Estimators, acting as noise whitening filters, are therefore required.

A cable length adaptive digital filter  $G_{est}(z)$  is therefore proposed in chapter 2 to produce optimal estimates of the motor-side currents  $\hat{i}_{mot_j,k}$  from measurements of the drive-side currents  $i_{drv_j,k}$ . The motor-side voltage estimate is given by the following expression:

$$\hat{u}_{mot_j,k} = u_{drv_j,k} - \frac{rh}{2} \left( \hat{i}_{mot_j,k} + i_{drv_j,k} \right), \quad (5.1)$$

where  $r$  is the cable resistance per unit length,  $h$  is the cable length and  $u_{drv_j,k}$  is the drive-side voltage.

The use of the motor-side current estimates as the measurements in the EKF implies:

$$\begin{aligned} \mathbf{y}_k &= \begin{bmatrix} \hat{i}_{mot_A,k} + n_{A,k} \\ \hat{i}_{mot_B,k} + n_{B,k} \end{bmatrix} & (5.2) \\ &= \begin{bmatrix} i_{mot_A,k} + n_{A,k}^{umd} + n_{A,k} \\ i_{mot_B,k} + n_{B,k}^{umd} + n_{B,k} \end{bmatrix} \\ &= \begin{bmatrix} i_{mot_A,k} + v_{A,k} \\ i_{mot_B,k} + v_{B,k} \end{bmatrix} & (5.3) \end{aligned}$$

where  $n_{A,k}$  and  $n_{B,k}$  are the measurement noises on the sampled drive-side currents filtered by  $G_{est}(z)$ , and  $n_{A,k}^{umd}$  and  $n_{B,k}^{umd}$  are the noise processes due to the unmodelled dynamics between the motor- and drive-side currents that are not captured by  $G_{est}(z)$ .  $v_{A,k}$  and  $v_{B,k}$  are the sums of the two noise sources, whose covariance matrix is  $\mathbf{R} = \mathbf{R}_{umd} + \mathbf{R}_i$ , since the two noise sources can be assumed uncorrelated.

Similarly, any unmodelled dynamics in the input estimation procedure should be included in the process noise and thus its covariance matrix  $\mathbf{Q}$ .

Since the noise processes due to the unmodelled dynamics in the estimators are cable length dependent, so  $\mathbf{R}$  and  $\mathbf{Q}$  are functions of the cable length.

## 5.2 Automatic parameter estimation and tuning procedure

The procedure for the estimation of the model parameters and noise covariance matrices is composed of five sequential stages:

1. Initial  $\mathbf{R}$  estimation
2. Linear model parameter estimation
3. Nonlinear model parameter estimation
4. Initial  $\mathbf{Q}$  estimation
5. Global parameter estimation

Apart from the first step, which requires data from its own specific experiment, this procedure runs sequentially on the same set of training data obtained from a single experiment. This training data should contain the estimated inputs and measurements used by the EKF, but also low noise measurements of the most important states to be estimated by the EKF i.e the angle and load torque. These state measurements will obviously not be available in normal operation, hence the need for their estimation, nonetheless they should be provided for the one-off tuning procedure.

### 5.2.1 Details

#### Step 1: Initial $\mathbf{R}$ estimation

The measurement noise covariance matrix,  $\mathbf{R} \in \mathbb{R}^{2 \times 2}$ , has two components  $\mathbf{R}_i$  and  $\mathbf{R}_{umd}$ . Nonetheless, under the assumption that it is possible to

carry out a one-off, special experiment where the drive-side and motor-side currents can be measured synchronously, the combined covariance matrix,  $\mathbf{R}$  can be estimated. Bearing in mind that the measurements, both motor and drive-side, will be contaminated by measurement noise, we can calculate the difference between the estimated and measured motor-side current as:

$$\begin{aligned}\epsilon_{mot_j,k} &= (\hat{i}_{mot_j,k} + n_{j,k}) - (i_{mot_j,k} + n_{mot_j,k}) \\ &= n_k^{umd} + n_{j,k} - n_{mot_j,k} \\ &= v_{j,k} - n_{mot_j,k}.\end{aligned}\tag{5.4}$$

If all the noises can be considered uncorrelated and stationary, and the motor-side current measurement noise variances,  $\sigma_{n_{mot_j}}^2$ , can be estimated using data from a separate experiment, then the measurement noises' variances can still be evaluated since:

$$\sigma_{v_j}^2 = \sigma_{\epsilon_j}^2 - \sigma_{n_{mot_j}}^2.$$

$\mathbf{R}$  can then be taken as

$$\mathbf{R} = \begin{bmatrix} \sigma_{v_A}^2 & 0 \\ 0 & \sigma_{v_B}^2 \end{bmatrix}.\tag{5.5}$$

### Step 2: Linear model parameter estimation

Despite the stepper motor being a nonlinear system, as seen from its modelling equations (2.3) - (2.4), the electrical and mechanical sub-systems are linearly dependent on their parameters. This property means that each sub-system's equations can be written in linear regression form and thus convex cost functions can be minimised to estimate the globally optimal parameters. This scheme is similar to the 2-stage approach proposed in [41], however, here load and detent torque terms are also included.

From Eq. (2.3) we have, for discrete time instants  $t = t_1, t_2, \dots, t_N$ :

$$\mathbf{U} = \begin{bmatrix} u_{mot_A}(t_1) \\ u_{mot_B}(t_1) \\ u_{mot_A}(t_2) \\ u_{mot_B}(t_2) \\ \vdots \\ u_{mot_A}(t_N) \\ u_{mot_B}(t_N) \end{bmatrix} = \mathbf{D}_e \boldsymbol{\rho}_e$$

where

$$\mathbf{D}_e = \begin{bmatrix} \frac{di_{mot_A}(t_1)}{dt} & i_{mot_A}(t_1) & -\omega(t_1) \sin p\theta(t_1) \\ \frac{di_{mot_B}(t_1)}{dt} & i_{mot_B}(t_1) & \omega(t_1) \cos p\theta(t_1) \\ \frac{di_{mot_A}(t_2)}{dt} & i_{mot_A}(t_2) & -\omega(t_2) \sin p\theta(t_2) \\ \frac{di_{mot_B}(t_2)}{dt} & i_{mot_B}(t_2) & \omega(t_2) \cos p\theta(t_2) \\ \vdots & \vdots & \vdots \\ \frac{di_{mot_A}(t_N)}{dt} & i_{mot_A}(t_N) & -\omega(t_N) \sin p\theta(t_N) \\ \frac{di_{mot_B}(t_N)}{dt} & i_{mot_B}(t_N) & \omega(t_N) \cos p\theta(t_N) \end{bmatrix} \quad (5.6)$$

and

$$\boldsymbol{\rho}_e = [L, R, K_m]^T. \quad (5.7)$$

The electrical system parameters can then be estimated by minimising a least squares criterion i.e.

$$\boldsymbol{\rho}_e^N = \arg \min_{\boldsymbol{\rho}_e} (\mathbf{U} - \mathbf{D}_e \boldsymbol{\rho}_e)^T (\mathbf{U} - \mathbf{D}_e \boldsymbol{\rho}_e) \quad (5.8)$$

Similarly for the mechanical part, referring to (2.4), we can write:

$$\boldsymbol{\tau}_{res} = \begin{bmatrix} \tau_{em}(t_1) - \tau_l(t_1) \\ \tau_{em}(t_2) - \tau_l(t_2) \\ \vdots \\ \tau_{em}(t_N) - \tau_l(t_N) \end{bmatrix} = \mathbf{D}_m \boldsymbol{\rho}_m$$

where  $\boldsymbol{\tau}_{\text{res}}$  is the resultant torque, which should be available from measurements for this step,

$$\mathbf{D}_m = \begin{bmatrix} \frac{d\omega(t_1)}{dt} & \omega(t_1) & \sin 2p\theta(t_1) & \cos 2p\theta(t_1) \\ \frac{d\omega(t_2)}{dt} & \omega(t_2) & \sin 2p\theta(t_2) & \cos 2p\theta(t_2) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{d\omega(t_N)}{dt} & \omega(t_N) & \sin 2p\theta(t_N) & \cos 2p\theta(t_N) \end{bmatrix}$$

and

$$\boldsymbol{\rho}_m = [J, B, \rho 1_m, \rho 2_m]^T \quad (5.9)$$

Implying:  $T_{dm} = \sqrt{\rho 1_m^2 + \rho 2_m^2}$  and  $\phi = \arctan(\rho 1_m / \rho 2_m)$ .

As before the mechanical parameters can be estimated by minimising a least squares criterion:

$$\boldsymbol{\rho}_m^N = \arg \min_{\boldsymbol{\rho}_m} (\boldsymbol{\tau}_{\text{res}} - \mathbf{D}_m \boldsymbol{\rho}_m)^T (\boldsymbol{\tau}_{\text{res}} - \mathbf{D}_m \boldsymbol{\rho}_m) \quad (5.10)$$

### Step 3: Nonlinear model parameter estimation

When the system's equations are written in linear regression form, what is minimised is not a cost function based on signals of key importance, such as the angular position. An optimisation problem minimising a cost function based on these signals can be defined as:

$$\begin{aligned} \boldsymbol{\rho}_{pe}^N &= \arg \min_{\boldsymbol{\rho}_{pe}} \sum_{k=0}^{N-1} \boldsymbol{\epsilon}_{pe,k}(\boldsymbol{\rho}_{pe})^T \mathbf{W}_{pe} \boldsymbol{\epsilon}_{pe,k}(\boldsymbol{\rho}_{pe}) \\ \text{s.t. } \boldsymbol{\rho}_{pe} &\in \mathbb{R}^{n_{pe}} \text{ and } \mathbf{x}_{pe,k+1} = f_{pe}(\mathbf{x}_{pe,k}, \mathbf{u}_k, \tau_{l,k}, \boldsymbol{\rho}_{pe}), \end{aligned} \quad (5.11)$$

where

$$\begin{aligned} \boldsymbol{\rho}_{pe} &= [R, L, K_m, J, B, T_{dm}, \phi]^T, \\ \boldsymbol{\epsilon}_{pe,k}(\boldsymbol{\rho}_{pe}) &= \mathbf{x}_{m,k}^{pe} - \mathbf{x}_{pe,k}(\boldsymbol{\rho}_{pe}), \end{aligned} \quad (5.12)$$

$\mathbf{x}_{m,k}^{pe}$  is vector of measured or estimated states at instant  $k$ ,  $\mathbf{W}_{pe}$  is a weighting matrix and

$$\begin{aligned} \mathbf{x}_{pe,k+1}(\boldsymbol{\rho}_{pe}) &= \begin{bmatrix} i_{mot_A,k+1} \\ i_{mot_B,k+1} \\ \omega_{k+1} \\ \theta_{k+1} \end{bmatrix} = f_{pe}(\mathbf{x}_{pe,k}, \mathbf{u}_k, \tau_{l,k}, \boldsymbol{\rho}_{pe}) \\ &= \begin{bmatrix} (1 - \frac{TR}{L}) i_{mot_A,k} + \frac{TK_m}{L} \omega_k \sin p\theta_k + \frac{T}{L} u_{mot_A,k} \\ (1 - \frac{TR}{L}) i_{mot_B,k} - \frac{TK_m}{L} \omega_k \cos p\theta_k + \frac{T}{L} u_{mot_B,k} \\ -\frac{TK_m}{J} (i_{mot_A,k} \sin p\theta_k - i_{mot_B,k} \cos p\theta_k) \\ \quad + (1 - \frac{TB}{J}) \omega_k - \frac{TT_{dm}}{J} \sin(2p\theta_k + \phi) - \frac{T}{J} \tau_{l,k} \\ \theta_k + T\omega_k \end{bmatrix} \end{aligned} \quad (5.13)$$

and  $n_{pe}$  is the dimension of  $\boldsymbol{\rho}_{pe}$ . It should be noted that in this minimisation the load torque is taken as an input to the model, unlike in the EKF model. This minimisation is a nonlinear optimisation problem, for which initial parameter estimates are required. These values can be taken as the estimates from the previous linear identification step.

#### Step 4: Initial $\mathbf{Q}_k$ estimation

The next step involves obtaining initial estimates for the process noise covariance matrix  $\mathbf{Q} \in \mathbb{R}^{5 \times 5}$ . The process noises are assumed stationary and uncorrelated, thus the covariance matrix becomes time step independent and diagonal. The prediction error vector from the previous step, Eq. (5.12), can be considered as representing a vector of realisations of the process noises for the first 4 states in the model (4.20). Since in Kalman Filter theory it is assumed that these realisations come from normally distributed processes, 99.7% of their values should be within  $3\sigma_l$ , where  $\sigma_l$  is the standard deviation of the  $l^{\text{th}}$  state's noise process. Based on this fact the variance term for each state is taken as  $\sigma_l^2 = (\max_k |\boldsymbol{\epsilon}_{pe,k}(l)|/3)^2$ , where  $\boldsymbol{\epsilon}_{pe,k}(l)$  is the  $l^{\text{th}}$  element of  $\boldsymbol{\epsilon}_{pe,k}$ . Similarly, a realisation of the process noise driving

the load torque state is evaluated as:

$$w_{5,k} = \tau_{l,k+1} - \tau_{l,k}, \quad (5.14)$$

based on the discretised version of the assumed random walk model of the load torque. Its variance is then estimated as  $\sigma_5^2 = (\max_k |w_{5,k}|/3)^2$ .

It should be mentioned that the measurement noise on the measured states is assumed negligible compared to the process noise.

### Step 5: Global parameter estimation

With initial estimates of all of the EKF parameters available, a logical final step is to carry out a global estimation of the complete parameter set. This can be achieved by defining an optimisation problem that minimises a cost function based on the EKF's estimates i.e.

$$\begin{aligned} \boldsymbol{\rho}_{ekf}^N &= \arg \min_{\boldsymbol{\rho}_{ekf}} \sum_{k=0}^{N-1} \boldsymbol{\epsilon}_{ekf,k}(\boldsymbol{\rho}_{ekf})^T \mathbf{W}_{ekf} \boldsymbol{\epsilon}_{ekf,k}(\boldsymbol{\rho}_{ekf}) \\ \text{s.t. } \boldsymbol{\rho}_{ekf} &\in \mathbb{R}^{n_{ekf}} \text{ and (4.16) - (4.18),} \end{aligned} \quad (5.15)$$

where

$$\begin{aligned} \boldsymbol{\rho}_{ekf} &= [R, L, K_m, J, B, T_{dm}, \boldsymbol{\phi}, Q_{11}Q_{22}, \dots, Q_{55}, R_{11}R_{22}]^T, \\ \boldsymbol{\epsilon}_{ekf,k}(\boldsymbol{\rho}_{ekf}) &= \mathbf{x}_{m,k}^{ekf} - \hat{\mathbf{x}}_{k|k}(\boldsymbol{\rho}_{ekf}), \end{aligned} \quad (5.16)$$

$\mathbf{x}_{m,k}^{ekf}$  is a vector of measured or estimated states at instant  $k$ ,  $\mathbf{W}_{ekf}$  is a weighting matrix and  $\hat{\mathbf{x}}_{k|k}(\boldsymbol{\rho}_{ekf})$  is the a posteriori estimate produced by the EKF using parameters  $\boldsymbol{\rho}_{ekf}$  and  $n_{ekf}$  is the dimension of  $\boldsymbol{\rho}_{ekf}$ .

Again this optimisation problem is nonlinear and so many local minima exist, meaning good initialisation of the decision variables is imperative. Initial values for  $\hat{\mathbf{x}}_{0|0}$ , the state estimate at time instant  $k = 0$ , and  $\mathbf{P}_{0|0}$ , the state estimate covariance matrix at time instant  $k = 0$  are also required.  $\hat{\mathbf{x}}_{0|0}$  can be set using the measured values at  $k = 0$ , since in the tuning procedure state measurements are available. Obviously, an initial state

---

estimation scheme will be needed in normal operation. A simple way to initialise  $\mathbf{P}_{0|0}$  is to set it equal to a multiple of the process noise covariance matrix i.e.  $\mathbf{P}_{0|0} = \alpha^2 \mathbf{Q}_k$  where  $\alpha$  is a scalar [42].

## 6 Experimental results

*However beautiful the strategy,  
you should occasionally look at  
the results.*

---

Winston Churchill

**Experimental results for the open-loop observer based solution are summarized in terms of position repeatability and electromagnetic emissions. Experimental results on the estimation capabilities, numerical stability and execution time for the EKF based solution are reported.**

In sections 2.3.3 and 4.3.3 two loop configurations to control a stepping motor connected to its drive electronics and sensors via arbitrarily long cables have been presented.

Due to demanding constraints on the amplitude and spectral contents of electromagnetic emissions in the LHC tunnel (the foreseen operating environment) a nonlinear actuator is necessary to modulate at relatively high frequencies the controlling voltages. The response of the connecting cables to these high-power, modulated signals makes it impossible to rely on the phase currents, as measured on the drive side of the cables to close the control loop. Furthermore, the radioactivity of the installation environment does not permit the installation of any electronic sensor in the proximity of the motor.

A first open-loop observer based solution, outlined in chapter 3, is based on the idea that, if it is possible to map the driver side current to the motor side current (i.e. to invert the cable's transfer function) then one can exploit the physical structure of hybrid stepping motors and close the control loop on the phase current as estimated by the mapping function. Results for this solution are presented in terms of positioning repeatability and electromagnetic emissions.

The second proposed solution, EKF based, relies on the assumption that the information available driver side is sufficient to implement an observer for the motor's state. This configuration is implemented in view of its application with a state feed-back controller. Results in this case refer to the estimation capabilities of the developed state observer.

## 6.1 Open-loop observer based solution

In this section, results are summarized in both a qualitative and quantitative way. Whenever possible, the comparison with commercially available *off the shelf* equivalent instrumentation is given.

### 6.1.1 Position repeatability

The repeatability  $Rep$  for a given random variable is normally expressed percentually and is defined as:

$$Rep = \frac{\sigma}{\mu} \quad (6.1)$$

where  $\mu$  is the expected value of the considered random variable and  $\sigma$  is its standard deviation. These figures need to be calculated from a statistically relevant population sample, in fact, their unbiased estimators (sample average and standard deviation) are used in real life.

$Rep$  gives a normalized measure of how often a given variable would *fall in the same bin* if an histogram of its values was made. As follows from

its definition, the smaller the  $Rep$  value, the higher the repeatability of the considered variable.

This measure, if calculated for the relative angular position of the stepping motor, is of high interest for our application. If our control loop guarantees that a motor step is repeatable, then the open-loop observer based solution is reliable and applicable to the problem presented in the first chapters.

This quantity depends on the third harmonic correction, but, above all, it is a figure of merit for the motor side current controller and motor side current estimator.

### Experimental set-up

To measure this quantity a dedicated LabVIEW VI has been developed. The measurement is performed according to the following steps:

1. The current shaft angular position is measured,
2. A step request is sent to the driver.
3. A pre-programmed delay (usually in the order of hundreds of milliseconds) takes place to let all the mechanical transients expire.
4. The shaft angular position is measured again so that a  $\Delta - step$  can be evaluated.

This procedure is repeated a number of times (at least 3 mechanical turns are performed so that a few hundreds of steps are recorded ensuring a statistically significant population sample is available to estimate  $\mu$  and  $\sigma$ ) and a histogram is created.

### Experimental results

The position repeatability is influenced by the third harmonic correction described in equation 2.7. Figure 6.1 shows qualitatively this effect. It can

be noted how the distribution changes from bimodal to monomodal as the best third harmonic correction value is approached. It is evident that the position error is more repeatable in the central histogram.

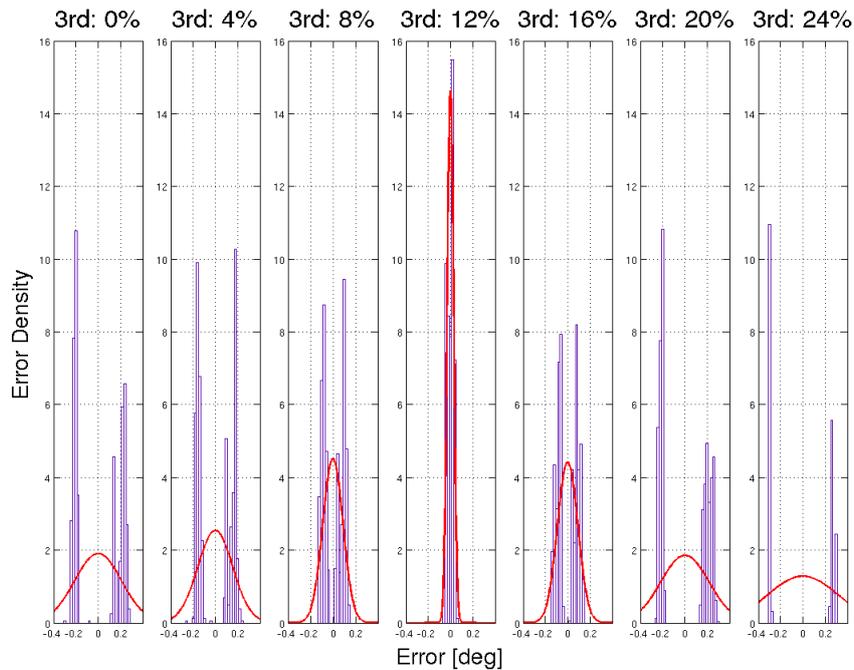


Figure 6.1: Qualitative evaluation of the third harmonic correction effects on the position repeatability.

The third harmonic correction that minimizes the positioning repeatability has been evaluated experimentally and found to be  $120 \text{ mA/A}$  for quarter step mode and  $20 \text{ mA/A}$  for half-step mode.

In table 6.1 a comparison between our driver, named EDA-02000-V1-0 and three commercially available equivalent drivers is reported. Obviously this comparison is made with a cable connecting the driver and the motor phases less than  $5\text{m}$  long. All off the shelf drives, in fact, fail to track any current reference if connected to the motor with long cables.

All drives were tested controlling a Maccon stepping motor in quarter-step mode. All drivers under test were configured with the same motor and controller parameters, reported in table 6.2.

<b>Driver name</b>	<b><i>Rep</i></b>
EDA-02000-V1-0	3.64%
Parker-Gemini GT-L5	3.91%
LAM Technologies DS-1044	9.65%
HCR SHS Star 2000	12.91%

Table 6.1: Static position repeatability benchmark without a cable.

<b>Winding Resistance</b>	<b>Winding Inductance</b>	<b>Closed Loop bandwidth</b>
3.7 $\Omega$	30 <i>mH</i>	500 <i>Hz</i>

Table 6.2: Controller parameters.

These results allow us to claim that the driver designed, realized and programmed internally is equipped with a motor-side current controller that outperforms those installed in commercially available drives.

The very same test has been carried out using a long cable to connect the driver to the motor phases. Only two cable lengths (namely 540*m* and 720*m*) can be installed in the laboratory set-up.

Table 6.3 summarizes the achieved repeatability for these two cable lengths and 3 different stepping modes.

<b>Stepping mode</b>	<b>Cable Length [<i>m</i>]</b>	<b><i>Rep</i> [%]</b>
Full Step	540	4.4
Full Step	720	4.0
Half Step	540	4.2
Half Step	720	4.5
Quarter Step	540	4.5
Quarter Step	720	4.7

Table 6.3: Static position repeatability with the cable.

Our driver outperforms 2 out of 3 commercially available equivalent drivers even when connected to the motor phases with very long cables proving the validity of the proposed implementation scheme.

It is worth noting that there is enough statistics to claim that these figures are characteristic of the driver. In fact the values shown are calculated as the step repeatability over 4 mechanical turns. This means 800, 1600 and 3200 steps were used for full, half and quarter step modes respectively.

### 6.1.2 Electromagnetic emissions

All equipment installed in the LHC tunnel must satisfy demanding requirement on electromagnetic compatibility. From electromagnetic field theory it is known that the cables connecting the drive to the stepping motors are a potentially harmful source of electromagnetic emissions [43].

It is required for these emissions to have negligible components at frequencies below  $20kHz$ .

#### Experimental set-up

Figure 6.2 shows the laboratory set up to measure such emissions is shown. It is composed of a spiral antenna capturing emissions from a branch of the long cable and an Agilent superheterodyne spectrum analyzer to process the signals on line [44].

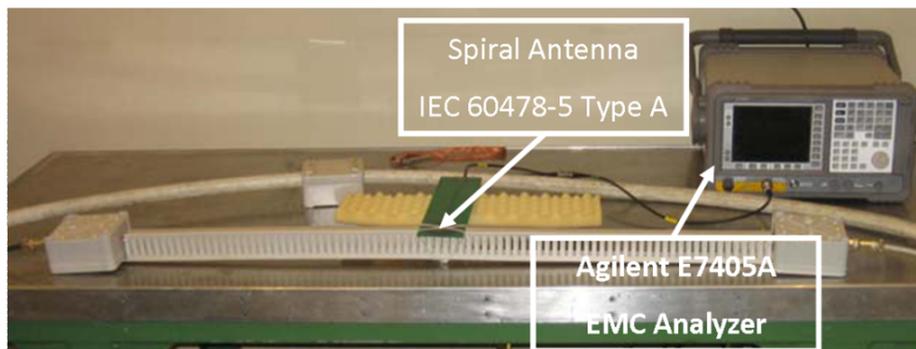


Figure 6.2: Emission measurement test-bench.

To perform these measurements, a complete LabVIEW VI [44] has been developed to simultaneously generate step requests for the driver and acquire signals as measured and processed by the spectrum analyzer.

### Experimental results

In figure 6.3 a comparative plot for the amplitude of the magnetic field  $H$  is shown. The prototype developed internally and a version of the HCR SHS Star 2000 driver, modified to work with the cable, are considered.

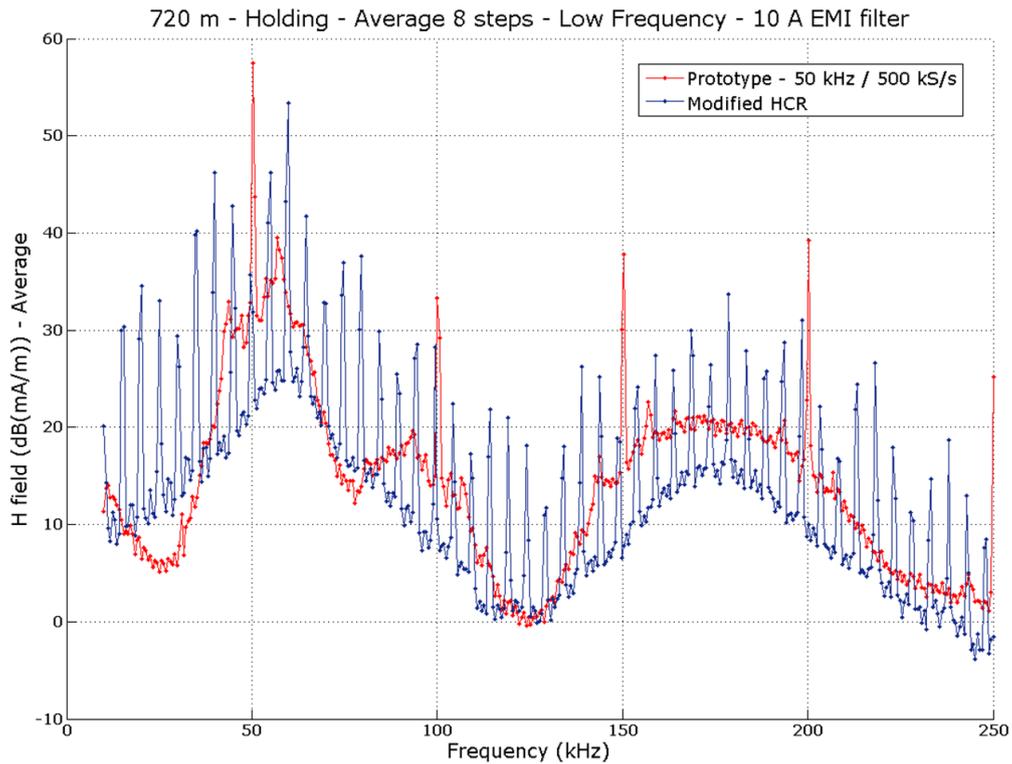


Figure 6.3: Comparative plot for the emitted H-field spectrum.

It is evident that the prototype meets the electromagnetic compatibility requirement, having the first harmonic component of the emitted magnetic field well above the  $20\text{kHz}$  threshold.

This implies the choice of modulating the control action with a  $50\text{kHz}$  PWM signal is enough to ensure design constraints are met.

## 6.2 EKF-based solution results

In this section we report the results obtained in terms of estimation capabilities (with special focus on angular position tracking), execution time and numerical stability for the developed EKF-based observer. Despite the necessity to present them in a distinct and clear manner it should be noted that these three aspects are closely related. Indeed numerical stability is necessary for the boundedness of the estimation error. Likewise, a short execution time ensures all dynamics of interest (including rather fast ones) are reproduced.

### 6.2.1 Numerical stability

The numerical stability for the developed observer has been empirically tested by wrapping the DSP C code in a MEX function and developing an equivalent MATLAB *m-file*. Both the considered DSP and the compiled MEX function make use of IEEE-754 floating point number representation whereas *m-files* are executed with double precision. In extensive testing (realized running the two functionally equivalent implementations on many data-sets) the differences between matrices' elements and estimated states have been found to be negligible, implying numerical stability has been achieved.

### 6.2.2 Execution time performances

Figure 6.4 shows two time-line plots, measured experimentally for the proposed implementation. In the first plot the DSP was programmed to keep a digital output line at a logical *one* while an EKF time step was being executed and to toggle a second digital output line each time the filter interrupt routine was called (see Chapter 3).

The second time-line plot shows the time needed to complete an EKF time step for both a trivially coded version of the algorithm and the proposed optimized implementation. The DSP was programmed to execute the

two functionally equivalent versions of the EKF one after the other waiting  $50\mu s$  in between.

It can be seen how, if not interrupted to estimate the motor side currents, the DSP can execute an EKF time step in  $16\mu s$  as opposed to the  $52\mu s$  required for the naive implementation considered.

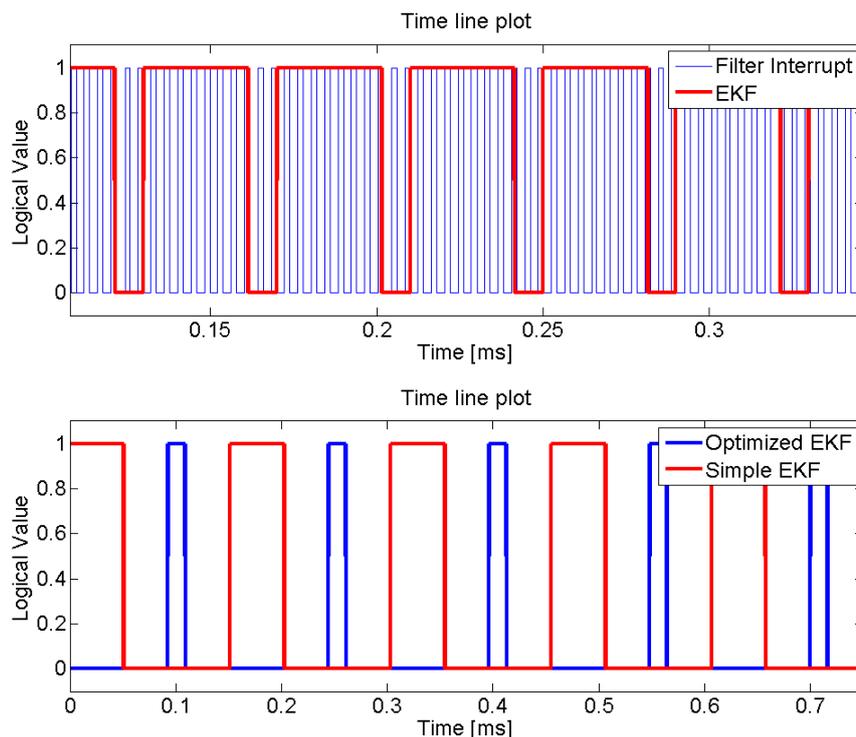


Figure 6.4: Time-lines as measured experimentally programming the DSP to toggle digital lines at specific events.

It is not trivial to benchmark the achieved execution speed as it is obviously dependent on the number of state variables of the considered system, on the DSP used for the application as well as on the number and complexity of tasks that need to be carried out aside the EKF algorithm.

Nevertheless, a brief literature survey [30, 37, 45, 46] shows that no previously proposed implementation achieved a sampling frequency of  $25kHz$  for an EKF tracking the state of a stepping motor.

### 6.2.3 State tracking performances

#### Experimental setup

A dedicated test-bench has been developed to evaluate the state tracking performances of the developed EKF-based observer.

In order to generate the data sets required to run the optimization procedure, the estimated control actions, motor phase currents and the shaft angular position have been acquired exploiting the prototype DAQ functionality described in chapter 3.

Using one of the available DSP output lines, these acquisitions have been synchronized with that of the output of a Scaime torque meter mounted on the motor shaft. The synchronous acquisition has been performed with a dedicated LabVIEW VI and by making use of a National Instruments PCI-6123 DAQ board. A synchronous motor acting on the same shaft as the stepper motor under test has been used as an active brake to apply a pre-programmed load torque that imitates the foreseen operating conditions. A sample data set is shown in figure 6.5.

The availability of currents and control actions, as measured or estimated directly by the DSP, guarantees that the noise generating processes preserve their characteristics from the identification/tuning phase to the final operation. This is an important aspect which has been disregarded in previous works on similar applications, e.g. [37].

#### Experimental results

Satisfactory state tracking performances have been achieved. The estimation error is surely influenced by the applied load torque but it does not grow without bounds and it is quickly recovered while the load torque is applied. We highlight that the chosen values for the load torque are not at all negligible: the motor under test has a nominal pull-in torque of  $2.5Nm$ , the the synchronous motor has been programmed to produce a constant load torque of  $-0.7Nm$  and, within the acquisition window, to induce a step

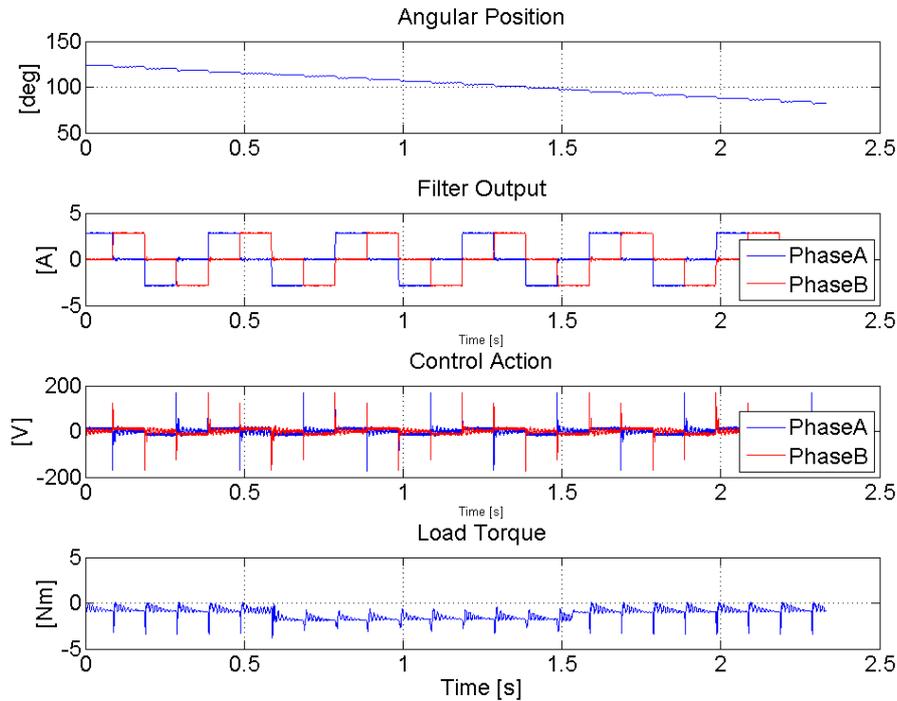


Figure 6.5: Sample data set.

down to  $-1.4Nm$  and then back to  $-0.7Nm$  (see figure 6.5).

Figure 6.6 shows the qualitative behavior of state tracking on a sample data set. Parameters for the system model and the EKF were set using the optimization procedure described in chapter 5.

Figure 6.7 shows a zoomed version on the two states of most interest i.e. the external torque and the shaft angular position.

For the designated application, particular attention is due to the position tracking capabilities of the complete observer. To assess the performances of the developed observer and the optimization procedure from this point of view 6 data-sets have been generated. The optimization procedure has been run on each one of them and the resulting observer has been validated on each data-set (including the one used to evaluate the various cost functions defined in the optimization procedure, henceforth referred to as the training

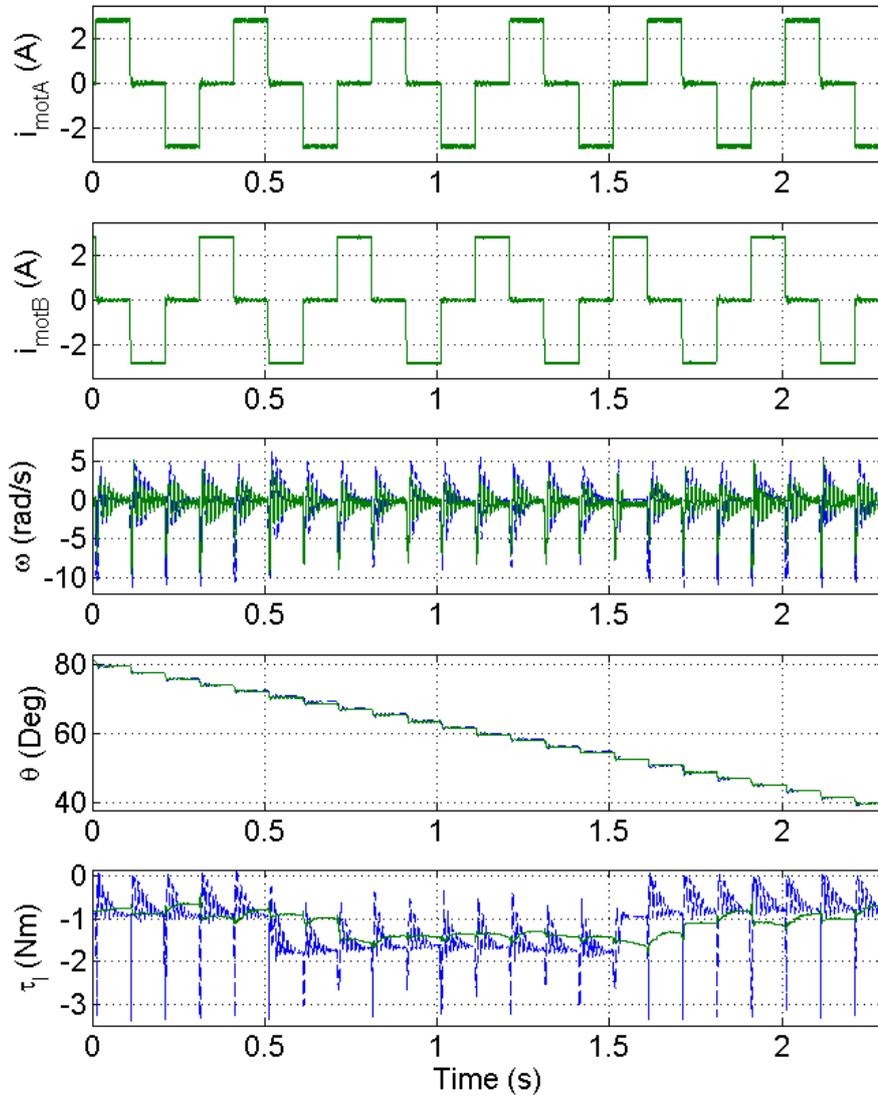


Figure 6.6: Measured (blue) and estimated (green) states. Training data set: 2. Validation data set: 1 (see table 6.4).

data set) using the MATLAB MEX code mentioned above.

Table 6.4 summarizes the position tracking performances with respect to the following performance factors:

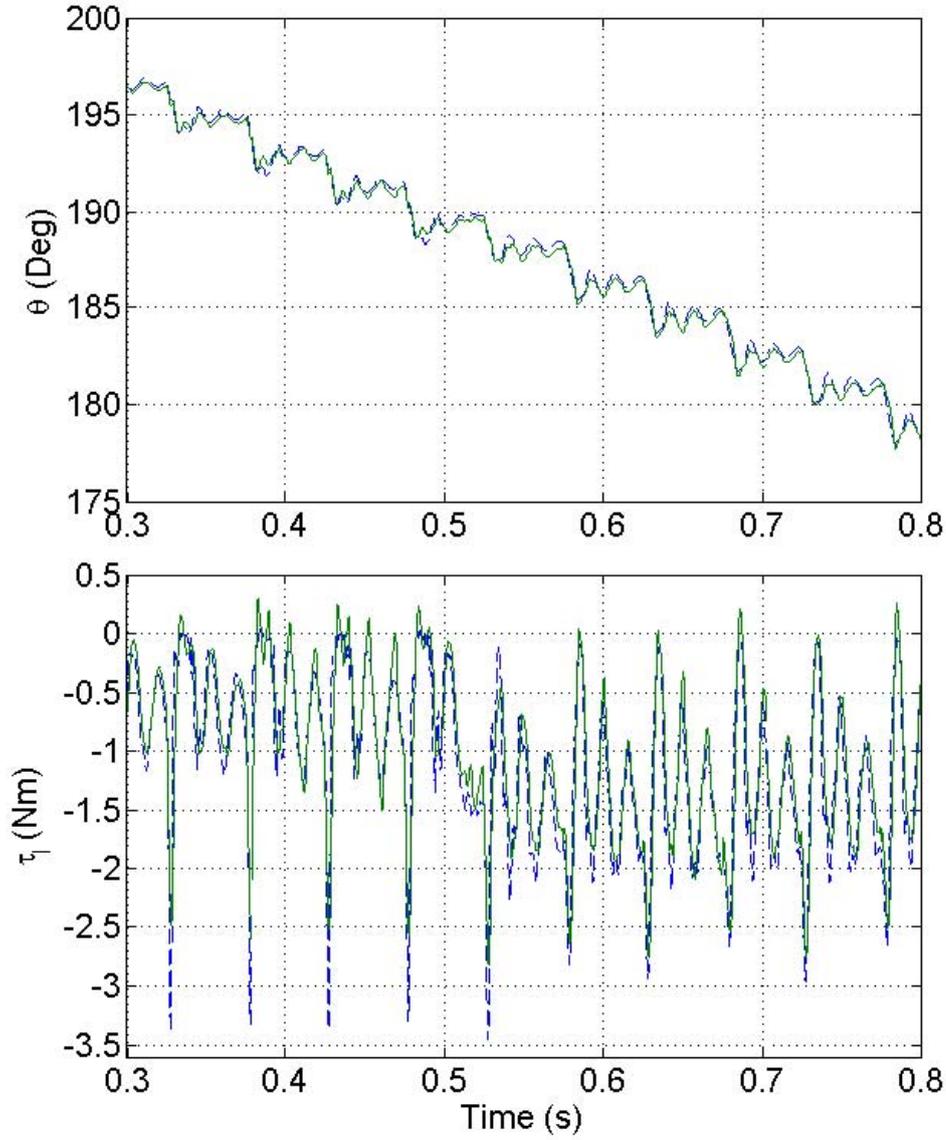


Figure 6.7: Measured (blue) and estimated (green) states. Zoom on external torque and position.

1. Mean error:  $\bar{e} = \frac{1}{N} \sum_{k=1}^N (\theta_k - \hat{\theta}_{k|k})$ .
2. Maximum error:  $e_{max} = \max_k |\theta_k - \hat{\theta}_{k|k}|$ .
3. RMSE:  $e_{rms} = \frac{1}{\sqrt{N}} \sqrt{\sum_{k=1}^N (\theta_k - \hat{\theta}_{k|k})^2}$ .

where  $\theta_k$  is the measured shaft angular position at instant  $k$  and  $\hat{\theta}_{k|k}$  is the shaft angular position as estimated by the EKF based observer at instant  $k$ .

Data-set		1	2	3	4	5	6
<b>1</b>	$\bar{e}$	0.0008	0.0009	0.0004	0.0002	-0.0015	-0.0013
	$e_{max}$	0.0440	0.0459	0.0457	0.0428	0.0447	0.0446
	$e_{rms}$	0.0073	0.0075	0.0083	0.0073	0.0092	0.0087
<b>2</b>	$\bar{e}$	0.0018	0.0032	0.0012	0.0073	0.0129	0.0111
	$e_{max}$	0.0240	0.0306	0.0259	0.0424	0.0511	0.0500
	$e_{rms}$	0.0058	0.0064	0.0064	0.0103	0.0151	0.0136
<b>3</b>	$\bar{e}$	-0.0016	-0.1716	-0.0022	-0.1910	-0.1427	-0.1785
	$e_{max}$	0.0262	1.0342	0.0300	1.0389	0.9191	1.0365
	$e_{rms}$	0.0056	0.3408	0.0064	0.3705	0.2994	0.3524
<b>4</b>	$\bar{e}$	0.0018	0.0035	-0.1123	-0.0386	0.0022	0.0025
	$e_{max}$	0.0722	0.0389	0.1657	0.1421	0.0375	0.0363
	$e_{rms}$	0.0089	0.0088	0.1178	0.0736	0.0081	0.0087
<b>5</b>	$\bar{e}$	0.0037	0.0035	0.0028	0.0024	-0.0004	0.0004
	$e_{max}$	0.0386	0.0404	0.0392	0.0371	0.0407	0.0486
	$e_{rms}$	0.0092	0.0089	0.0097	0.0087	0.0090	0.0091
<b>6</b>	$\bar{e}$	0.0022	0.0023	0.0080	0.0041	0.0051	0.1108
	$e_{max}$	0.0376	0.0411	0.0747	0.0862	0.0718	0.1937
	$e_{rms}$	0.0083	0.0084	0.0195	0.0157	0.0190	0.1199

Table 6.4: Position tracking performance *Montecarlo* results.

Rows span training data sets and columns span validation data sets. All figures are reported in radians and refer to data-sets generated with a 720m long cable connecting the motor to its driving electronics and sensors.

The optimization procedure described in chapter 5 has been implemented using the MATLAB environment exploiting both the optimization and the statistical toolboxes.

It took roughly 40 hours to calculate the 6 optimal parameters sets (one for each data-set) running on a dedicated node at the CERN LXPLUS facility. Each node is equipped with two Intel Xeon Processors E5420 (4 cores each) and 16GB of RAM.

In table 6.5 average and standard deviation are estimated for each performance factor reported in table 6.4. This is done along the 5 validation data-sets and for each training set.

<b>Data-set</b>	<b>Mean</b>	<b>Standard Deviation</b>
<b>1</b>	$\bar{e}$	-0.0002      0.0011
	$e_{max}$	0.0447      0.0012
	$e_{rms}$	0.0082      0.0008
<b>2</b>	$\bar{e}$	0.0069      0.0053
	$e_{max}$	0.0387      0.0130
	$e_{rms}$	0.0103      0.0041
<b>3</b>	$\bar{e}$	-0.1371      0.0778
	$e_{max}$	0.8110      0.4416
	$e_{rms}$	0.2737      0.1522
<b>4</b>	$\bar{e}$	-0.0204      0.0513
	$e_{max}$	0.0701      0.0555
	$e_{rms}$	0.0305      0.0488
<b>5</b>	$\bar{e}$	0.0026      0.0013
	$e_{max}$	0.0408      0.0046
	$e_{rms}$	0.0091      0.0004
<b>6</b>	$\bar{e}$	0.0043      0.0024
	$e_{max}$	0.0623      0.0216
	$e_{rms}$	0.0142      0.0055

Table 6.5: Position tracking performance data-set related results.

From these figures it is evident that our optimization procedure and implemented observer guarantee satisfactory positioning tracking performances in most cases (for a qualitative insight one should see figure 6.6). In fact only data set 3 produces a parameter set yielding unacceptable position tracking errors in validation. The fact that the optimization procedure does not always succeed is expected: some data-sets may not be informative enough for the given model set [40].

### 6.2.4 Identification and tuning procedure results

As mentioned in the previous section the identification and tuning procedure yields acceptable results in terms of state tracking. In particular, the position estimation error achieved is small enough to envision a controller built upon this information.

Nevertheless a few open issues remain. The model set is quite complex as the given structure is non linear and this implies that a few local minima can exist for the cost functions proposed in chapter 5. This is reflected in the dependence of the parameters on the training data set.

In table 6.6 the parameter vector is unrolled for each data-set.

Parameter		Data-set					
Symbol	Unit	1	2	3	4	5	6
$R$	$[\Omega]$	4.6027	1.9918	3.0404	2.9878	2.8547	2.9256
$L$	$[H]$	0.0274	0.0480	0.0694	0.0560	0.0447	0.0317
$K_m$	$[Nm/A]$	3.3791	1.1494	1.0546	1.6444	3.6105	3.1027
$J$	$[Nm s^2 rad^2]$	0.0002	0.0004	0.0004	0.0012	0.0045	0.0038
$B$	$[Nm s rad]$	0.0696	0.1771	0.2429	0.7630	0.2545	0.6189
$T_{dm}$	$[Nm]$	0.1180	0.5113	0.2284	1.9936	5.0169	4.2253
$\phi$	$[rad]$	-3.0850	-2.6863	-2.6449	-2.6558	0.2590	0.3715
$Q_{\{1,1\};\{2,2\}}$	$[A^2]$	0.3607	0.6405	0.1416	0.2851	0.8331	0.2301
$Q_{3,3}$	$[rad^2/s^2]$	1.0112	7.2882	28.3222	85.0108	55.9636	27.4993
$Q_{4,4}$	$[rad^2]$	0.0003	0.0003	0.0000	0.0000	0.0002	0.0000
$Q_{5,5}$	$[Nm^2]$	0.0108	0.0114	0.0156	0.0066	0.0076	0.0076
$R_{\{1,1\};\{2,2\}}$	$[A^2]$	0.0030	0.0019	0.0008	0.0019	0.0013	0.0003

Table 6.6: EKF optimal parameters for each data-set.

It is clear looking at these parameters that the model structure needs to be improved: the parameters vector should not depend on the training set (or at least its variations should be negligible).

# 7 Conclusions and future directions

*My interest is in the future  
because I am going to spend the  
rest of my life there.*

---

Charles F. Kettering

**In this chapter the main contributions of the work presented in this thesis are emphasized. Likewise, advantages and cost reductions over possible alternatives are summarized. Lastly, directions for future investigation are outlined.**

## 7.1 Contributions

Two solutions have been proposed to achieve fine positioning of stepper motors when driven via arbitrarily long cables.

A first solution, based on an open loop observer has been fully implemented and its performances have been summarized in terms of position repeatability and electromagnetic emissions.

A second, more complex control architecture, based on the combination of the open-loop observer mentioned above and an Extended Kalman Filter, has been proposed. The feed-back path has been fully designed, optimized

and implemented for this solution and its behavior has been extensively qualitatively and quantitatively described.

The main contributions of this thesis, which have been put forward to satisfy our design goals, but remain of general interest are:

1. An automatic procedure to simultaneously tune the model parameters and the EKF covariance matrices for a given EKF based control loop.
2. A versatile software architecture and some code optimization details for the implementation of an EKF based control loop.
3. Innovative requirements for the driver's hardware prototyping phase that ensure informative and consistent data-sets, to be used in the aforementioned automatic procedure, can be generated on-board.

## 7.2 Conclusions on the developed methods

The proposed control architectures present many advantages over the standard alternative options. Above, all the fully data-driven optimization procedure described in chapter 5 along with the self-adjusting cable-length dependant motor-current estimator allow the installation of the driver to be almost automatic and unmanned.

Furthermore, because of the data-driven nature of the optimization procedure, the proposed EKF algorithm could be potentially used with any hybrid stepping motor whose model falls in the proposed model set. This is a remarkable achievement and a notable advancement over commercially available sensorless drivers. Drivers are normally supplied with a given motor and only work with that specific motor.

In brief, quickly configuring the driver with both the motor parameters (with values normally available in the documentation supplied by the motor's manufacturer) and cable's characteristic figures (which can be easily measured for a cable sample in a laboratory, for example, with an RLC

meter) it is possible to acquire enough information to tune the EKF observer. The optimized observer can then be plugged into the DSP code at almost no additional installation cost. These easy steps can be performed with no information about the length of the cable connecting the motor phases to their driving electronics and potentially with any hybrid stepping motor.

### **7.3 Conclusions on the added value for the considered application**

The designed application for this driver is mission critical to the LHC operation: if the collimation system fails the beam is immediately dumped and injection is aborted. This implies recovery from failures must be handled as quickly as possible and lengthy configuration is not affordable.

In the LHC alone 139 collimators are installed. Each of the 4 stepping motors used to actuate the two carbon-composite bars is controlled independently and therefore 556 drivers must be installed. Installation cost plays a non-marginal role on the LHC Collimation Project budget and must be minimized.

The proposed self-adjusting cable-length dependent IIR filter used to estimate the motor side current greatly reduces installation costs and replacement time over an analogue alternative (which could, as a matter of fact, be implemented on a cheaper DSP). If the open-loop motor side current estimator had been realized in hardware with a second order lowpass filter the DSP would have been relieved from the computational burden of the estimation but a given driver would have operated only with one specific cable length.

Likewise, if the identification and EKF tuning had to be performed “by hand” each driver would have requested a few man-hours of an experienced engineer to tweak the various parameters involved in the Extended Kalman Filter algorithm yielding a likely worse estimation performance anyway.

## 7.4 Outlook

Future investigation should focus on improving the model structure. Hybrid stepping motors are complex physical systems and modeling them from first principles requires a considerable effort. In particular, the need for a computationally light model leads to various simplifications. Unlike other motor types such as DC motors or synchronous motors, a lack of a well established model for hybrid stepping motors is evident from literature surveys [47–49]. Some terms, such as the detent torque, are often neglected and even when included are loosely justified, furthermore, the proposed structure of their contribution varies from source to source. In fact, depending on the magnetic circuit approximation adopted, this torque turns out to be dependent either on the second or on the fourth electrical harmonic. A quick experiment carried out with our testbench suggested a dependance on the shaft angular velocity may help in discriminating between the two cases.

The next major effort should naturally be devoted to design and implementation of the state feedback controller.

# Bibliography

- [1] L. Evans and P. Bryant, “The CERN Large Hadron Collider: Accelerator and experiments,” *Journal of Instrumentation*, 2008.
- [2] R. W. Assmann, O. Aberle, G. Bellodi, A. Bertarelli, C. Bracco, H. Braun, M. Brugger, S. Calatroni, R. Chamizo, A. Dalocchio, B. Dehning, A. Ferrari, P. Gander, A. Grudiev, E. B. Holzer, J. B. Jeanneret, J. M. Jimnez, M. Jonker, Y. Kadi, K. Kershaw0, J. Lendaro, J. Lettry, R. Losito, M. Magistris, A. Masi, M. Mayer, E. Mtral, R. Perret, C. Rathjen, S. Redaelli, G. Robert-Dmolaize, S. Roesler, F. Ruggiero, M. Santana, P. Sievers, M. Sobczak, A. Tsoulou, V. Vlachoudis, T. Weiler, I. S. Baishev, and I. Kurochkin, “The final collimation system for the LHC.” in *10th European Particle Accelerator Conference*, Jul 2006, p. 4.
- [3] D. M. Fleetwood, P. S. Winokur, and P. E. Dodd, “An overview of radiation effects on electronics in the space telecommunications environment,” *Microelectronics Reliability*, vol. 40, no. 1, pp. 17 – 26, 2000. [Online]. Available: <http://www.sciencedirect.com/science/article/B6V47-3Y9GBSS-3/2/b5440f83c61e6d8ca7a6b1e2ba6e9891>
- [4] H. Chen, “R&D studies of the ATLAS LAr calorimeter readout electronics for super-LHC,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. In Press, Corrected Proof, pp. –, 2010.

- [Online]. Available: <http://www.sciencedirect.com/science/article/B6TJM-4YHT7JK-F/2/e610650d435b5e6213c02c15bb0c038d>
- [5] E. Beuville, P. Cenci, A. Federspiel, C. Gssling, E. H. M. Heijne, P. Jarron, H. W. Kraner, T. Massam, D. Munday, and T. Pal, "Measurements of degradation of silicon detectors and electronics in various radiation environments," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 288, no. 1, pp. 68 – 75, 1990. [Online]. Available: <http://www.sciencedirect.com/science/article/B6TJM-47STG4B-J/2/52f0c098f1d85ffdf9a933c4add71aa>
- [6] M. M. Angarano, W. Beaumont, M. Biasini, G. M. Bilei, M. T. Brunetti, B. Checcucci, C. Civinini, J. Coughlan, D. Creanza, M. de Palma, F. Drouhin, L. Fan, L. Fiore, M. French, A. Furtjes, A. Giassi, M. Giorgi, J. Gutleber, G. Hall, P. Lariccia, M. Loreti, G. Maggi, G. Mantovani, N. Marinelli, P. Mattig, G. Messina, S. My, A. Papi, V. Radicci, M. Raymond, R. Santinelli, G. Selvaggi, L. Servoli, L. Silvestris, P. Tempesta, A. Tsirou, P. G. Verdini, and B. Wittmer, "Study of radiation damage and substrate resistivity effects from beam test of silicon microstrip detectors using lhc readout electronics," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 488, no. 1-2, pp. 85 – 93, 2002. [Online]. Available: <http://www.sciencedirect.com/science/article/B6TJM-44XK7M3-1/2/433e6de0c9d3facbba245dc1103d6d11>
- [7] E. M. Rossi, S. Nicosia, and L. Zacarian, "A versatile PWM driver for electric machines: design and experiments," in *Proc. IEEE Int. Symp. Industrial Electronics ISIE 2002*, vol. 2, 2002, pp. 583–588.
- [8] D.-C. Lee, D.-H. Kim, and D.-W. Chung, "Control of PWM current source converter and inverter system for high performance induction

- motor drives,” in *Proc. IEEE IECON 22nd Int Industrial Electronics, Control, and Instrumentation Conf*, vol. 2, 1996, pp. 1100–1105.
- [9] K. G. Buss and A. Marshall, “PWM controllers for automotive applications,” in *Proc. Automotive Power Electronics*, 1989, pp. 114–118.
- [10] P. Acarnley, *Stepping Motors a guide to theory and practice*. The Institution of Electrical Engineers, 2002.
- [11] A. Kenjo, T. Sugawara, *Stepping Motors and their microprocessor controllers*. Oxford University Press, 2003.
- [12] U. S. D. Fitzgerald A. E., Kingsley C., *Electric machinery*. McGraw Hill, 2003.
- [13] T. Wildi, *Electrical Machines, Drives and Power Systems*. Prentice Hall, 2002.
- [14] G. Miano and A. Maffucci, *Transmission lines and lumped circuits*. Academic Press, 2001.
- [15] P.-G. Fontoliet, *Systèmes de télécommunications (volume XVIII du Traité d’Electricité)*. Editions Georgi, 1999.
- [16] A. Masi, G. Conte, R. Losito, and M. Martino, “DSP-based stepping motor drivers for the LHC collimators,” *IEEE Transactions on Nuclear Science*, vol. 55, no. 1, pp. 341–348, 2008.
- [17] G. A. Baker, *Essentials of Pade approximants*. Academic Press, 1975.
- [18] H. Foroush, “Angle estimation in PWM based stepping motor drivers for the LHC collimators,” Master’s thesis, EPFL, 2010.
- [19] K. J. Astrom and T. Hagglund, *PID Controllers: Theory, Design, and Tuning*. International Society for Measurement and Control, 1995.
- [20] R. L. Astrom K. J., “Integrator windup and how to avoid it,” *Proceedings of the American Control Conference*, pp. 1693–1698, 1989.

- [21] K. Ogata, *Modern Control Engineering*. Prentice Hall, 2010.
- [22] Modbus.org, “Modbus over serial line specification & implementation guide,” modbus.org, Tech. Rep., 2002.
- [23] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Transactions of the ASME—Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.
- [24] H. B. Kailath T., Sayed A. H., *Linear Estimation*. Prentice Hall, 2000.
- [25] M. Boutayeb, H. Rafaralahy, and M. Darouach, “Convergence analysis of the extended Kalman filter used as an observer for nonlinear deterministic discrete-time systems,” *IEEE Transactions on Automatic Control*, vol. 42, no. 4, pp. 581–586, 1997.
- [26] G. Evensen, “Using the Extended Kalman Filter with a multilayer quasi-geostrophic ocean model,” *J. Geophys. Res*, vol. 97, pp. 17 905–17 924, 1992.
- [27] G. W. Brown and M. T. Cliff, “Investor sentiment and the near-term stock market,” *Journal of Empirical Finance*, vol. 11, no. 1, pp. 1 – 27, 2004. [Online]. Available: <http://www.sciencedirect.com/science/article/B6VFG-49WPS3M-2/2/5c0912c4a4741662299f0451eb76f5b9>
- [28] S. Bolognani, L. Tubiana, and M. Zigliotto, “Extended Kalman filter tuning in sensorless PMSM drives,” *IEEE Transactions on Industry Applications*, vol. 39, no. 6, pp. 1741–1747, 2003.
- [29] L. R. Ray, A. Ramasubramanian, and J. Townsend, “Adaptive friction compensation using extended Kalman-Bucy filter friction estimation,” *Control Engineering Practice*, vol. 9, no. 2, pp. 169 – 179, 2001.
- [30] J. Persson and Y. Perriard, “An optimized Extended Kalman filter algorithm for Hybrid Stepper Motors,” in *Proc. 29th Annual Conf. of*

- the IEEE Industrial Electronics Society IECON '03*, vol. 1, 2003, pp. 297–300.
- [31] I. Landau and G. Zito, *Digital control systems: design, identification and implementation*. Springer Verlag, 2006.
- [32] M. Martino, A. Masi, R. Losito, and R. Picatoste, “Low emission self-tunable dsp based stepping motor driver for use with arbitrarily long cables,” in *12th IFAC-LSS Symposium*, 2010.
- [33] A. Tessarolo, “Getting the most from your C code on the TMS320C28x using CCS software,” in *TI developer conference*, 2005.
- [34] J. Persson, “Innovative standstill position detection combined with sensorless control of synchronous motors,” Ph.D. dissertation, École Polytechnique Fédérale de Lausanne, 2005.
- [35] S. Buyamin and J. Finch, “Comparative study on optimising the EKF for speed estimation of an induction motor using simulated annealing and genetic algorithm,” in *IEEE International Electric Machines & Drives Conference*, vol. 2, May 2007, pp. 1689–1694.
- [36] L. Bodizs, B. Srinivasan, and D. Bonvin, “Preferential Estimation via Tuning of the Kalman Filter,” in *DYCOPS 7*, 2004, p. 108. [Online]. Available: <http://www.dycops.org/>
- [37] K. L. Shi, T. F. Chan, Y. K. Wong, and S. L. Ho, “Speed estimation of an induction motor drive using an optimized extended Kalman filter,” *IEEE Transactions on Industrial Electronics*, vol. 49, no. 1, pp. 124–133, 2002.
- [38] P. Abbeel, A. Coates, M. Montemerlo, A. Ng, and S. Thrun, “Discriminative training of Kalman filters,” in *Proc. of Robotics: Science and Systems*. Citeseer, 2005.

- [39] S. Bolognani, L. Tubiana, and M. Zigliotto, “Extended Kalman filter tuning in sensorless PMSM drives,” *Industry Applications, IEEE Transactions on*, vol. 39, no. 6, pp. 1741 – 1747, nov. 2003.
- [40] L. Ljung, *System identification: theory for the user*. Prentice Hall PTR Upper Saddle River, NJ, 1999.
- [41] A. Blauch, M. Bodson, and J. Chiasson, “High-speed parameter estimation of stepper motors,” *IEEE Transactions on Control Systems Technology*, vol. 1, no. 4, pp. 270–279, 1993.
- [42] H. Durrant-Whyte, *Introduction to Estimation and the Kalman Filter*. Australian Centre for Field Robotics, 2001.
- [43] C. R. Paul, *Introduction to Electromagnetic Compatibility*. Wiley-Interscience, 2006.
- [44] G. Iadarola, “EMI Test-Bench for Stepper Motor Drivers,” CERN, Tech. Rep., 2008.
- [45] L. Idkhajine, E. Monmasson, and A. Maalouf, “Fully FPGA-based sensorless control for AC drive using an Extended Kalman filter,” in *Proc. 35th Annual Conf. of IEEE Industrial Electronics IECON '09*, 2009, pp. 2925–2930.
- [46] M. Bendjedja, Y. Ait-Amirat, B. Walther, and A. Berthon, “Sensorless control of hybrid stepper motor,” in *Proc. European Conf. Power Electronics and Applications*, 2007, pp. 1–10.
- [47] C. Kuert, M. Jufer, and Y. Perriard, “New method for dynamic modeling of hybrid stepping motors,” in *Proc. 37th IAS Annual Meeting Industry Applications Conf. Conf. Record of the*, vol. 1, 2002, pp. 6–12.
- [48] B. Kuo, G. Singh, and R. Yackel, “Modeling and simulation of a stepping motor,” *IEEE Transactions on Automatic Control*, vol. 14, no. 6, pp. 745–747, 1969.

- [49] K. Mizutami, S. Hayashi, and N. Matsui, “Modeling and control of hybrid stepping motors,” in *Proc. Conf Industry Applications Society Annual Meeting Record of the 1993 IEEE*, 1993, pp. 289–294.