VHDL-AMS MODELING AND SIMULATION OF A PMSM CONTROL SYSTEM FOR AUTOMOTIVE APPLICATIONS

Jorge Lopez Sanz. Matrikelnummer: 1822601
Master Thesis
Begin : 01/04/2012
End : 03/09/2012
Supervisor : Dr.-Ing. Faizal Arya Samman, Dipl.-Ing. François Philipp, Prof. Dr. Manfred Glesner
Herewith I declare, that I have made the presented thesis myself and solely with the aid of the means permitted by the examination regulations of the Darmstadt University of Technology. The literature used is indicated in the bibliography. I have indicated literally or correspondingly assumed contents as such.

Darmstadt, August 31, 2012

Jorge Lopez Sanz
Abstract

The goal of the present thesis is to present a novel alternative for modeling and simulating the controls of the PMSM of a battery electric car using VHDL AMS. With this aim a valid model integrating the mechanics and electronic concepts behind the PMSM and the method to control it were implemented using the mentioned hardware description language. The motor model with its equations related to the d-q reference was choosed in order to apply the effective motor control strategy called Field Oriented Control. Finally the validation with the simulator of the model and its controls was successfully carried out obtaining the desired response and concluding with useful observations not only about the system itself like the importance of the inverter switching frequency for a smooth response but also about the best method to apply for solving the problem optimizing time.
1 Introduction

1.1 History of electric motors .................................................. 11
1.2 The evolution of PMSM ....................................................... 14
1.3 State of art: FOC vs DTC ..................................................... 15
   1.3.1 Control rough idea ..................................................... 15
   1.3.2 Direct Torque Control:DTC ........................................... 16
   1.3.3 Field Oriented Control: FOC ......................................... 17
   1.3.4 Comparison FOC vs DTC .............................................. 18
   1.3.5 Summary: Model and Simulation, a clear goal ..................... 18

2 VHDL-AMS

2.1 VHDL ⇒ VHDL-AMS ......................................................... 21
2.2 New Modeling concepts introduced by VHDL-AMS ........................ 22
2.3 Overview of Languages for Hierarchical Design. Why VHDL-AMS? ... 24
2.4 Summary ................................................................. 25

3 PMSM theory

3.1 Classification of PMSM ....................................................... 27
3.2 Advantages of PMSM, comparisons ....................................... 28
3.3 PMSM equations: first approach to a model .............................. 30
3.4 PMSM equations: valid model for analysis ............................. 32
3.5 Summary ................................................................. 33

4 Control

4.1 Field Oriented Control ....................................................... 35
4.2 General complete scheme .................................................. 35
4.3 Control concepts block by block ........................................... 36
   4.3.1 Space vector modulation ............................................. 36
   4.3.2 Sequence ............................................................. 38
   4.3.3 Driver ............................................................... 39
   4.3.4 Inverter .............................................................. 40
   4.3.5 Substractor .......................................................... 41
   4.3.6 PID ................................................................. 41
   4.3.7 PMSM .............................................................. 42
4.4 Complete concrete Scheme ................................................ 44
4.5 Assumptions and Simplifications ......................................... 44
4.6 Implementation block by block .......................................... 45
   4.6.1 Space vector modulation ........................................... 45
   4.6.2 Sequence ............................................................ 47
   4.6.3 Driver ............................................................... 48
   4.6.4 Inverter ............................................................. 48
4.6.5 Subtractor ...................................................... 48
4.6.6 Motor ........................................................... 49
4.6.7 Measure ......................................................... 49
4.7 Summary .......................................................... 49

5 Simulations .......................................................... 51
  5.1 Preparation ....................................................... 51
      5.1.1 The BEV for which the engine is designed ................. 51
      5.1.2 The chosen PMSM ........................................... 52
      5.1.3 The tests specifications: Suburbanization .................. 55
  5.2 Results .......................................................... 56
      5.2.1 PID tuning. Optimization of experimental designs ........ 58
      5.2.2 Usual situation nr1: 0-100km/h ............................ 65
      5.2.3 Usual situation nr2: Parking ramp .......................... 77
      5.2.4 Usual situation nr3: New European Driving Cycle ........ 83
  5.3 Summary .......................................................... 85

6 Conclusions ......................................................... 89
  6.0.1 Conclusions ................................................... 89
  6.0.2 Recommendations for Further Work .......................... 90
## List of Tables

1.1 Classical DTC look-up table ................................................................. 17  
1.2 Comparison FOC vs DTC ................................................................. 18  
2.1 Comparison of Simulation Languages ................................................. 25  
3.1 Comparison BLDC-Brushed motors .................................................... 29  
3.2 Comparison BLDC-Induction motors .................................................. 29  
5.1 List of full-size BEV ............................................................................ 51  
5.2 Electrical parameters of the chosen PMSM ......................................... 53  
5.3 Mechanical parameters of the chosen PMSM ....................................... 53  
5.4 Translation to motor reference ............................................................ 54  
5.5 Calculation of Jm ................................................................................. 54  
5.6 Reference table for linear and angular speeds ...................................... 55  
5.7 List of 0-100 km/h acceleration ............................................................ 65  
5.8 Comparison of results depending on frequency .................................... 77  
5.9 Different test stages ............................................................................. 84
List of Figures

1.1 World Oil Energy Consumption by Sector, 1973-2007 .................................................. 12
1.2 Net Imports and Domestic Petroleum as Shares of US Demand, 2011. .......................... 13
1.3 Gross Imports by Major Sources Share of gross oil imports, 2011. ................................ 13
1.4 Magnetic fluxes in a PMSM ......................................................................................... 16
1.5 DTC Control Scheme for PMSM .................................................................................. 17
2.1 VHDLS-AMS code ....................................................................................................... 22
2.2 Example of discontinuity: operational amplifier .......................................................... 23
2.3 Example of different quantity types .............................................................................. 23
3.1 Classification of PMSM ................................................................................................. 28
3.2 Axis change .................................................................................................................. 30
3.3 Rotation of the magnet ................................................................................................. 31
3.4 Axis rotation ................................................................................................................ 32
4.1 General scheme implementation .................................................................................... 35
4.2 SVM Block .................................................................................................................... 36
4.3 Sectors in plane ............................................................................................................ 37
4.4 Sequence Block ........................................................................................................... 38
4.5 Switch sequence .......................................................................................................... 39
4.6 Driver Block ................................................................................................................ 39
4.7 Inverter Block ............................................................................................................. 40
4.8 Inverter circuit ............................................................................................................. 40
4.9 Subtractor Block ......................................................................................................... 41
4.10 PID Block .................................................................................................................. 41
4.11 PMSM Block ............................................................................................................. 42
4.12 Concrete scheme implementation .............................................................................. 44
4.13 Space vector modulation test bench .......................................................................... 46
4.14 Sequence test bench .................................................................................................. 47
4.15 Sequence test bench after clock falling edge .............................................................. 48
5.1 Forces opposed to the movement of the car ................................................................. 52
5.2 Transmission motor-wheels of the car ......................................................................... 54
5.3 Population growth and the growth of built-up areas (mid 1950s to late 1990s), selected European cities ......................................................................................................................... 55
5.4 Built-up area, road network and population increases, selected EEA countries. .......... 56
5.5 Procedure for PID tuning and simulations ................................................................. 57
5.6 Linear model with Simulink ....................................................................................... 58
5.7 Kp tuning mantaining Ki, Kd equal zero. Wm response ............................................. 59
5.8 Zoom of Kp tuning mantaining Ki, Kd equal zero. Wm response ................................. 59
5.9 Kp tuning mantaining Ki, Kd equal zero. Torque response ........................................ 60
5.10 Ki tuning mantaining Kp = 250 and Kd equal zero. Wm response ......................... 60
5.11 Zoom of Ki tuning mantaining Kp = 250 and Kd equal zero. Wm response .............. 61
5.12 Ki tuning mantaining Kp = 250 and Kd equal zero. Torque response ....................... 61
5.13 Zoom of Ki tuning mantaining Kp = 250 and Kd equal zero. Torque response ........... 62
5.15 Kd tuning maintaining Kp = 250 and Ki = 15. Speed response .......................... 62
5.16 Zoom of Kd tuning maintaining Kp = 250 and Ki = 15. Speed response .............. 63
5.17 Zoom of Kd tuning maintaining Kp = 250 and Ki = 15. Speed response .............. 63
5.18 Zoom of Kd tuning maintaining Kp = 250 and Ki = 15. Torque response .......... 64
5.19 Zoom of Kd tuning maintaining Kp = 250 and Ki = 15. Torque response .......... 64
5.20 Zoom of Kd tuning maintaining Kp = 250 and Ki = 15. Torque response .......... 65
5.21 Situation: acceleration 0-100km/h ......................................................... 66
5.22 Basic waveforms for 3kHz ...................................................................... 67
5.23 Zoom of id and tau electric for 3kHz ....................................................... 68
5.24 Vab, Vac and Vbc voltage waveforms ..................................................... 69
5.25 Stator currents in the transitory ............................................................... 69
5.26 Stator currents in the steady state ........................................................... 70
5.27 Supplied and demanded power ............................................................... 71
5.28 Supplied and demanded power in the steady state ................................... 72
5.29 Basic waveforms for 6kHz ...................................................................... 72
5.30 Steady state Iq and torque zoom for 6kHz ............................................... 73
5.31 Vab, Vac and Vbc voltage waveforms ..................................................... 73
5.32 Stator currents in the transitory ............................................................... 74
5.33 Stator currents in the steady state ........................................................... 74
5.34 Supplied and demanded power ............................................................... 75
5.35 Supplied and demanded power in the steady state ................................... 76
5.36 Supplied power zoom ........................................................................... 76
5.37 Situation: parking ramp ........................................................................ 78
5.38 Basic waveforms case parking ramp ..................................................... 79
5.39 Zoom speed response case parking ramp ............................................ 80
5.40 Supply stator voltages Vab, Vac and Vbc ............................................... 80
5.41 Initial transitory stator currents .............................................................. 81
5.42 Stator currents at speed 18km/h ............................................................. 81
5.43 Stator currents at speed 5km/h ............................................................... 82
5.44 Supplied and Demanded Power parking ramp ...................................... 82
5.45 Simulation of the driving cycle innercity .............................................. 85
5.46 PID tuning proceed .............................................................................. 86

8

List of Figures
1 Introduction

"Let the future tell the truth and evaluate each one according to his work and accomplishments. The present is theirs; the future, for which I really worked, is mine."

— Nicolas Tesla

In this chapter the reasons for the research that represents this thesis will be discussed. Permanent magnet synchronous motors and drives will be placed in history and from this point of view, the need of such a motor and an easy and effective drive to control it, will be clear.

The historical participation of electrical engines in transportation development will be underlined based on [1] and after explaining the details like the dc-ac battle, the appearance of the combustion engines... the role of electric motors (and above them PMSM) in the near future will be obvious and this way the quote with which the chapter was opened agrees to the prophecy of this legendary scientist father of AC electric engines.

Having understood why electric motors are very interesting for humanity the next sections will deal with a brief description of the state of art: first of the PMSM (the problem) and second the control (the solution). In the summary the purpose of this thesis, which has been drawn step by step will be clearly reflected.

Knowing the state of art of PMSM and its control, the last part of this chapter will deal with the empty place found in technical bibliography for exact modeling and valid simulation of the system and controls of PMSM: a crucial step before implementing the controls in hardware. This crucial step which relates in detail the electronic world with the mechanical sector of automotive is the reason for this thesis, a necessary background to face the huge project of implementing with FPGA the controls of a PMSM.

At the end of this section a brief concept of the changes and tasks to realize to enter the next step (implementation) will be introduced.

1.1 History of electric motors

The first human race’s experience with electric motor was the primitive model of a dc motor invented by the English physicist and chemist Michael Faraday in 1821. Some decades later, 1870, the Belgian electrical engineer Zénobe-Théophile Gramme developed the first commercially viable dc motor.

In 1900 this dc motor was commonly used in street railways, mining and industrial applications and scientists and industry professionals began to point his curiosity to the different architectures of dc motors and their controls in order to meet specifications for everyday applications.

But the boom of electric dc engines was slowed by the mass production of combustion engines in the early 1900s and the development of new technology to profit the fossil fuels and their apparent abundance delayed the use of the electric motor in automotive applications and in many other scenes. Suddenly the world started to rant about the old toy and its disadvantages appeared to be suddenly
increasing: excessive wear in the electromechanical commutator, low efficiency, and fire danger due to sparking, limited speed...

The dc motor representing the electrical engine was losing the battle. However, Faraday’s discovery of the concept of electromagnetic induction in 1831 opened the path for the invention of induction motors.

It was 1883 when the Serbian-American engineer Nikolai Tesla invented the first alternating-current induction motor. This is generally considered to be the prototype of the modern electric motor. Proof of it is the famous electrical cars company called "Tesla Motors" which manufactures and sells electric cars and electric vehicle powertrain components.

This induction motor was the first brushless motor. The synchronous motor, which is also a brushless motor, was invented by Tesla as well. By the year 1900, exactly when the combustion engines were gaining ground, the induction motor and synchronous motor were well known but because the world was still dc they could not be commercially available.

Some years later AC power would defeat the monopoly of DC power due to its flexibility, easy production, distribution and utilization.

Even though the AC power available, the automotive industry continued working with the combustion engines because of the apparent advantages of fossil fuels as energy source.

It wasn’t till the past decades that the disadvantages of the apparent perfect combustion engine started to be questioned. There are different reasons that make the combustion hegemony impossible and in this scenario electric cars will play a vital importance role.

Global warming is one of the first arguments to take into account. The greenhouse gas more abundant and that produces more enhanced greenhouse effect is carbon dioxide (CO2) and this is a product of HC combustion in conventional motors.

Since the Kyoto protocol (adopted in 1997 but entered into force 2005) the alarm of a climate change made the countries more concerned about the pollution. As seen in fig. 1.1 (Source: International Energy Agency) transport sector represents more than the half of petroleum consumption.

![Figure 1.1: World Oil Energy Consumption by Sector, 1973-2007](image-url)
Moreover the fuel fossils represent another great inconvenient: they are limited. Even if the debate is around what is understood under resources the fact that the petroleum is limited is commonly accepted. Most pessimistic statistics even give to it a live expectancy of 43 years.

Furthermore, the geographic distribution of oil consumers and oil reserves is completely complex. That makes fuel consumers dependent on other countries. The first world which consumes the major part of the fuels has no reserves while other countries in development where is not very common that every family owns a car have rich reserves of oil. For this reason international conflicts and wars are promoted by the pressing demand of consumers of the first World. Taking a short look at the two following graphs the last wars in the East can be seen as a result of the oil curse to the poor people in these countries.

These ideas get clearer with pictures 1.2 (Source: U.S. Energy Information Administration (March 2012)) and 1.3 (Source: Congressional research service. U.S. Oil Imports and Exports ("Persian Gulf OPEC" members are Saudi Arabia, Kuwait, United Arab Emirates, Iraq, and Qatar. Other OPEC members are Angola, Nigeria, Algeria, Libya, Venezuela, and Ecuador)). In the first one it can be seen the dependency of U.S.A. (the biggest oil consumer) on foreign countries and in the second one the countries with most resources are shown.

![Figure 1.2: Net Imports and Domestic Petroleum as Shares of US Demand, 2011.](image1)

![Figure 1.3: Gross Imports by Major Sources Share of gross oil imports, 2011.](image2)

All in all, the actual system of energy supply and consumption is no longer sustainable: environment, international conflicts, limited resources... and in transportation scenario it is time to summon an ancestor, to come back to forgotten times of Faraday or Tesla: the electric motor is back.

1.1 History of electric motors
In the last years much effort and economic resources have been put in HEV (hybrid electric vehicle) development, which combines a conventional combustion motor with one or more electric motors. World famous Toyota Prius, Ford Escape Hybrid, Honda Civic Hybrid...and a huge list evidence that the world is turning towards electric motors.

The next step will be BEV (battery electric vehicles), fully powered by an electric engine and the aim of this thesis will be useful. Governments all over the world are economically supporting industry and consumers that decide to buy a BEV ("National Electric Mobility Platform" in Germany is a good example), the companies that invest in such projects have several legal advantages too and in universities all over the world thousands of scientist and engineers are working together in one direction: the change to electric cars. Challenges like the supply and organization to cover new electrical net demand, the research of new elements for batteries determining the autonomy of the car or (the topic of this thesis) the selection of the right motor and its controls...evidence that BEV is coming, it is the next reality in the near future and it must be well known to be properly controlled and to satisfy the expectations of the applications, this is the aim of the present thesis, to get more familiar with the coming motor.

1.2 The evolution of PMSM

As said before, the induction motors, since their birth, awoke the curiosity of the brilliant minds like Tesla’s. There were some advantages that made them look in great measure attractive. The main reasons were their lack of commutators and that their speed is only limited by the physical constraints of the motor.

Induction and synchronous motors utilize the same type of stator. But synchronous motors use a wound dc field or permanent magnet rotor instead of the wire-wound or squirrel cage rotor of induction motors. Induction motors can generate torque in a wide range of speed, while synchronous motors in the can only generate torque at the synchronous speed. And this synchronous speed could not be varied easily in the beginning because it depends on the source frequency and it did not exist PWM or such methods to change the frequency. For these reasons the first synchronous motors had to run at speeds of 3600, 1800, 900... RPM for a line frequency of 60 Hz in USA (50Hz in Europe). It was also very uncomfortable the fact that the speed of synchronous motors had to be increased first to synchronous speed by means of an auxiliary motor before the motor could be used.

Introduction of the line-start PMSM in the 1950s provided a solution to this problem. The rotor of line-start PMSM is made of permanent magnet embedded inside a squirrel-cage winding. Many induction motors use squirrel-cage rotors. Induction of current in the squirrel cage produces torque at zero or higher speeds the same way torque is generated in induction motors. Therefore, the line-start PMSM can develop torque at zero speed, and run as an induction motor, until the synchronous speed is reached.

After having reached the rated speed the rotor is synchronized with the power source and no more current is induced in the squirrel cage. The main disadvantage of the line start PMSM was the high cost.

Shortly after the motor drives started to be used to convert dc power into ac power with any desired frequency and this way the delivery of power to the motor could be perfectly controlled. With this development the PMSM could be used efficiently at any speed and therefore the line start PMSM is almost obsolete.

The next problem that PMSM had to face was that because of the strong nonlinear system of the controls of PMSM the simulation was complex and took a lot of time. Park’s transformation, 1993, paved
the way towards linear and instantaneous control over torque for PMSM by transforming the variables to control the motor in the stationary and the rotor reference frames which yields the two-axis equivalent circuit for a PMSM. Treating these variables from the rotor’s references they became constant in steady state making the analysis easier.

With this transformation available the PMSM control world took the direction of the so called "vector control" field with the great hopes put on linear control over torque and the demand for high performance variable speed motor drives using this techniques, this will be discussed in the next section.

### 1.3 State of art: FOC vs DTC

As just said, the trend nowadays to solve the problem of controlling a PMSM points in vector control strategies. Two methods among this family are mostly used for controlling the PMSM: DTC (Direct Torque Control) and FOC (Field Oriented Control).

FOC was the first vector control method that solved the complexity of steady state and transient simulation. In the middle of the 80’s DTC arrived in parallel with two different names: Direct Torque Control (DTC) and Direct Self Control (DSC).

Both methods were first implemented in the control of IM drives and more recently they have been applied to PMSM, a motor in upwards.

In the next subsections the two methods will be roughly explained and in the last one a comparison between them will be carried in order to understand the decision of using FOC. The major part of this section information and the pictures are from [2] and the Wikipedia.

#### 1.3.1 Control rough idea

Later it will be explained in detail (and with other names of variables) the architecture of a PMSM and how it works, but some brief concepts are necessary at this point to understand the basic idea of the control solutions here proposed.

The PMSM is fed by three voltages (AC) created by a power inverter. These three voltages enter the inductance of the stator windings (three coils physically placed every 120 degrees) create a magnetic field of arbitrary magnitude and orientation that interacts (because of repulsion and attraction) with the permanent magnet’s field of the rotor. Controlling the orientation and magnitude of this field, the forces that move the rotor can be controlled. And to control the magnetic field is necessary to control the input currents in the stator windings.

In 1.4 can be seen the fluxes of the PMSM. $\Psi$ is the flux of the permanent magnet (rotor) and $\Psi_s$ is the stator’s flux which is decomposed in d and q axis.
Using this notation there are two basic equations of the PMSM to take into account in order to control it:

\[ \tau_e = \frac{3}{2} p \frac{\Psi \cdot \Psi_s}{L_s} \sin \delta \]  \hspace{1cm} (1.1)

\[ u_s = R_s \cdot i_s + \frac{d\Psi_s}{dt} \]  \hspace{1cm} (1.2)

1.3.2 Direct Torque Control:DTC

In DTC two variables are measured: stator voltages and motor current vector in order to estimate two variables: torque and stator flux linkage.

Integrating the stator voltages, the stator flux linkage \( \Psi_s \) is estimated using 1.2 and neglecting the voltage drop in the stator resistance.

Torque is estimated with 1.1 as a cross product of estimated stator flux linkage vector and measured motor current vector \( (=\Psi_s \frac{3}{2} \frac{p}{L_s} \)).

With the stator flux loop the torque is controlled, while the measured electric torque feedbacks the loop to control the speed.

Once the flux magnitude and torque are estimated, they are compared with their reference values and in case either the estimated flux or torque deviates from the reference more than allowed tolerance, the transistors of the variable frequency drive are turned off and on in such a way that the flux and torque will return in their tolerance bands as soon as possible.

This is realized by the hysteresis blocks (1.5) which are used to place the actual situation of the error: in 1.1 the hysteresis outputs can be read. The hysteresis block for the flux has two possible values: 1 or -1 while the output of the torque hysteresis block can be 0, 1 or -1.
Using this information and the sector (the idea of sectors will be described in detail in FOC control section, middle of the thesis), the action to be followed by the switches in the inverter can be found in the table 1.1.

![DTC Control Scheme for PMSM](image)

Figure 1.5: DTC Control Scheme for PMSM

<table>
<thead>
<tr>
<th>Table 1.1: Classical DTC look-up table</th>
</tr>
</thead>
<tbody>
<tr>
<td>K(Ψs)</td>
</tr>
<tr>
<td>dΨ = 1, dΨ = 1</td>
</tr>
<tr>
<td>dΨ = 1, dΨ = 0</td>
</tr>
<tr>
<td>dΨ = 1, dΨ = -1</td>
</tr>
<tr>
<td>dΨ = -1, dΨ = 1</td>
</tr>
<tr>
<td>dΨ = -1, dΨ = 0</td>
</tr>
<tr>
<td>dΨ = -1, dΨ = -1</td>
</tr>
</tbody>
</table>

### 1.3.3 Field Oriented Control: FOC

The Field Oriented Control is the solution chosen in this thesis. For this reason the exact concept behind this method will be discussed exhaustively during all the present work, in this part only a summarized idea will be described for understanding the comparisons between the two methods.

FOC achieves a decoupled control of the torque and the flux. Characteristic from this method is the use of the d-q transformation that separates the stator current in two components: one orthogonal to the magnetic field of the rotor (iq) and the other one parallel (id). Very roughly could be said that the idea of FOC is based on maximizing the orthogonal component which is responsible for the torque and minimizing the parallel component which only "serves to compress the motor bearings", see [3].

The control system, as will be seen later, is divided into three different loops: a loop for controlling the direct component (id) and force it to become zero, and two loops for the q current which controls speed and torque. The fact that the torque can be controlled only with the orthogonal component comes from a simplification due to how the motor is built physically (as will be explained; only SPMSM have this special feature).

From the equations of the PMSM (shown later) can be seen that id and iq are not independent from each other and for this reason the control cannot separate its controls completely. To achieve completely independent regulation it would be necessary to cancel the effect of the coupling terms in the equations. Even knowing that the use of decoupling achieves the linearization of the control system as well as higher
dynamics, the decoupling strategy is out from the goal of this thesis.

1.3.4 Comparison FOC vs DTC

Paper [2] concludes its deep comparison of the two methods summarizing that both methods provide high performance response with quicker torque dynamics in DTC and better steady state behavior in FOC. Having a look at 1.2 taken from [2], it is evident that both methods are valids in any case. The authors of this paper also emphasize in the fact that "depending on the requirements of a particular application one method can be more convenient than the other". To the same results arrive the authors of [4].

In this case and having in mind that in the control of a BEV is of high interest the role that steady state plays, the Field Oriented Control method has been chosen with some limitations that will be discussed in the further works section.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>DTC</th>
<th>FOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic response for torque</td>
<td>Quicker</td>
<td>Slower</td>
</tr>
<tr>
<td>Steadystate behaviour for torque, stator flux and currents</td>
<td>High ripple and distortion</td>
<td>Low ripple and distortion</td>
</tr>
<tr>
<td>Requirement of rotor position</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Switching frequency</td>
<td>Variable, depending on the operating point and during transients</td>
<td>Constant</td>
</tr>
<tr>
<td>Current control</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Audible noise</td>
<td>Spread spectrum, high noise especially at low speed</td>
<td>Low noise at a fixed frequency</td>
</tr>
<tr>
<td>Complexity and processing Requirements</td>
<td>Lower</td>
<td>Higher</td>
</tr>
</tbody>
</table>

1.3.5 Summary: Model and Simulation, a clear goal

Till now it has been argued why electric motors are a new and attractive challenge, why PMSM is a good option as the problem to be solved in this challenge and why FOC provides a good solution to solve the problem. All these aspects can be found in hundreds of papers, thesis, journal publications and other scientific documents.

What is the contribution of this thesis in PMSM control applied to electric vehicles regarding all the existing literature? First of all it should be remarked that most papers talking about the PMSM and its controls concentrate too much in the electronic field. After very precise details of the way to command the power switches of the driver using complex techniques such as space vector modulation, the results appear suddenly and briefly, usually without deep reasons for the values of the mechanical parameters used. This is the case of [5], [6]. It is not uncommon to find elaborated thesis about the FOC for a PMSM in automotive section that concludes with the results collected in a little PMSM of a few kW and submitted to torques which values are not comparable to a power application of the BEV dimensions.

On the other hand there are papers like [7], [8], [9] that give an accurate description of the mechanical parameters: how to calculate the inertia of the motor, what is physically the load torque, how to translate the angular velocity in rpm to linear speed (transmission)...but the electronic concepts remain a little in the air: there is no detailed description of the FOC method or the PID tuning.
Having observed this lack in the documentations, this thesis aims to cover this deficiency representing a basic step for those who pretend the further implementation of the PMSM FOC controller, offering them a solid and detailed method to model and simulate the complex control of a PMSM for a BEV neglecting neither the tricky electronic blocks nor the hard mechanical concepts related to the huge and interesting problem to solve: the PMSM.
2 VHDL-AMS

"The successful construction of all machines depends on the perfection of the tools employed, and whoever is a master of the art of toolmaking possesses the key to the construction of all machines"

— Charles Babbage

"The real world is analog": This is a common sentence repeated in classrooms all over the world where basic electronic ideas are broadcasted. Electronic solutions could be mostly understood as digital processed solutions (FPGA, ASIC...) and here the need of a powerful tool able to handle with this duality: analog-digital, real system-computer, problem-solution is becoming increasingly obvious.

In this chapter first a short review on history will reveal the origins and birth of the new tool called VHDL-AMS used in this Thesis to simulate the control of the PMSM. Afterwards the main new modeling concepts related to the introduction of the AMS extension will be described and at last a brief comparison with other simulation tools will be discussed.

It is time to immerse in the late 80. The evolution and advantages of VHDL-AMS can be found in the preface of [10].

2.1 VHDL $\Rightarrow$ VHDL-AMS

VHDL arouse out of the United States government’s Very High Speed Integrated Circuits (VHSIC) program. It was developed under the auspices of the Institute of Electrical and Electronic Engineers (IEEE) because of the need for a standard language for describing the structure and function of integrated circuits (ICs). The first standardization of the language appeared in 1987 (IEEE 1076) and was extended six years later (IEEE 1076-1993).

Because of the increasing need to describe and simulated analog- mixed circuit and systems an IEEE Working Group started to develop a set of design objectives for an extension to VHDL and completed a draft Language Reference Manual in 1997. The draft was refined and subsequently approved in 1999, becoming IEEE Standard 1076.1. The standard extension 1076.1 together with IEEE 1076 is informally known as VHDL-AMS (IEEE 1076.1-1999).

VHDL-AMS incorporates a lot of advantages in the design process. First of all it allows to divide the description of the system into subsystems and then it is only necessary to interconnect the subsystems. These subsystems can use equations of quantities of different natures, because one of VHDL-AMS strengths is the possibility of multi-disciplinary modeling and design of electro-mechanical, mechatronic and micro electromechanical systems (MEMS).

It also allows the specification of the function of a system using familiar programming language and equation forms.
Simulating the designed system allows the designer to quickly compare alternatives and test for correctness without the delay and expense of hardware prototyping. At last it should be mentioned that it allows the detailed Fourth, it allows the structure of a design to be synthesized from a more abstract specification, allowing designers to concentrate on more strategic design decisions and reducing time to market.

How can all this new advantages be translated to the aim of this thesis? Using VHDL-AMS the controller of the PMSM can be simulated and proofed before being manufactured and tried in the real motor. Modeling the motor to control and simulating its response to the digital commands makes a time- and money saving possible.

2.2 New Modeling concepts introduced by VHDL-AMS

The first difference between VHDL and VHDL-AMS is due to the continuous character of the analog values. VHDL presented an "event driven behavior" and that means that every step of simulation depended only on the signal changes. An example would be a clock, very useful that always appears in VHDL codes: a sensor sends a signal in form of digital word and when do the values of the simulation change? When the clock presents an edge or whenever another signal changes its value and appears in a sensitive list of a process.

In VHDL-AMS the world becomes bigger and type-richer. Just imagine the value of the current flowing through the stator windings of a motor, this is an analog value and varies with the time. It does not work with events, in every time division the value is unique and possibly different from the previous and different from the next, it changes constantly. For this reason it has to work regarding on a continuous behavior, too.

Just to see some little difference in the language see the figure taken from [11](All the figures of this subsection have been taken from there).

The color blue represents the typical VHDL lines and the yellow-oranges the new definitions for cotinuous quantities of VHDL-AMS.

Continuous models are based on differential algebraic equations (DAEs) and they are solved by a dedicated simulation kernel: the analog solver. With VHDL-AMS initial conditions, piecewise-defined behavior and discontinuities can be set, but optimization of the set of DAEs being solved and how the
analog solver computes its solution are outside the scope of VHDL-AMS.

Here, in fig 2.2, is an example of discontinuity. In this case discontinuities are used to model the behavior of an operational amplifier. When the input voltage arrives to one of the limits, the output must saturate thus implying a sudden change. This is solved by using the ‘above attribute and a break statement that will point to the the solver the sudden change of the quantity.

![Figure 2.2: Example of discontinuity: operational amplifier](image)

Let’s focus on more concrete aspects of modeling. The analog subsystems created with VHDL-AMS are based on a new kind of data different from variables or signals, they are called quantities and are the unknowns in the equations to solve.

There are three types of quantities depending on their definition: free quantity (they are not associated to any physical node or function), branch quantity (they are described as amount of some physical quantity: heat, light, current... across or through two nodes) and implicit quantities (defined by other quantities or signals like derivative, integral, ramp: ‘dot, ‘integ, ‘slew,’ramp...)

In figure 2.3 can be seen an example with different quantity types. Interface_q is a port quantity and will be connected with another entity, v1 and v2 are branch quantities because they are defined between the nodes positive and negative and free_q is a free quantity.

![Figure 2.3: Example of different quantity types](image)

VHDL-AMS permits modeling a circuit just defining first every component separately and connecting them with the nodes afterwards. With this information the analog solver should be able to use energy...
conservation laws (such as Kirchoff in electric nature) to solve the system.

Important is to remark that working with analog values in every entity the number of equations must be equal to the number of trough quantities + free quantities + interface quantities with mode out.

2.3 Overview of Languages for Hierarchical Design. Why VHDL-AMS?

Before explaining the reasons for choosing VHDL-AMS to simulate the controls of the PMSM, a short review and comparison of available languages was made, here are the main ideas of this review. In the following paragraphs, five different languages will be described to finish the section with a table underlining the strengths and weaknesses of each. The overview starts with the chosen one:

1) VHDL-AMS: Just to remark the most important points, because it was already described in the previous section, VHDL-AMS permits that the equations describing the conservative aspects of a system do not need to be explicitly annotated by the user, that means the VHDL-AMS solver automatically verifies the conservation of energy.

For working with VHDL-AMS there are softwares available such as Questa, that incorporate a good simulator that makes the study of the model behavior more comfortable.

2) Performance Language-SLAM: SLAM is a high-level performance modeling language. It offers the opportunity to describe the overall system as a stochastic system. It also combines alternate modeling methodologies within a single simulation model. It allows printing several statistical reports for final data analysis, and it also provides a useful simulation methodology for performance evaluation. However, it lacks the capacity to model and simulate hierarchical multiple-level behavior. Its modeling capability is limited to abstract high-level models, and it does not support component-level coupled-energy descriptions.

3) C/C++: This languages present some advantages. They are popular, powerful and flexible languages, and therefore a wide variety of C/C++ compilers and helpful accessories are available. They provide powerful dynamic data structures. Another positive aspect is their flexibility by semantics and description of mathematic functions to build a wide variety of system models.

The negative aspects of C/C++ are crucial for not choosing this language in this thesis: there is no natural way in C/C++ to represent constrained data types, concurrency, and clocks. In addition, the C/C++ language does not provide an associated simulator, the designer is required to build the model solver.

4) Matlab: MATLAB is a powerful high-level language that is especially suitable for demonstrating mathematical concepts. Matlab offers a useful working environment for quick model calculation and full simulation tasks. It is used all over the world and in thousands of different disciplines, the common data flow on the net is huge. It could be said as an exaggeration that the one who works with Matlab never works alone.

5) SystemC: SystemC is a new open source library in C++. SystemC and standard C++ development tools can be used to create a system model from the system level to the component level, quickly simulate to validate and optimize the design, explore various algorithms, and provide the hardware and software development team with an executable specification of the system.

Another advantage is that SystemC supports a rich set of port and data types. The multiple-level abstract design methodology is one of the most important properties of SystemC, ranging from the higher system level to lower component level. Moreover, to model and simulate continuous perspective with...
SystemC, differential equations with respect to time can be discretized and transformed into corresponding difference equations.

In summary, in table 2.1 from [12] the main features of each of these languages can be seen. The first decision is to use a language that provides a simulator, because this is the main target of the thesis.

By this selection the two candidates left are VHDL-AMS and Matlab. VHDL-AMS offers a great advantage in describing the Hardware to implement later. As will be remarked in the theory part, in this thesis the implementation won’t be realized but prepared. For this reason Matlab could have been a very easier tool for simulating the system of PMSM and its control with a lot of work done by the huge Matlab community.

But important is to focus on the advantage of VHDL-AMS because even if it is a little harder to model and simulate (the new aspects pointed in previous section should be learned), the effort is not in vain, because the next step (and this will be very concretely treated in further works) would be to implement and synthetize the control of the PMSM developed in this thesis making only some little changes in the design.

Therefore all the effort put in describing the model in VHDL-AMS is worth, there is no need to use some automatic VHDL translator like Matlab codes need that can vary some details and make the code less efficient that the one developed directly by the VHDL-AMS user.

Nevertheless as an advance it should be said that Matlab will be used as an auxiliar powerful calculating software when tuning the PID. For this reason the combination of Simulink and VHDL AMS will make the solution of this thesis really easier.

<table>
<thead>
<tr>
<th>Table 2.1: Comparison of Simulation Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
</tr>
<tr>
<td>Associated Simulator</td>
</tr>
<tr>
<td>Component-level Description Capability</td>
</tr>
<tr>
<td>Behavior Modeling Perspective</td>
</tr>
<tr>
<td>Data Structure</td>
</tr>
<tr>
<td>User-Definable Behavior</td>
</tr>
<tr>
<td>Concurrency/Timing Mechanism</td>
</tr>
<tr>
<td>Multi-level Description Capacity</td>
</tr>
<tr>
<td>Analysis Capability</td>
</tr>
</tbody>
</table>

2.4 Summary

In this chapter the election of the basic tool for modeling and simulating the problem and its solution has been argued. The main ideas behind VHDL-AMS has been studied. In the last section a comparison has been done highlighting the strengths of VHDL-AMS for describing a multidisciplinary problem in a comfortable environment of simulation. The need of a tool with these positives features will become gradually clear in the next chapters from the description of the model in the next chapter till the final simulations.
3 PMSM theory

The following section deals with the description of the motor’s model. First a short overview of the classification of permanent magnets motors is presented. Having classified the chosen motor, a brief comparison with other motors for BEV is made in order to understand the reasons for the selection. Afterwards, using the theory the mechanics behind the engine are described in detail concluding with the mathematical model to implement in VHDL AMS, what is the goal of this section.

3.1 Classification of PMSM

In figure 3.1 a classification of permanent magnet motors is made to place the brushless motor used in this thesis.

The first division separates the motors depending on the nature of their input: DC brushless motors are controlled applying directly DC voltage (PMDC) while AC brushless motors need to be fed with AC voltages (PMAC). In any case, it is important to remark, that AC brushless motors need an AC wave, but most of the times (like in this thesis) the input wave is created using a DC source and the right controls.

The next classification makes a gap in PMAC splitting it according to the form of the back electromotive force (BEF) generated in the windings of the stator when the rotor rotates. BLDCM (commonly brushless dc motors) generate a trapezoidal BEF in their windings, while PMSM (permanent magnet synchronous motors) generate a sinusoidal one. The control strategy chosen in this thesis requires sinusoidal BEF, because this makes the torque output smoother than that of a trapezoidal motor. The price to pay for this advantage is the increase of copper (and thereby of cost) in the stator windings.

The last division corresponds to the physical situation of the magnets in the rotor. If the magnets are placed on the surface of the rotor then the motor is called SPMSM (surface permanent magnets synchronous motor). If they are placed inside the rotor the name is IPMSM (Interior permanent magnets s. m.). In this thesis the motor modeled will be a SPMSM because these motors present some characteristics that make them easier to control (parameters Ld and Lq equals). This is a very common architecture used in Field Oriented Control as will be seen in 3.3
3.2 Advantages of PMSM, comparisons

To decide which motor is needed it is important to focus on the car it will fit in. The goal of this thesis is to implement the control of a PMSM for an electrical car similar to the Mitsubishi i-MiEV, [13], that means a hatchback car for using in the city and its surroundings prioritizing the efficiency over high speeds or acceleration.

Nowadays the most used electric motors in automotive are AC motors, PMSM (or wrongly abused term: dc brushless motors) and induction motors. Brushed dc motors require more maintenance and therefore have a shorter life compared to brushless motors. The reason for that is among others the friction of the brushes. For a regular car such as this thesis target, durability is an important factor to persuade the costumers of the commodity of acquiring one vehicle with a PMSM.

Another advantage over brushed motors and even induction motors is that brushless motors produce more output power per frame size than the other direct competitors. This derives in lighter motors and is largely known that the weight in automotive industry is a critical point to consider.

Moreover the rotor of a brushless motor is made of permanent magnets. Therefore the rotor inertia is less compared with other types of motors and acceleration and deceleration characteristics are improved.

Another positive aspect of BLDC motors is the fact that they operate much more quietly than brushed and that reduces the electromagnetic interferences (EMI’s) . In tables 3.1 and 3.2 can be seen the advantages described by Padmaraja Yedamale in his [14]. Remember the only difference between PMSM and BLDC are the windings (sinusoidal and trapezoidal respectively) and its controls.
### Table 3.1: Comparison BLDC-Brushed motors

<table>
<thead>
<tr>
<th>Feature</th>
<th>BLDC Motor</th>
<th>Brushed DC Motor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commutation</td>
<td>Electronic commutation based on Hall position sensors</td>
<td>Brushed commutation</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Less required due to absence of brushes</td>
<td>Periodic maintenance is required</td>
</tr>
<tr>
<td>Life</td>
<td>Longer</td>
<td>Shorter</td>
</tr>
<tr>
<td>Speed/Torque Characteristics</td>
<td>Flat: Enables operation at all speeds with rated load.</td>
<td>Moderately flat: At higher speeds, brush friction increases, thus reducing useful torque.</td>
</tr>
<tr>
<td>Efficiency</td>
<td>High: No voltage drop across brushes.</td>
<td>Moderate.</td>
</tr>
<tr>
<td>Output Power/ Frame Size</td>
<td>High: Reduced size due to superior thermal characteristics. Because BLDC has the windings on the stator, which is connected to the case, the heat dissipation is better.</td>
<td>Moderate/Low: The heat produced by the armature is dissipated in the air gap, thus increasing the temperature in the air gap and limiting specs on the output power/frame size.</td>
</tr>
<tr>
<td>Rotor Inertia</td>
<td>Low, because it has permanent magnets on the rotor. This improves the dynamic response.</td>
<td>Higher rotor inertia which limits the dynamic characteristics.</td>
</tr>
<tr>
<td>Speed Range</td>
<td>Higher: No mechanical limitation imposed by brushes/commutator.</td>
<td>Lower: Mechanical limitations by the brushes.</td>
</tr>
<tr>
<td>Electric Noise Generation</td>
<td>Low.</td>
<td>Arcs in the brushes will generate noise causing EMI in the equipment nearby.</td>
</tr>
<tr>
<td>Cost of Building</td>
<td>Higher: Since it has permanent magnets, building costs are higher.</td>
<td>Low.</td>
</tr>
<tr>
<td>Control</td>
<td>Complex and expensive.</td>
<td>Simple and inexpensive.</td>
</tr>
<tr>
<td>Control Requirements</td>
<td>A controller is always required to keep the motor running. The same controller can be used for variable speed control.</td>
<td>No controller is required for fixed speed; a controller is required only if variable speed is desired.</td>
</tr>
</tbody>
</table>

### Table 3.2: Comparison BLDC-Induction motors

<table>
<thead>
<tr>
<th>Feature</th>
<th>BLDC Motor</th>
<th>AC Induction Motors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed/Torque Characteristics</td>
<td>Flat: Enables operation at all speeds with rated load.</td>
<td>Nonlinear: Lower torque at lower speeds.</td>
</tr>
<tr>
<td>Output Power/ Frame Size</td>
<td>High: Since it has permanent magnets on the rotor, smaller size can be achieved for a given output power.</td>
<td>Moderate: Since both stator and rotor have windings, the output power to size is lower than BLDC.</td>
</tr>
<tr>
<td>Rotor Inertia</td>
<td>Low: Better dynamic characteristics.</td>
<td>High: Poor dynamic characteristics.</td>
</tr>
<tr>
<td>Starting Current</td>
<td>Rated: No special starter circuit required.</td>
<td>Approximately up to seven times of rated. Starter circuit rating should be carefully selected. Normally uses a Star-Delta starter.</td>
</tr>
<tr>
<td>Control Requirements</td>
<td>A controller is always required to keep the motor running. The same controller can be used for variable speed control.</td>
<td>No controller is required for fixed speed; a controller is required only if variable speed is desired.</td>
</tr>
<tr>
<td>Slip</td>
<td>No slip is experienced between stator and rotor frequencies.</td>
<td>The rotor runs at a lower frequency than stator by slip frequency and slip increases with load on the motor</td>
</tr>
</tbody>
</table>

3.2 Advantages of PMSM, comparisons
As can be seen in 3.1 and 3.2, the PMSM offers a lot of advantages: good torque speed characteristics, lack of starter circuit, no slip between stator and rotor... The only price to pay for all this advantages is the design of an appropriate and economical control unit, which is exactly the aim of this thesis. One of the challenges of this control is to overcome the difficulty of control compared to induction motor, this disadvantage is the need to know the angle of rotation in every moment.

Nevertheless as long as the only real disadvantage is related to the controls, it becomes a mere problem to be solved, depending only of the time and economical resources spent in developing strategies such as this thesis.

3.3 PMSM equations: first approach to a model

In PMSM the rotor consists of a permanent magnet and the stator which is formed by three equally spaced windings. The current flowing across these windings produces a magnetic field vector. Thus, controlling these currents a magnetic field of arbitrary direction and magnitude can be produced by the stator.

The attraction or repulsion between the stator field and the magnetic field of the rotor produces a torque. Advancing what will be discussed in the control section: for any position of the rotor (see 3.3 taken from [15]) there is an specific direction of the stator field, which gives maximum and at the same time a position which would produce no torque at all (This happens when the permanent magnet rotor is placed in the same direction as the field produced by the windings).

What is important at this point is to notice that the position of the rotor will play a crucial role in the description of the motor and its behavior. In figure 3.2 taken from [15] can be seen the schematics of the relative position between stator and rotor.

![Figure 3.2: Axis change](image_url)
Figure 3.3: Rotation of the magnet

The description of the motor need to be simplified by doing following assumptions:

- The space distribution of the windings is assumed to be sinusoidal and thereby the BEF should be treated as sinusoidal waves, too.

- There is no thermic effect neither on the resistors of the stator nor on the permanent magnets

- Magnetic material is supposed to be linear, there are no saturation effects.

- There are no losses in the iron of the machine.

Assuming all these points, the voltage for everyone of the windings in the stator is easy to calculate with the following system of equations:

\[
\begin{bmatrix}
    V_{as} \\
    V_{bs} \\
    V_{cs}
\end{bmatrix} =
\begin{bmatrix}
    r_s & 0 & 0 \\
    0 & r_s & 0 \\
    0 & 0 & r_s
\end{bmatrix}
\begin{bmatrix}
    i_{as} \\
    i_{bs} \\
    i_{cs}
\end{bmatrix} +
\frac{d}{dt}
\begin{pmatrix}
    L_{aa} & L_{ab} & L_{ac} \\
    L_{ba} & L_{bb} & L_{bc} \\
    L_{ca} & L_{cb} & L_{cc}
\end{pmatrix}
\begin{bmatrix}
    i_{as} \\
    i_{bs} \\
    i_{cs}
\end{bmatrix} +
\lambda_m
\begin{bmatrix}
    \sin(\theta_r) \\
    \sin(\theta_r - 2\pi/3) \\
    \sin(\theta_r + 2\pi/3)
\end{bmatrix}
\tag{3.1}
\]

where

\[
L_{aa} = L_s + L_a - L_b \cdot \cos(2\theta_r)
\]

\[
L_{bb} = L_s + L_a - L_b \cdot \cos(2\theta_r + 2\pi/3)
\]

\[
L_{cc} = L_s + L_a - L_b \cdot \cos(2\theta_r - 2\pi/3)
\]

\[
L_{ab} = L_{ba} = -\frac{1}{2} L_a - L_b \cdot \cos(2\theta_r - 2\pi/3)
\]

\[
L_{ac} = L_{ca} = -\frac{1}{2} L_a - L_b \cdot \cos(2\theta_r + 2\pi/3)
\]

\[
L_{bc} = L_{cb} = -\frac{1}{2} L_a - L_b \cdot \cos(2\theta_r)
\]
Being $r_s$ the stator resistance, $L_l$ the leakage inductance, $L_a$ and $L_b$ the inductance of the coils, $\lambda m$ is the amplitude of the flux and $\theta_r$ the position of the rotor.

This system is easy to understand, but presents several problems to handle with it: the system of differential equations is not linear, there are terms that depend on $\theta_r$, it is difficult to calculate the torque and all in all it is an invalid system for analysis.

The first problem to solve is the difficulty that present the equations to calculate the torque. To obtain constant torque flux created by the stator, it must also rotate as the flux of the magnet rotates with the rotor. In order to make the flux of the stator rotational, three sinusoidal currents displaced 120 grades must be created.

But this is not the only problem: the dependency of the terms on the angle of the rotor. It is mandatory to eliminate this last problem. By eliminating it, by making the terms independent from the angle, the resulting equation system will be linear and therefore suitable for analysis.

3.4 PMSM equations: valid model for analysis

As shown in figure 3.4 (taken from [15]) $\mathbf{I}$ is a current vector composed of three sinusoidal currents in a plane formed by the axis a, b and c, which are not orthogonal.

The first transformation to be done is Clarke’s transformation, which transforms a non-orthogonal system in an orthogonal one. This is the transformation $I_a$, $I_b$, $I_c$ in $I_\alpha$ and $I_\beta$ and a third component that is zero. Now the system is orthogonal but still has a dependency on the rotor angle. This can be solved applying a rotation matrix.

![Figure 3.4: Axis rotation](image)

The mathematics of these transformations will be discussed in more detail in the control section, here is enough to understand that an axis-reference change is needed and that this way the description of the motor which depended on the rotor angle becomes really easier and makes the analysis possible.

In this section is a priority to focus on the equations system obtained, the kind of black box in which the currents of the stator ($i_a$, $i_b$ and $i_c$) enter and the appropriate output currents for analysis ($i_d$, $i_q$ and $i_0$). Here is the black box:
\[
\begin{bmatrix}
i_0 \\
i_d \\
i_q \\
\end{bmatrix} = \frac{2}{3} \begin{bmatrix}
\frac{1}{2} \\
\cos(\theta_r) & \frac{1}{2} \\
-\sin(\theta_r) & \frac{1}{2} \\
\cos(\theta_r - 2\pi/3) & \cos(\theta_r + 2\pi/3) \\
-\sin(\theta_r - 2\pi/3) & -\sin(\theta_r + 2\pi/3) \\
\end{bmatrix} \begin{bmatrix}
i_{qs} \\
i_{bs} \\
i_{cs} \\
\end{bmatrix}
\] (3.2)

And here is the resulting equations system. Io is not used in the model, because Id and Iq are enough for describing the system and relating the motor speed and torque (mechanical characteristics) with the voltages (electrical characteristics):

\[
V_d = r_s i_d + L_d \frac{d}{dt} i_d - w_r L_q i_q
\] (3.3)

\[
V_q = r_s i_q + L_q \frac{d}{dt} i_q + w_r L_d i_d + w_r \lambda_m
\] (3.4)

In the equations 3.3 and 3.4 only appears a new value and it is the derivative of the rotor angle. The expression for the torque is obtained making a power balance between the electrical power (voltage \cdot current) and the mechanical one (torque \cdot rotation speed). The results of this balance are the following:

\[
\tau_e = \left( \frac{3}{2} \right) \cdot \left( \frac{n}{2} \right) \cdot (\lambda_m \cdot iq + (Ld - Lq) \cdot id \cdot iq)
\] (3.5)

\[
w_r = \frac{n}{2} \cdot w_m
\] (3.6)

Being \(\tau_e\) the torque, \(n\) the number of poles and \(w_m\) the rotor mechanical rotation speed. Here can be found the reason for selecting a SPMSM, having \(Ld\) and \(Lq\) the same value the equation 3.5 is simplified and the control of the PMSM reminds of a simple DC Motor control. Varying the input current the electric torque varies proportionally depending only on the constants \(\lambda_m\) and the number of poles.

3.5 Summary

In this chapter the concrete PMSM physical construction has been explained. The equations that model its behaviour have been discussed. All in all, the problem to solve in this thesis, the complexity of PMSM, has been presented. The described mathematical transformations to linearize the model of PMSM, a mandatory step to solve the problem, lead to the following section which deals with the solution: the control of the PMSM.
4 Control

In this section the solution (control) for the problem (PMSM) is presented. First of all, the general scheme will be discussed to make way for the concrete theory behind every single block. Afterwards the implemented scheme with its simplifications will be described in the same way from the complete scheme to the single blocks. But this time, in every block the code delivered with this thesis will be commented in order to let the reader understand the concrete solution proposed with its particularities which represent the contribution of this thesis.

4.1 Field Oriented Control

![General scheme implementation](image)

Figure 4.1: General scheme implementation

4.2 General complete scheme

As commented in chapter 1 the field oriented control is based on decoupling the flux and the torque. Now after having studied the PMSM model, in chapter 3, the details of this control will be easier to understand.

As mentioned before the main problem when analyzing the PMSM model is the difficulty of its non-linearity, for this reason d-q transformation or also called Park's transformation (proposed by R.H. Park in 1929) is one of the main blocks of the PMSM. In 4.1, this block can be seen with the motor currents and angle of the rotor read in the sensors as inputs and giving the transformed values as outputs.

This concept is part of the heart of FOC, because the d-q currents are just a clever way to decompose the real current in two components in special axis (orthogonal to the rotor’s magnetic field and parallel to it) which offer the special trick of maximal torque. As explained in 3, the torque in a SPMSM depends only on the q-current, for this reason the inputs of the inner loop in 4.1 are iq* and id*.

Demanding a zero direct current, the iq component is maximized making the stator flux always orthogonal to the rotor one and achieving the maximum torque. The out loop with its PID, translates...
the speed command to the needed iq* to reach the desired speed. This is due to the basic equations of
PMSM, first the one mentioned, that relates directly iq* and the torque (τe = 3/2λ_m n/2 iq) and secondly
the one that relates the derivative of the speed with torque (τ_e − τ_load = J dωm / dt).

These are the loops with their inputs, but how is the control done? Using the values corrected of the
transformed voltages (Vd and Vq) and the measured rotor position, a vector is created in a conceptual
space that decides the switches to open or close and all the involved times. This information is send
directly to a driver that will translate the digital values to the power signals required by the power
inverter. (This module is refered in 4.1 as Space Vector Modulation + Driver)

The feedback with the actual motor information is done by two current sensors (the other current can
be calculated using the others) and a position sensor, due to the high demands of precision a Hall effect
sensor is not enough like in BLDC motor controls (trapezoidal control), for this reason an encoder should
be used (this is the module sensor+proc, "proc" stands for process which means that a simple module to
calculate the speed using the rotor position signal is needed). The sensors are the A/D connection of the
general scheme. The inverter obeys to the demands of the space vector modulation thus creating the right
sinusoidal currents to feed the PMSM taking the energy from the battery power. The role of the inverter
(+ the driver) is also to connect the analog and digital world, they receive the digital signal to open and
close the switches and at the end they send an analog waveform to the motor. They are the D/A converter.

After this general overview, it is time to describe the therory related to every block in detail in order
to understand exactly how the control works.

4.3 Control concepts block by block

4.3.1 Space vector modulation

![Space Vector Modulation](image)

Figure 4.2: SVM Block

The target of the space vector modulation block is to command the open and close times of the switches
in the inverter. This is the main block of the field oriented control because it makes possible that the
switches are opened or closed depending on the position of the rotor’s motor, thus tracking the evolution
of the controlled plant.
In the paper [16], a new method which implies less heavy calculations is presented and this solution is used in this thesis for its simplicity.

The classical method of space vector modulation is based on the following steps: first depending on the angle of the rotor and the angle formed by Vd and Vq, the actual vector is positioned in the plane:

![Sectors in plane](image)

Figure 4.3: Sectors in plane

Each of the vectors in the plane represents which switches must be closed. For example, vector (101) is referred to switch T1 and T5 on. It is important to point that space vector modulation assumes that when T1 is closed the switch in the same branch (T0) is open and vice versa. Therefore only the signals to open or close T1, T3 and T5 (upper branch) are represented in the method. Every vector in the plane means then (T1 T3 T5).

Once the vector is placed in the plane, due to the angles, the vector is decomposed in two components (the two adjacent vectors) and one zero-vector (in the origin: 000 or 111). The decomposition in two vectors will give as result to components which are directly related to the time that this space vector will be used. As an easy example, just imagine the angle of the rotor is 180 degrees, Vq has some value and Vd is equal to zero. In this case the resulting vector would be placed exactly in the middle of region 5. This way the two adjacent vectors would rule (some time the vector 101 and some amount of time the vector 001). That implies that the switch T5 would for sure more than T1 and that T3 would be closed only when the zero vectors work.

To complete one cycle of the SVM a zero-vector (000, T1, T3, T5 opened or 111, all closed) has to conduce the rest of the time T0. This will be explained with more in more detail in the next block called ‘sequence’. The method used in this thesis avoids calculating the angle of Vd and Vq (in 4.3 from [16] the angle called $\theta_{dq}$), and this way is not necessary to implement function arctan. The steps of this method are very clear and are very simple:

1) A transformation is used to translate Vd, Vq in Va, Vb and Vc

$$
\begin{bmatrix}
    v_{d0} \\
    v_{b0} \\
    v_{c0}
\end{bmatrix} =
\begin{bmatrix}
    1 & 0 & -\frac{1}{2} \\
    -\frac{1}{2} & \sqrt{3} & -\frac{1}{2} \\
    -\frac{1}{2} & -\frac{\sqrt{3}}{2} & \frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
    \cos(wt) & \sin(wt) \\
    \sin(wt) & \cos(wt)
\end{bmatrix}
\begin{bmatrix}
    v_d \\
    v_q
\end{bmatrix}
$$

(4.1)
For simplicity, the notation of the mentioned paper is kept, to the model of this thesis, \( \omega_t \) is \( \theta_r \).

2) And the conduction times are given by the expression:

\[
\begin{bmatrix}
T_a \\
T_b \\
T_c
\end{bmatrix} = \frac{T_s}{V_d} \begin{bmatrix}
v_{a0} - v_{min} \\
v_{b0} - v_{min} \\
v_{c0} - v_{min}
\end{bmatrix}
\] (4.2)

3) At last the \( T_0 \) is given by:

\[ T_0 = T_s - \frac{T_s}{V_d} (v_{max} - v_{min}) \] (4.3)

Where \( v_{min} \) and \( v_{max} \) are the minimum and maximum of \( V_{ao}, V_{bo} \) and \( V_{co} \).

As seen in the table developed by the authors of [16], with this algorithm square root and arctan functions are avoided.

As it will be explained in the implementation part to achieve this simplification a new challenge is presented, to calculate the maximum and minimum of the three values, this will be solved by the creation of two specific functions for this task.

4.3.2 Sequence

This block is developed apart in order to clarify more the design and to make the search of errors easier and faster, but it really belongs to the space vector modulation module properly. The main goal of the block 'sequence' is to create the signals that will be sent to the driver with the orders of opening and closing the switches.

The function of this block is to use the "close" times calculated in the space vector modulation and build the correct sequence for every switch using this information.

There are different approaches to generate the signals of the switches, depending on the selection of the zero vectors, all of them focused on the losses by switching and the THD due to transients. Of all the different existing methods, the decision of taking the right aligned sequence is done due to its simplicity and coherence with the method used in SVM. The right aligned sequence is based on the equal participation of the two zero-vectors (000 and 111). At the beginning of switching cycle the open switches are all opened (000) and at the end of the cycle they are all closed.
In easy terms: the SVM indicates the conduction times T0, T1, T3 and T5. One of T1, T3 and T5 will be the maximum and another will be zero (because they are calculated subtracting to $v_{a0}, v_{b0}$ and $v_{c0}$ the minimum of them). If for example at a certain moment T5 is zero and T1 is the maximum, the sequence will be: First all of them are opened (zero vector 000), after T0/2 S1 will start to conduce during (T1-T3), then S3 will be switch on and it and S1 will conduce during T3. Finally, S5 will be closed and the three of them will conduce during T0/2 (zero vector 111).

A general desired sequence can be seen in the figure and in the implementation part, a simulation will evidence that the used algorithm is correct obtaining the same result.

![Switch sequence diagram](image)

**Figure 4.5: Switch sequence**

### 4.3.3 Driver

The driver of the switches transforms the bit signals created by the SVM and the sequence in quantities, that means, it creates the signals able to drive the switches in the inverter. A logic '1' is translated in 5V and a '0' in 0V.
4.3.4 Inverter

The inverter consists of three branches with 2 switches and 2 diodes in parallel each one (see 4.8 from [3]). Here, the conversion DC-AC is done using the DC source and opening and closing the switches as planned in the control-strategy. The model of a simple power diode and power Mosfet or IGBT are used in this block:

![Inverter Block](image1)

![Inverter circuit](image2)

The inverter is implemented using the advantages of VHDL-AMS, declaring the different components of the circuit and connecting them. It is also important to underline that the inverter block has the task to convert D/A, because using the orders of the driver and a battery we get the analog waves to feed the motor.

To simplify the model a decision has been taken for the components of the circuit: the battery is a constant voltage (in real world it should be modeled taking care of parameters such as SOC, state of charge, making it more complex), the diode and the power switch behave like resistors in several regions and their behavior does not vary with temperature for example.
4.3.5 Subtractor

![Subtractor Block](Figure 4.9: Subtractor Block)

Again a block which is simple and does not need an own test, subtractor gets the value of the desired current and the real current and gives as result the difference between them.

4.3.6 PID

![PID Block](Figure 4.10: PID Block)

The PID block and PI blocks are the classical block used in control theory:

\[
\text{output}(\text{PID}) = \left( \frac{Ki}{s} + Kp + Kd \cdot s \right) \cdot \text{input}
\]

\[
\text{output}(\text{PI}) = \left( \frac{Ki}{s} + Kp \right) \cdot \text{input}
\]

Like in every control problem the PID is a crucial block to implement. As said before, for controlling the PMSM one PID is necessary for the speed loop and two PI’s are used to control the q-current coming from the PID and the d-current desired to be zero.

The tuning of PID is not a trivial question. As masterfully treated in [17], PID sometimes is halfway between science and art. At the end the science of PID tuning demands an easy simplification of the blocks to control the system depending on the delays caused by the switching for example and the equations to solve with this approach are usually very cumbersome (like in [18], [19], [20]).

Nevertheless all the PID tuning methods proposed are always proposed for the implementation of the PMSM controls. As already advanced before and will be seen detailed in the next section, the aim of this thesis is to model and simulate all the system, to implement it some simplifications that are assumed.
For this reason and because every block, like the SVM one, has been taken or implemented in an original way not comparable with the final model to implement, the use of these papers was not worth due to these particularities.

In further works section some advices in the actual direction of PID tuning will be given to the reader interested in implement the hardware of the control using this thesis as basis, but the application of them is out of the time range for this thesis. Just as a little preview: PMSM and its controls are a fuzzy system. This makes the PID tuning a complex task and there are some alternatives nowadays, that require some time for implementation, based on computational methods to optimize the control such as Particle Swarm Optimization (PSO) or neural network algorithm’s (this is the case of [21], [22] or [23]). The PID tuning has been done linearizing the system and simulating it with the help of Simulink. In the results chapter (simulations) the PID tuning will be analyzed in detail. In any case it is important to remark that at the end it has been simplified assuming equal values for the two PT’s (like done in [9] ) and playing only with the Kp, Ki and Kd of the w loop.

The decision for this half-manual tuning, as said before, lies on the fact that the model or system analyzed in this thesis has its particularities, is not easy to find papers that do not work with current sensors or focus on the control for a vehicle application (big J, load...). The linearization of the system offers a quite good approximation.

4.3.7 PMSM

As explained in the 'PMSM motor' section, a change of axis is needed in order to make the model and simulation of the motor manageable. With the aim of making it clearer the motor model has been split in three modules: modul, motor and integrator.

In the modul box the transformation Va, Vb, Vc to Vd, Vq is done using the angle $\theta$ given by the integrator. The motor box contains the main equations of the motor and it gives as a result the speed that enters the integrator box, which will return the mentioned angle $\theta$ thus closing the loop.

The equations used in every block are the following:
Modul:

\[
\begin{bmatrix}
    v_d \\
    v_a \\
    v_b \\
    v_c \\
\end{bmatrix} = \frac{2}{3} \begin{bmatrix}
    \cos(\theta_r) & \cos(\theta_r - 2\pi/3) & \cos(\theta_r + 2\pi/3) \\
    \sin(\theta_r) & \sin(\theta_r - 2\pi/3) & \sin(\theta_r + 2\pi/3) \\
    \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\
\end{bmatrix} \begin{bmatrix}
    1 \\
    2 \\
    1 \\
\end{bmatrix}
\]

PMSM:

\[
V_d = r_s i_d + L_d \frac{d}{dt} i_d - w_r L_q i_q
\]

\[
V_q = r_s i_q + L_q \frac{d}{dt} i_q + w_r L_d i_d + w_r \lambda_m
\]

\[
\tau_e = \frac{3p}{2} \lambda_m \cdot i_q
\]

\[
\tau_e = \tau_{load} + J \frac{dw_m}{dt} + Bw_m
\]

\[
w_r = \frac{p}{2} \cdot w_m
\]

The load torque depends on several factors, the expression used for it has been taken from the paper [9]:

\[
F_{rolling} = f \cdot M \cdot g
\]

\[
F_{grade} = M \cdot g \cdot \sin(\alpha)
\]

\[
F_{aerod} = \frac{1}{2} \cdot C_x \cdot S \cdot l_s \cdot v^2
\]

\[
\tau_{load} = R_{wheels} \cdot (F_{rolling} + F_{grade} + F_{aerod})
\]

The forces that produce a torque against the movement of the car are caused because of aerodynamics, friction and the presence of a possible slope. All these factors work against the torque produced in the electric motor.

Integrator:

\[
\theta_r = \int_{0}^{t} w_r(\epsilon) d\epsilon + \theta_r(0)
\]
4.4 Complete concrete Scheme

In this part the implemented model will be explained in detail. First of all the generic scheme will be described and later each of the single blocks code will be analyzed in detail and tested.

![Concrete Scheme Implementation](image)

Figure 4.12: Concrete scheme implementation

In this thesis the task is focused on the simulation of the system, therefore some simplifications have been assumed in order to make the simulation more tractable. These simplifications are described in the next lines and afterwards, the implementation block by block will be done like in the previous section.

4.5 Assumptions and Simplifications

In the first approach to implement and simulate the control explained in the previous section some simplifications will be assumed:

- In the first approach, the encoder is not modeled. The real value of the rotor’s angle is read from the motor block and used for the control. With real motor an encoder will be needed to give digitally the value of this magnitude that influence all the control.

- The values $i_q$, $i_d$ are available from the motor box as real values. The sensors are not part of the controller, that means in this first approach it is used directly the value of the real $i_q$, $i_d$ in the motor (as quantity). In real motor this access did not exist, first of all because $i_d$, $i_q$ are the result of transformations and they are only conceptual.

- In real world the sensors described in the previous two assumptions would give a digital value. Using this value the $d$, $q$ transformation should be done (park+Clarke transformation). For this aim is necessary to implemented functions such as trigonometric (cosinus and sinus) and they are not simple to be digitally synthetized. Therefore because it is beyond the aim of this study, the two previous assumptions will be assumed having in mind that the next step to develop further this thesis is not complicated but requires some time more to convert the different assignments of quantities in the codes by digital processes that can work with the digital signal received.
The same should be applied to the value of the battery voltage, which in this thesis is treated as a real quantity and the SVM uses it for calculating the switching times, but in real life, the other inputs of SVM (voltages \( v_d \) and \( v_q \), and position) would be digital words and in the same way should be translated with a A/D converter the voltage measured in the battery for processing the information in the controller.

All this assumptions are done for making the model and its control easier from a conceptual point of view, the direction to take for improving the model in order to synthetize the controller will be treated more deeply in further works chapter.

In the following sections the implementation of every block from 4.12 will be analyzed. It is interesting to have a look at the code in order to follow the concrete explanations and to understand the decisions taken.

4.6 Implementation block by block

4.6.1 Space vector modulation

As seen in the theory part, the space vector modulation algorithm chosen has the advantage of avoiding computational burden (such as arctangent and square root). Therefore the proposed SVM algorithm gives directly the conduction times of switches to modulate the voltage vector to be generated without considerable computational burden.

Inputs: \( V_{dc} \) (battery), \( V_d \), \( V_q \) (from the comparison motor-desired), \( \theta \) (from the motor) and \( T_s \) (sampling time).

Outputs: \( T_0 \), \( T_1 \), \( T_3 \) and \( T_5 \) (conduction time of the switches)

For the implementation of the SVM two new functions have been created in the package (mypackage): max and minim, which receiving three real values return the maximum or the minimum of the three.

Inside the SVM a transformation is done, from \( d, q \) reference to \( a, b, c \) reference and with the values of \( v_a \), \( v_b \) and \( v_c \), max and minim functions are called. Having the values of the maximum and minimum, the value of the battery and the sampling time (\( T_s \)), the conduction times are calculated.

To remark here is the fact that the conduction times calculated are the times if the zero vector would be \((000)\). But in this control another scheme will be used which shares \((000)\) and \((111)\) zero vectors.

Another important aspect to take into account is that a process is used commanded by a clock signal. This clock signal is used to sample the input \( \theta \).

For this reason when the whole model is connected and implemented together it is important to remember that this clock is related with the variation of the input \( \theta \).

In the simulation testsvm.vhdll a test bench was created in order to check the correctness of this module. A test bench was extra done only in the cases of SVM and sequence module, due to their importance in the control. Driver, inverter and the other blocks were also tested but the decision to include only the most significant tests was taken.
In the svm test, a linear input $\theta$ with period $200T_s$ is used with constant $V_d$ and $V_q$ (50V). The battery voltage is 400V and the switching time $T_s$ is 50 us (frequency 20kHz).

A clock signal was generated with frequency $20T_s$, that means 10 clock edges during the increase of $\theta$. For this reason a change in the conducting times is expected in every clock change, because they are calculated using the voltages $V_d$, $V_q$ (in this case constants) and $\theta$, which varies constantly.

As it can be seen in 4.13 in the two cursors used, the time of $T_0$ added to the maximum of the two conducting switches is equal 50us. (First cursor: $35.2314 + 14.78766 = 50.0\text{us}$, second cursor: $36.3593 + 13.6407 = 50.0\text{us}$).

Another aspect to underline is the fact that always two switches are selected and that due to the linear variation of the speed, the two switches selected change with a sequential order, passing through the different sectors of the space vector plane shown before.

It is important to highlight, it is really understood how this block works that the time calculated here is not the total time of the switching for the switches (that means: $T_3$ is not the total amount of time $T_3$ will be closed). As it will be clearer in the test of the sequence, $T_0/2$ should be added to every switch in order to get the total amount of time it is closed. For example, for the values in the first cursor: $T_1$ would be closed during $14.78766 + 35.21234/2 = 32.39 \text{us}$, $T_3$ during $10.82532 + 35.21234/2 = 28.43 \text{us}$ and $T_5$ only during $T_0/2 = 17.60617\text{us}$.

Just to enter already in the next test, with this values the desired switch signal following a right aligned scheme would be: first during $T_0/2$ ($17.60617\text{us}$) all the upper switches are opened, after this the one who has the biggest conducting time, $T_3$, closes and after $28.43\text{us}-16.2\text{us}$ ($T_3-T_1$) = 12.23 us it conduces alone, being 1 and 5 opened. Then after this 12.23 us $T_1$ closes to, and both of them conduct during 16.2 us ($T_1$) to finish conducting all of them $T_1$, $T_3$ and $T_5$ during $T_0/2$ ($17.60617\text{us}$) again.

For this reason the biggest time obtained from the svm module added to $T_0$ should be and is, as proofed in the test, equal to the switching frequency $T_s$.

![Figure 4.13: Space vector modulation test bench](image-url)
4.6.2 Sequence

As explained in the theory part, this block will create the signals to open and close the switches according to the right aligned patron discussed before. For this task it will need:

Inputs: T0, T1, T3 and T5 (conduction time of the switches)

And will give:

Outputs: C0, C1, C2, C3, C4, C5 bits.

Now, the same test is modified in order to transmit this times calculated to the sequence blocks allowing it to build the commented sequences.

Taking a brief look on the code, it can be seen that the implementation has been done using signals. For this reason, when a new value of theta is read by the svm module and T0, T1, T3 and T5 are instantly calculated, the signals commanding c0, c1, c2, c3... present a delay of one clock.

Due to the signal character of the implementation, in the beginning (when the initial values of T0, T1, T3 and T5) are equal zero, the output of the switches commands is kept zero in order to avoid 0 us break statement which lead into errors.

For this reason a little error is introduced in the beginning and a delay during the normal behavior, but regarding the frequency of switching and comparing it to the inertial behavior of the motor, it can be conclude that this assumptions can simplify the model (remaining always in mind for possible future errors).

In figure 4.14 the calculated conduction times for a certain moment can be seen in the cursor (T0 = 34.88 us, T1 = 9.63, T3 = 15.12 us and T5 = 0). These times do not correspond to the switching sequence shown in c1, c3 and c5. This is due to the reason already mentioned: the clock delay.

Effectively if the next clock change is lighted (figure 4.15), the conduction times ordered one clock before correspond exactly to the switching scheme.

Figure 4.14: Sequence test bench
During 17.44us c0, c1 and c3 are opened (exactly 34.88us/2 calculated before) the same time they are all closed at the end and T3 for example would be watching the figure \(32.57 - 17.44 = 15.13\text{us}\) exactly the value calculated during the rising edge of the clock.

It can be concluded that the svm and sequence blocks, that are the main calculations of the control part work as expected and that the insignificant simplifications done are valid.

4.6.3 driver

As explained in the theory part, this block will create the signals to open and close the switches according to the right aligned patron discussed before For this task it will need:

Inputs: C0, C1, C2, C3, C4, C5 bits

And will create:

Outputs: p0, p1, p2, p3, p4, p5 electrical quantities. As said before, the test of this block is not included in the thesis.

4.6.4 inverter

The inverter is the block responsible for the feed of the motor. As said in the theory content, it is designed using the advantages of VHDL-AMS as a circuit (connecting easy components such as diodes or switches).

Inputs: Vdc, p0, p1, p2, p3, p4, p5 electrical quantities

And will create:

Outputs: pa, pb, pc electrical quantities.

As said before, the test of this block is not included in the thesis.

4.6.5 Substractor

Again a very simple block modeled with quantities.

Inputs: idesired, ireal (real quantities)
Outputs: i (real quantity)
4.6.6 Motor

The most important aspect of the motor model is to clarify that the equations chosen worked with real quantities and not as electrical ones. The advantages of working with real quantities is that the analysis of the solver (Kirchoff’s laws for every node, current loops...) is easier and is allowed, because it is only a model of the motor just to study how it would work by being driven by the controller synthetized in this thesis.

For this purpose an electrical-real module is necessary to connect the motor with the output of the driver. This decision is a particularity of the model of this thesis.

Another fact should be highlighted and it is that the modeling of the motor using three different blocks is a particular solution brought by this thesis, too. The first models of the PMSM in VHDL AMS used in this work represented only one block with all equations and the compilator had some problems. Finally the decision to separate the motor in the three blocks explained avoiding the derivative of the angle using an integrator were taken with succesful results.

Input: Va, Vb, Vc (electrical)—transform—Va, Vb, Vc (real)

Output: τ, we, wm, id, iq

It is also important to remark that the speed should use the initial condition that can be seen in lines 49-55 of the pmsm.vhdl code to start the iterations at the logical value of 0 rad/s.

Having arrived at this point, it is time to simulate all the components together, because testing the motor alone is not possible, its inputs shall be controlled and this is the aim of the thesis. This test will be carried in the following section due to its importance.

4.6.7 Measure

In the implementation done finally, as said before, the d-q currents read from the model are used directly, while with a real motor only the stator currents a, b and c would be available. In the implementation of this thesis the driver supplies voltages to the module mentioned before that transforms Va, Vb and Vc in Vd and Vq and with this voltages the currents id and iq are calculated. This way, the currents ia, ib and ic are not used in the model and therefore cannot be read. To solve this matter, the block "measure.vhdl" was incorporated to transform the currents id and iq into the stator currents ia,ib and ic which are interesting to observe while running simulations.

Input: id, iq

Output: ia, ib and ic

The inverse d-q transformation can be seen in the code delivered with this thesis.

4.7 Summary

In this chapter first the general control scheme to control a PMSM using FOC was presented. Then the different blocks and the theory behind them were commented. The next point treated was the simplified control scheme applied and the assumptions for this simplification and finally every block modeled was presented explaining the reason for the different decisions taken such as split the motor in three blocks.
5 Simulations

In the previous section, the election of the language was reasoned and in section 3 the theoretical model of a PMSM was introduced. 4 dealt with the code used to model the PMSM and its controls and now is time to present the results.

In this chapter the reader will find first a short preparation of the mechanical concepts to be able to understand the different simulations run. After the brief preparation, first the PID tuning method and later the results will be presented.

The reader will find that some different situations of the daily urban drive have been planned to be reproduced in the simulations, the reasons for these values will be given. The tests realized were concretely: the acceleration 0-100km/h, the overcome of a slope and finally a european driving cycle.

After the presentation of the results, a short summary will close the chapter to emphasize the validation of the results obtained.

5.1 Preparation

5.1.1 The BEV for which the engine is designed

Electric cars are used in different automotive application nowadays. While there are some bold attempts of powerful sports car (such as Tesla Roadster that reaches 60mph in 3.7s), the major effort on electric vehicles is done in urban vehicles.

The autonomy of the electric car, due to the nature of the batteries and the lack of an adapted electric power system (in development) make the urban cars the perfect application for a PMSM motor.

While the first urban electric cars were more focused on small prototypes for two persons and very low weight (in order to let the motor work with less demanded torque), nowadays a lot of important car brands are adapting their full-size cars models to electrical powertrain technology (see fig 5.1)

Table 5.1: List of full-size BEV

<table>
<thead>
<tr>
<th>Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coda Sedan</td>
</tr>
<tr>
<td>Ford Focus Electric</td>
</tr>
<tr>
<td>Mitsubishi iMiiev</td>
</tr>
<tr>
<td>Malone TAZR</td>
</tr>
<tr>
<td>Nissan Leaf</td>
</tr>
<tr>
<td>Peugeot iOn</td>
</tr>
<tr>
<td>Citroën C-Zero</td>
</tr>
<tr>
<td>Renault Fluence ZE</td>
</tr>
<tr>
<td>Citroën C-Zero</td>
</tr>
<tr>
<td>Smart ED</td>
</tr>
<tr>
<td>Tesla Model S</td>
</tr>
<tr>
<td>Think City, Europe</td>
</tr>
<tr>
<td>Wheego Whip LiFe</td>
</tr>
</tbody>
</table>
The desired car in this thesis is thought for the modern city, that means it should work perfectly at low speeds and then since the modern cities are growing in surface terms and every day more suburbs are populated it should be able to be driven at a maximum speed of 130 km/h (like the Mitsubishi i-Miev) to reach the city center in half an hour. The maximum power at high speeds will be around 70kW (i-Miev gives around 49) in order to assure the transmission of enough power to the wheels.

The specifications related to the model depend strongly on the load torque demanded by the environment, for this reason a short overview of these concepts is presented in the next paragraphs.

As it can be seen in 5.1 taken from [9] there are three basic forces responsible for the creation of the so called load torque.

As explained in 4 these forces represent: first the friction due to the weight of the car, the presence of a slope and the force due to the fluid opposed to the movement, in this case, the air. This can be understood in the following equations:

\[
F_{\text{rolling}} = f \cdot M \cdot g
\]

\[
F_{\text{grade}} = M \cdot g \cdot \sin(\alpha)
\]

\[
F_{\text{aerod}} = \frac{1}{2} \cdot C_x \cdot S \cdot l_s \cdot v^2
\]

And using the wheel’s radius the load torque is explained

\[
\tau_{\text{load}} = R_{\text{wheels}} \cdot (F_{\text{rolling}} + F_{\text{grade}} + F_{\text{aerod}})
\]

Using these equations the specifications for the different simulations will be defined by means of the driving situation.

5.1.2 The chosen PMSM

The first thing to mention in this section is the difficulty to find a PMSM according to the use in automotive application. Most papers and catalogues (such as) have been consulted but it had been always difficult to find a motor for an application of power above 50kW.

Being the i-Miev of Mitsubishi the perfect model (for its specifications) the first attempt was to find its characteristics on the net, but although having read the torque, maximum speed values hundred of times, the concrete parameters such as inductances, stator resistor...were never to be found. The decision
to ask directly to Mitsubishi information service explaining the reasons for the need of these parameters conclude with no answer by the time being.

For these reasons and having in mind that the important fact is the system and how to control it, the motor of [9] was taken with some changes. In general paper [9] is a very clear approach to the goal of this thesis and the car: same controls, same desired application...and compared to other specifics papers that focus completely on the control but not on the mechanics of the system to control is a very useful reference. For this reasons the main parameter of the PMSM have been taken from there.

In 5.2 the electric parameters of the chosen PMSM with some changes are presented. The decision that were taken were: First the voltage DC used to feed the motor is 400V (100V more than the paper) because the car will be submitted to different slopes and looking at the little information about the simulations in paper [9] and the simulations run in this thesis, it seems to be a rational value . And second: the parameter Jm and λm are not given, therefore they are taken from typical motors values regarding that the important fact is the order of magnitude to simulate and get logical results.

### Table 5.2: Electrical parameters of the chosen PMSM

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>4</td>
</tr>
<tr>
<td>Lq–Ld</td>
<td>0.416mH</td>
</tr>
<tr>
<td>Rs</td>
<td>0.05 Ω</td>
</tr>
<tr>
<td>Jm</td>
<td>0.002 kg/m2</td>
</tr>
<tr>
<td>Vbat</td>
<td>400V</td>
</tr>
<tr>
<td>λm</td>
<td>0.192Wb</td>
</tr>
</tbody>
</table>

### Table 5.3: Mechanical parameters of the chosen PMSM

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cx</td>
<td>0.55</td>
</tr>
<tr>
<td>ρ</td>
<td>1.2 kg/m3</td>
</tr>
<tr>
<td>S</td>
<td>1.8 m2</td>
</tr>
<tr>
<td>Rwheels</td>
<td>0.26m</td>
</tr>
<tr>
<td>rm</td>
<td>3</td>
</tr>
<tr>
<td>M</td>
<td>800kg</td>
</tr>
<tr>
<td>f</td>
<td>0.025</td>
</tr>
</tbody>
</table>

Now, in order to prepare the simulation of the mechanical system is time to concrete an equation presented in 3 which had not been analysed in detail till now:

\[
\tau_e - \tau_{load} = Jm \cdot \frac{dw_m}{dt} \quad (5.1)
\]

This equation that seems extremely simple needs some deeper concepts. First of all is important to know that the BEV used in the thesis has a reduction ratio (as can be seen in 5.3) rm = 3. This affect to every mechanical variable: in the equation every term has to be related to the motor reference since this equation tries to describe the rotation of the motor. In 5.2 transmission’s schematic is presented.
Figure 5.2: Transmission motor-wheels of the car

Now a table (5.4) has been included in order to concretize how the different variables have to be translated to the motors reference.

Table 5.4: Translation to motor reference

<table>
<thead>
<tr>
<th>Motor reference</th>
<th>Wheel reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \tau_{load} )</td>
<td>( \tau_{load} )</td>
</tr>
<tr>
<td>( \frac{J_{wheels}}{3^2} )</td>
<td>( J_{wheels} )</td>
</tr>
<tr>
<td>( \frac{J_{car}}{3^2} )</td>
<td>( J_{car} )</td>
</tr>
</tbody>
</table>

For this reason the \( \tau_{load} \) presented in the previous section that was the result of the different forces (friction, slope and aerodynamics) will be divided by 3 in equation 5.1. \( \tau_e \) and \( w_m \) are already related to the motor reference, only \( J_m \) has to be adapted. In [24] and in [8] the following equations appear as solution:

\[
J_m = \frac{J_{wheels} + J_{car}}{r m^2} + J_{motor} \tag{5.2}
\]

In 5.2 \( J_{wheels} \) is taken from [7], \( J_{car} \) is calculated from [8] taking in account that \( m \) is 800 kg and \( r \) is 0.26 m. In 5.5 are the results for the final \( J_m = 3.5 \text{ kg m}^2 \) used in the simulation.

Table 5.5: Calcule of \( J_m \)

| \( J_{car} \) | 27.04 kg m\(^2\) |
| \( J_{wheels} \) | 2.7 kg m\(^2\) |
| \( J_{car} + J_{wheels} \) | 29.74 kg m\(^2\) |
| \( \frac{J_{car} + J_{wheels}}{r m^2} \) | 3.30 kg m\(^2\) |
| \( J_{motor} \) | 0.002 kg m\(^2\) |
| \( J_m \) | 3.302 kg m\(^2\) |

In 5.6 the different linear speeds of the vehicle are related with the angular speed of motor and wheels. The equations used for these transformations are just conversion factors and the unique value related to the car is the Rwheel (0.26m) and the transmission factor \( (r m = 3) \):

\[
v \left( \frac{km}{h} \right) \cdot \frac{1000m}{1km} \cdot \frac{1h}{60min} \cdot \frac{1rev}{2\pi \cdot 0.26m} = w_{wheel} \left( \frac{rev}{min} \right) = \frac{rev}{min} \]

\[
w_m = 3 \cdot w_{wheel}
\]
Table 5.6: Reference table for linear and angular speeds

<table>
<thead>
<tr>
<th>v (km/h)</th>
<th>w.wheel (rev/min)</th>
<th>w.motor (rev/min)</th>
<th>w.motor (rad/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>30.60</td>
<td>91.82</td>
<td>9.62</td>
</tr>
<tr>
<td>5</td>
<td>51.01</td>
<td>153.03</td>
<td>16.03</td>
</tr>
<tr>
<td>10</td>
<td>102.02</td>
<td>306.07</td>
<td>32.05</td>
</tr>
<tr>
<td>15</td>
<td>153.03</td>
<td>459.10</td>
<td>48.08</td>
</tr>
<tr>
<td>18</td>
<td>183.64</td>
<td>550.92</td>
<td>57.70</td>
</tr>
<tr>
<td>20</td>
<td>204.04</td>
<td>612.13</td>
<td>64.10</td>
</tr>
<tr>
<td>30</td>
<td>306.07</td>
<td>918.20</td>
<td>96.15</td>
</tr>
<tr>
<td>32</td>
<td>326.47</td>
<td>979.42</td>
<td>102.56</td>
</tr>
<tr>
<td>35</td>
<td>357.08</td>
<td>1071.24</td>
<td>112.18</td>
</tr>
<tr>
<td>40</td>
<td>408.09</td>
<td>1224.27</td>
<td>128.21</td>
</tr>
<tr>
<td>50</td>
<td>510.11</td>
<td>1530.34</td>
<td>160.26</td>
</tr>
<tr>
<td>60</td>
<td>612.13</td>
<td>1836.40</td>
<td>192.31</td>
</tr>
<tr>
<td>70</td>
<td>714.16</td>
<td>2142.47</td>
<td>224.36</td>
</tr>
<tr>
<td>80</td>
<td>816.18</td>
<td>2448.54</td>
<td>256.41</td>
</tr>
<tr>
<td>90</td>
<td>918.20</td>
<td>2754.60</td>
<td>288.46</td>
</tr>
<tr>
<td>98</td>
<td>999.82</td>
<td>2999.46</td>
<td>314.10</td>
</tr>
<tr>
<td>100</td>
<td>1020.22</td>
<td>3060.67</td>
<td>320.51</td>
</tr>
<tr>
<td>120</td>
<td>1224.27</td>
<td>3672.81</td>
<td>384.62</td>
</tr>
</tbody>
</table>

5.1.3 The tests specifications: Suburbanization

As mentioned before, the goal vehicle to control is an urban car thought for the coming cities, the specifications to fulfill are related to this environment. Taking a brief look at cities growing trends the word "suburbanization" appears on the front page.

Nowadays more and more residents of metropolitan regions work within the central urban area, while they fix their residence in suburbs (satellite communities around the center). In Barcelona, for example, there are more than 2.5 million people residing in suburbs closer than 25 km from the city (Source: INE. Instituto Nacional de Estadistica) and the same phenomena is being experienced in lots of big cities all around the world.(see 5.3 from [25])

![Population growth and the growth of built-up areas (mid 1950s to late 1990s), selected European cities](image-url)
The electric car for those people who commute from their home to the center is a perfect option. I-Miev and most electric cars have autonomies above 100 km, enough to drive from home to work and come back and still have stockpile for traveling around the city center or the suburb. Normally these growing cities are connected with highways that make possible to reach the center in less than one hour (see 5.4 from [25])

All in all, the suburbanizations inhabitants represent a growing market demand which would take a lot of profit from a vehicle able to get charged in some hours, cheap to charge, respectful with the environment and offering silent driving, but all these positive aspect should be combined with a valid mechanical performance, that means: be able to be driven not only at speeds around 100km/h with the same ease as a conventional vehicle but also to provide a light driving performance in the city streets. This mechanical performance will be proofed in the following typical cases.

Figure 5.4: Built-up area, road network and population increases, selected EEA countries.

5.2 Results

As said in the introduction the simulations correspond to different situations all of them typical of a daily urban drive. In every simulation the load torque, depending on the external factors (see 4) and the desired specifications like motor speed will be set to understand if the model is working or not.

The first thing to underline before go into detail is the simulation platform used and its limitations. The software used for modeling and simulating in this thesis is Questa ADMS from Mentor Graphics and as it can be found in the distributor’s website, it "gives designers a comprehensive environment for verifying complex analog/mixed-signal System-on-Chip designs". They also underline that "ADMS combines four high performance simulation engines in one efficient tool: Eldo for general purpose analog simulations, Questa for digital simulations, ADiT for fast transistor-level simulations and Eldo RF for modulated steady state simulation". The machine, in which the software was installed, comes standard with a Dual Core processor with AMD E2 VISION technology.

During the realization of this thesis, it has been realized that the environment just described is not ideal for the multiple simulations that were thought to be run. The switching time of the speed that provides the information to the space vector modulation module, is crucial for the torque ripple, for example. As it will be demonstrated later, the higher the switching frequency the less ripple in the electric torque. For the real implementation, a switching frequency higher than 3kHz would be desirable,
but in practical terms there is no time for simulating hundreds of samples if to run 15s of the motor speed response takes more than one hour of waiting for the results. For this reason and having in mind that the PID tuning of a fuzzy system such as this would require a long time of simulations to be adjust to the desired specification, a organized procedure was necessary. This procedure was: First, based on the master thesis [26] page 44, the system is reduced to the linear system that can be seen in the upper box of the figure.

Figure 5.5: Procedure for PID tuning and simulations

In this system it is assumed that the current loop is at least 10 times faster than the speed loop. In the case of this thesis it is completely truth since the currents are quantities and therefore an actual feedback value is always available.

The blocks used in this simplified system are just the basic equations of torque-iq and torque-speed. As a variation of the solution proposed in [26], in this thesis, the load torque has been included in the simplified system, because of its importance in the speed evolution. The linear system used is treated with Simulink the perfect platform to carry the Optimization with experimental designs due to its fast response. A simulation of the simplified model in Simulink takes seconds, while the whole simulation with Questa takes amounts of minutes close to hours. The experimental results for the optimization with Simulink can be found in the next subsection, after it,
the results run in Questa using the PID parameters obtained with Simulink are presented and commented.

The aim of the following simulations is to transmit to the reader the experiences to use the model in different situations and validate that the response is effectively the desired one.

5.2.1 PID tuning. Optimization of experimental designs

The procedure used for the PID tuning uses a linearization of the system. In this section, the tuning of the PID for achieving the 0-100km/h acceleration is presented step by step. It should be underlined that the values of the P and I of the inner controllers for the currents were kept always at 0.1 because this loops are faster than the big loop.

The first points to describe are the requirements used in the diagramm 5.6: the speed command arrives to 100km/h in 8s, therefore the rise time should be close to it: that means rise time from 0 to 98 % of the final value should be 8s. The steady state error (after 15s) should be less than 1% and present no oscillations. The specifications related to the speed can be seen in the blue boxes. The desired torque (black boxes) should have an overshoot less than 30% and a fall time, after assuming constant speed at 8s, shorter than 1 second, while the ripple of the torque should be kept as small as possible.

The difficulties in the control are due to the fact that two signal responses should be regarded: the speed and the torque. Even if they are directly related, as it will be shown now, both have to be regarded separately while tuning the PID.

Before starting with the procedure, underline that the Simulink tool is real powerful for such simulations and the idea of using it came after a big amount of test using Questa. The PID tuning using Simulink really takes only some minutes and the results that are validated in the Simulations chapter evidence the improvement of the method using this software.

Creating a script in Matlab, permits to run in some seconds the desired number of simulations varying, for example one parameter and plot all the responses in one single figure. The system modeled with Simulink, based on the linear system explained in the Simulations chapter, is the following:

![Linear model with Simulink](image)

Figure 5.7: Linear model with Simulink

Where, the two out signals are used to capture the desired signals to study. The blocks inside the loop drawn, model the load torque using the feedback of the speed, the gain is equal $3 \lambda$ in order to
translate the q-current to the electric torque.

As explained in the diagram 5.6, the first step used was to increase the Kp maintaining Ki and Kd in zero.

![Figure 5.8: Kp tuning maintaining Ki, Kd equal zero. Wm response](image)

As it can be seen in 5.8 the effect of increasing the Kp derives in a shorter rise time and a smaller error in the steady state. The result for Kp = 250 is very exact to the desired speed that cannot be distinguished from it in the figure. But if the focus is on the steady state is clear that the Kp of 250 is the best one because the steady state error is smaller.

![Figure 5.9: Zoom of Kp tuning maintaining Ki, Kd equal zero. Wm response](image)

Always, trying to relate the information of the speed response with the torque response it can be said that the higher the Kp, the higher the slope and also the more constant, this will affect the torque causing that high Kp’s present constant electric torques from the very beginning (because the torque depends directly on the speed derivative). This suppose is proofed in 5.10.
As it can be seen in the figure, the Kp = 250 achieves to sink to the load torque value (which is the value it should arrive in the steady state) in a shorter time, compared to the others and the response does not oscillate very much as with other Kp’s. Maybe the Kp 10 could be seen as a good option, too. But remembering that both responses specifications should be taken into account, the Kp 250 is finally chosen even having a higher overshoot around 8s compared to Kp = 10. The fact that the rise time for the speed is shorter, the steady state error is smaller, the initial torque is more constant and the falling time after the torque overshoot is shorter, the Kp = 250 is fixed. Taking a brief look at 5.9, it is clear that the steady state error is small but not zero and the response oscillates. To eliminate this oscillation the Ki is introduced in the controller. Figure 5.11 shows that the speed response for the different Ki’s is in terms of rise time and error very similar, a zoom is needed to decide what value of Ki is the best for the speed.

In the steady state, the Ki = 15 offers a closer response to the desired speed. If the zoom would be done some seconds later, the error would become very close to zero.
Figure 5.12: Zoom of Ki tuning maintaining Kp = 250 and Kd equal zero. Wm response

What impact has the presence of the Ki in the torque response?

Figure 5.13: Ki tuning maintaining Kp = 250 and Kd equal zero. Torque response

In 5.13 the different waveforms cannot be recognized, but the figure is included in order to observe that the mere presence of a Ki implies a ripple in the torque response, which was smoother only with the Kp. But this problem will be solved including the derivative in the design. Nonetheless at that point a decision for choosing a Ki should be made. In 5.14 it can be seen that the Ki = 15 offers the lowest overshoot.
Having fixed the Kp and Ki, and being the following specifications already validated: speed rise time, steady state error and torque overshoot, the Kd will be used to eliminate the undesired ripple in the torque.

Figure 5.14: Zoom of Ki tuning maintaining Kp = 250 and Kd equal zero. Torque response

Again, the speed response evidences that the mentioned specifications are met.

Figure 5.15: Kd tuning maintaining Kp = 250 and Ki = 15. Speed response
Figure 5.16: Zoom of Kd tuning maintaining Kp = 250 and Ki = 15. Speed response

In 5.16 the abrupt change around the 8s can be seen for the speed response of the system without Kd and the effect of the derivative eliminating this abrupt changes is evidenced. For this reason, the Kd = 0 is excluded from the selection.

Figure 5.17: Zoom of Kd tuning maintaining Kp = 250 and Ki = 15. Speed response

Figure 5.17 shows that Kd=3 and Kd=30 seem valid responses, the torque figure 5.18 will decide. From this point of view it can only be said that there are very different responses depending on the Kd, in some cases the oscillation remains, in others not. To choose the right one, the following zooms are helpful.

5.2 Results
As it could be imagined looking at the speed response, the abrupt change of slope in the speed for Kd = 0 provokes a bigger overshoot in the torque. Kd = 3 appears in 5.19 as a possible option.

In 5.20 the decision is done: with a Kd = 3 there is less overshoot and as this figure shows, there is no ripple in the steady state for the torque.
As a summary, it can be said that the different steps of the diagram 5.6 where followed during the tuning and it was easy and fast to come to a solution that eliminates the error in steady state, presents an optimal rise time and the overshoot and ripple of the torque are controlled. Having calculated these values, the next step, that will be carried out in the following subsection, is the export of the results obtained with Simulink to the Questa environment.

5.2.2 Usual situation nr1: 0-100km/h

The first situation simulated corresponds to the, for cars lovers well known, 0-100 km/h characteristic. Since its early days the power of a car has been usually measured with the time it takes for it to achieve the 100 km/h speed.

Just to have some background: in 5.7 is a list of these times for different vehicles:

<table>
<thead>
<tr>
<th>Car</th>
<th>time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tesla Roadster</td>
<td>3.9s</td>
</tr>
<tr>
<td>Ferrari 360 modena 400hp</td>
<td>5s</td>
</tr>
<tr>
<td>Mercedes SLK 55 AMG 360 hp</td>
<td>5.1s</td>
</tr>
<tr>
<td>BMW Z3 Coupe M 325 hp</td>
<td>5.2s</td>
</tr>
<tr>
<td>Subaru Impreza 22B 280 hp</td>
<td>5.3s</td>
</tr>
<tr>
<td>Audi Quattro Sport 306 hp</td>
<td>5.6s</td>
</tr>
<tr>
<td>Porsche Booster S 3.2 252 hp</td>
<td>6.1s</td>
</tr>
<tr>
<td>Seat Leon Cupra R 225 hp</td>
<td>7.0s</td>
</tr>
<tr>
<td>VW Golf V GTI 200 hp</td>
<td>7.1s</td>
</tr>
<tr>
<td>Toyota Corolla TS 192 hp</td>
<td>7.5s</td>
</tr>
<tr>
<td>Opel Astra Cabriolet 2L T 190 hp</td>
<td>7.7s</td>
</tr>
<tr>
<td>Mitsubishi Colt CZT 150 hp</td>
<td>7.8s</td>
</tr>
<tr>
<td>Audi A4 2.5 Tdi Quattro 180 hp</td>
<td>8.0s</td>
</tr>
<tr>
<td>Ford Focus ST 170 173 hp</td>
<td>8.4s</td>
</tr>
<tr>
<td>Opel ampera electric</td>
<td>9.0s</td>
</tr>
<tr>
<td>Peugeot 206 Gti 1,6 115 hp</td>
<td>9.1s</td>
</tr>
<tr>
<td>Mitsubishi i-Miev</td>
<td>10.0s</td>
</tr>
<tr>
<td>Peugeot iOn</td>
<td>15.9s</td>
</tr>
<tr>
<td>Citroen C-Zero electric</td>
<td>15.9s</td>
</tr>
</tbody>
</table>
The time to reach the 100km/h varies from 5s to 16s more or less depending on the type of car, if it is a sports car or a family car or, what is more interesting in this thesis: an electric car.

In the list can be seen that the electric cars are around an acceleration time of 10s (i-Mitsuishi is 10, Opel Ampere 9, Peugeot iOn 16, Citroen c-Zero 16...). Only the Tesla-Roadster operating with an induction motor achieves 3.9s (3.7s the sport model).

Having these numbers in mind, it seems rational to demand to the electric car modeled in this thesis to reach the desired speed in 8s.

This is what will be proofed in this simulation. Since daily traveling of short distances of highways is becoming more and more common, due to the already commented problematic of suburbanization, the situation modeled will try to analyze the acceleration from 0 to 100km/h, typical driving situation after a toll post (very abundant in Barcelona and surroundings), see 5.21.

![Figure 5.21: Situation: acceleration 0-100km/h](image)

The ref arrow in 5.21 symbolizes when the speed command caused by the acceleration pedal arrives. In 8s the command speed varies from 0 to 100km/h and the target of the control is to achieve the desired speed in that time without overshoot and having a steady state error close to 0.

Having a brief look at the equations explained in the Control section:

\[
F_{\text{rolling}} = f \cdot M \cdot g
\]

\[
F_{\text{grade}} = M \cdot g \cdot \sin(\alpha)
\]

\[
F_{\text{aerod}} = \frac{1}{2} \cdot C_x \cdot S \cdot l_s \cdot v^2
\]

\[
\tau_{\text{load}} = R_{\text{wheels}} \cdot (F_{\text{rolling}} + F_{\text{grade}} + F_{\text{aerod}})
\]

It is easy to understand that the torque to overcome in this simulation depends only on the friction due to the weight of the vehicle and on the aerodynamics, there is no slope to climb in this experience and for this reason Fgrade does not contribute.

For this reason, the load torque demanded will start only with a value caused by the friction, not the aerodynamics, because the initial velocity is zero:

\[
\tau_{\text{load}} = (m \cdot g \cdot f) \cdot R_{\text{wheels}} / r_m = 800 \cdot 9.81 \cdot 0.025 \cdot 800 \cdot 0.26 / 3 = 17.004Nm
\]
and as the speed varies the load torque varies, too, adding to the initial torque (due to the mass) the aerodynamic component, due to the opposition of the fluid, in this case air, that is displaced, that makes the final torque, when the speed is constant and equal to 100km/h:

\[
\tau_{load} = 17.004N \cdot m + \frac{R_{wheels}}{r_m} \cdot \frac{1}{2} \cdot C_x \cdot S \cdot l_s \cdot v^2 = 17.004 + \frac{0.26}{3} \cdot 0.5 \cdot 0.55 \cdot 1.8 \cdot 1.2 \cdot \left(\frac{100}{3.6}\right)^2 = 56.73N \cdot m
\]

Results with a frequency of 3kHz

Using the parameters of the controller obtained in the previous section (Kp = 250, Ki = 15, Kd = 3) the results obtained using a switching frequency of 3kHz for the electric torque, motor’s rotation speed and the controlling currents are shown in 5.22

![Figure 5.22: Basic waveforms for 3kHz](image)

As it can be seen, the speed response is really closed to the desired speed reaching at 14s the speed 320.64 rad/s. The maximum speed error arrives shortly after the 8s (8.32s), it is the overshoot, and it is only 1.72 rad/s above the desired speed (around 0.5 % error).

The electric torque has an average value during the first 6 seconds of 163.57 Nm, an acceptable value for an electric car. As it can be seen in the graph, when the speed changes to the constant value at time 8s, the electric torque experiences a maximum of 278.74Nm. After the maximum, the torque decreases abruptly to lower values around the load torque value, which is logical in the steady state due to the equation that relates these two magnitudes.

The ac currents feeding the motor, that are result of all the control will be analyzed in detail later. It should be underlined the sudden change of all this waveforms at 8s, when the speed becomes constant.

About the control currents it can be said that the q-current has obviously the same form as the electric torque (remember equation \(\tau-i_q\)) and about id it is important to say that in the steady state it
will achieve the zero value desired. In 5.23 a zoom of $i_d$ and the electric torque at the last seconds of simulation is shown. If the simulation would go on, in some seconds more, the clear trend shown in 5.23 would finish with $i_d$ being zero and the electric torque exactly the same Nm as the load torque.

![Figure 5.23: Zoom of $i_d$ and $\tau_e$ for 3kHz](image)

In 5.24 the voltages created by the inverter owing to space vector modulation control can be seen. They are really similar to the typical triphasic voltages and as it can be seen the amplitude of the pulses try to emulate this characteristic.
The first pulse after 8s has been taken and the misalignment between the three voltages was measured obtaining: 0, 57.50 and 122.13 degrees.

The stator currents obtained with these voltages present the typical transitory waveforms that can be found in every paper simulating PMSM (5.25)

In the first seconds of simulation these currents present an amplitude of 300A. Again a peak current appears around the 8s (7.995s) of 490.5 A and the steady state value of the amplitude is around 150 A

5.2 Results
as it can be seen in 5.26

![Stator currents in the steady state](image)

**Figure 5.26: Stator currents in the steady state**

The last results that will be presented in this test will be related with the power of the motor and its efficiency. From [15] is known that the power balance equation is:

\[
P_{electric} = \frac{3}{2} \cdot (v_d \cdot i_d + v_q \cdot i_q)
\]

\[
P_{losses} = \frac{3}{2} \cdot r_s (i_d^2 + i_q^2)
\]

\[
P_{magnetic} = \frac{3}{2} (L_d \cdot \frac{d}{dt} i_d^2 + L_q \cdot \frac{d}{dt} i_q^2)
\]

\[
P_{mechanic} = \tau_e \cdot w_m
\]

\[
P_{electric} = P_{losses} + P_{magnetic} + P_{mechanic}
\]

Where \(P_{electric}\) symbolizes the electric power supplied by the battery, \(P_{losses}\) is the power losses due to the calorific power developed in the resistances of the stator, \(P_{magnetic}\) is due to the variation of the stored energy and the rest \(P_{mechanic}\) is the mechanical power developed by the motor. The magnetic losses due to the simplifications done in the model are neglected.
If the focus is on the transitory, the losses in the resistances the torque and currents, and therefore the losses in the resistances are bigger than in the steady state. That can be seen in 5.27, where the average power of the supply, motor and resistor losses are indicated. The values of the transitory are in average 28.091 kW for the supply power, 4.5244 kW for the resistance losses and 23.568 kW for the mechanical power obtained.

The efficiency in converting the electrical energy in mechanical energy can be calculated as:

$$\eta = \frac{P_{mechanic}}{P_{electric}} = \frac{23.568}{28.091} \cdot 100 = 83.9\%$$

Nevertheless if the analysis is done only in the steady state the results are completely different, like shown in 5.28. With this new values, which avoid the instabilities of the transitory the results are really improved:

$$\eta = \frac{P_{mechanic}}{P_{electric}} = \frac{17.969}{18.798} \cdot 100 = 95.58\%$$

5.2 Results
Results with a frequency of 6kHz

The same simulation was done, with the same PID parameters but varying the switching frequency to 6kHz. In 5.29 can be seen clearly that the ripple of torque and currents has decreased. Around the 8s there are not strange peaks like it happened before.

The steady state results are very similar, iq trends to zero and electric torque to the load torque, too.
The voltages achieved with the field oriented control present compared to the results of 3kHz the double of widths, because the algorithm calculates with every new sample of the rotor position, and the more sample arrive per time unit, the more pulses it generates.

The stator currents present in the transitory the same amplitude of 300A, but in comparison with the results of 3kHz the peaks become smoother and more regular in a short time.
In the steady state, the currents present in the case of 6kHz an amplitude of 120A, less than the currents of 3kHz, and the waveforms are really smoother, without undesired peaks that affect the performance of the torque.

Just to finish the comparison the power measures obtained with the high frequency are presented.
Figure 5.34: Supplied and demanded power

Again, due to the elimination of the strange peaks around the 8s, the new powers present smoother waveforms without strange values as in 3 kHz, the balance of powers to obtain the efficiency which is in this case just a little higher than the one of 3kHz because using average values, the undesired peaks do not affect very much (due to the fact that they are punctual):

$$\eta = \frac{P_{\text{mechanic}}}{P_{\text{electric}}} = \frac{23.484}{27.941} \cdot 100 = 84.05\%$$

And the same in the steady state:

$$\eta = \frac{P_{\text{mechanic}}}{P_{\text{electric}}} = \frac{17.963}{18.734} \cdot 100 = 95.88\%$$
Figure 5.35: Supplied and demanded power in the steady state

Just to finish this results, say that the supplied power has this abrupt transitions because it was calculated using $v_d$ and $v_q$ that depend on the voltages supply by the inverter that also have the abrupt transitions due to the switching. In fig 5.36 it can be seen the supplied power around the 12 seconds.

Figure 5.36: Supplied power zoom

All in all, in view of the results it can be said that the test of 0-100km/h proofed that the response follows the command properly. The balance powers show that the conversion of energy is done efficiently and the fact that the currents obey the desired values implies that the control module was built successfully.
Dependence on frequency

To show the differences in the response depending on the frequency of switching, another simulation working at 12 kHz was realized and the real time it ran and the ripple of the torque obtained were compared to the other two simulations already described. As comments to the table should be added that the ripple is compared close to the stationary state (around 14s) and that the simulation times can vary depending on the use of the CPU were Questa is run. This results can be found in 5.8.

The comparison of the frequency dependent result evidenced that the higher the frequency, which implies more time of computing due to the higher number of operations to be realized, derives in a smoother response for currents and torque, fact that is crucial for the proper behavior of the motor. Just as a brief remark for further works using this model to simulate the PMSM controls: If the frequency used is too low like it can be seen in the first row of table 5.8 (what normally is desired during simulations in order to obtain the results faster and to run the program more easily) the peaks of torque, and therefore current become so high, that the simulation is aborted due to, as the message indicates: "a negative wait statement". This is due to the svm.vhdl and sequence.vhdl blocks that, remember, build the switching scheme according to the times calculated with vd, vq and theta. When the torque peaks becomes very high, the iq and id follows and vd and vq, which are also related to id and iq (maybe not very direct, but in any case related) become uncontrollably higher and the equation \( T_0 = T_s - \frac{I_s}{V_d}(v_{max} - v_{min}) \) assigns negative values to the parameter T0 which appears in the wait statement of sequence.

This error is a particularity of this thesis, because the VHDL AMS implementation using the wait statements for the switching times was developed here in a novel way. It is not a negative aspect, that this error appears sometimes, as said, is due to a bad switching frequency selection, if the error did not appear, the results obtained would not be valid due to the big torque ripple nevertheless.

<table>
<thead>
<tr>
<th>Simulation nr</th>
<th>Frequency (kHz)</th>
<th>Ripple between 14:00 and 14:50 (Nm)</th>
<th>Simulation time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>error</td>
<td>error</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>84.996</td>
<td>1h 7min</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>48.095</td>
<td>2h 30min</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>15.42</td>
<td>4h 47min</td>
</tr>
</tbody>
</table>

Table 5.8: Comparison of results depending on frequency

All in all, it can be concluded that the frequency is a decisive parameter in order to obtain preciser response, the only prize to pay is the simulation time. For this reason, as said in the PID tuning subsection, it’s important to develop a method for preparing the last simulations (which should be run at high speeds) with a faster technique (like linearization + Simulink).

5.2.3 Usual situation nr2: Parking ramp

The next situation analyzed is due to the abundant number of private subterranean parking situated in big cities. It is an important situation to study because when taking the car out from the garage, the load torque is one of the biggest demanded. Here are again the equations presented before:

\[
F_{rolling} = f \cdot M \cdot g \\
F_{grade} = M \cdot g \cdot \sin(\alpha) \\
F_{aerod} = \frac{1}{2} \cdot Cx \cdot S \cdot ls \cdot v^2
\]
\[ \tau_{\text{load}} = R_{\text{wheels}} \cdot (F_{\text{rolling}} + F_{\text{grade}} + F_{\text{aerod}}) \]

The slope used is of 15 degrees and the motor speed should be around 18 km/h in order to overcome the 4m height in more or less:

\[
t = \frac{1 \text{h}}{18 \text{km}} \cdot \frac{1 \text{km}}{1000 \text{m}} \cdot \frac{4 \text{m}}{\sin(15)} \cdot \frac{3600 \text{s}}{1 \text{h}} = 3.09 \text{s}
\]

Taking a look at 5.37 the conditions for the test can be understood step by step depending on the position numbered.

Before 1): the car is in standstill

1) The command speed arrives (ref1), the accelerator is pushed till the end and the translated signal grows from 0 to 18 km/h (57.69 rad/s) in 2s. The initial torque is only due to the friction, that means:

\[
\tau_{\text{loadmotor}} = \frac{R_{\text{wheels}} \cdot F_{\text{rolling}}}{r_{\text{m}}} = \frac{R_{\text{wheels}} \cdot m \cdot g \cdot f_{\text{rm}}}{r_{\text{m}}} = \frac{0.26 \cdot 0.025 \cdot 9.81 \cdot 800}{3} = 17 \text{Nm}
\]

And while it accelerates the aerodynamic force produces a small torque too due to the low speed.

2) The torque grows abruptly, the slope is suddenly there and it represents a torque increase of:

\[
\tau_{\text{grade}} = \frac{R_{\text{wheels}} \cdot (F_{\text{grade}})}{r_{\text{m}}} = \frac{0.26 \cdot (\sin(15) \cdot 9.81 \cdot 800)}{3} = 176.04 \text{Nm}
\] (5.3)

3) The command speed that translates the pedal (ref2), emulates the situation of release a little the pedal in order to get out of the ramp with a lower speed (around 5 km/h = 16.03 rad/s) due to the presence of a crossing street or pedestrian crossing after the ramp. This signal operates during 1s.

4) The torque changes, the slope is over. The only presence of the initial torque of 17 Nm of the friction and the small one due to the aerodynamics are restored.

![Figure 5.37: Situation: parking ramp](image)

The parameters found in the 100 km/h simulation test were reused in order to see the need or lack for recalculating the controller parameters. Here are the results:
Now the challenge consists on overcoming the sudden change of torque which moreover increases largely. In order to get an accurate torque response and better dynamics in general, the high frequency of 6kHz was chosen.

In figure 5.38 the smooth characteristics of this frequency can be seen. Neither in the torque nor in the currents are strange peaks to see like it happened in the 100km/h test using a frequency of 3kHz. As it can be seen the torque response is acceptable, while climbing the slope the electric torque is bigger than the load torque in order to have a positive derivative in the motor speed. In the two constant regions, the electric torque tracks the load torque as it is logical when the derivative of speed is zero and while the car is decreasing the speed, the electric torque supplied is under the load torque. The maximum value achieved in the electric torque is 214.54 Nm.

If the focus is on id, it will be clear that the id tries to reach the zero level during the constant speed climbing of the slope, but does not really achieve it, later when the constant speed of 5km/h is established; the id becomes zero validating the goal of the FOC control theory.

In the previous figure it could not be seen, but if we take a look at the zoom of the speed response (5.39), it will be clear that the response is not as perfect as in the case of the 0-100 km/h. More than in the PID values, the reason for this fact should be searched in the heavy torque applied abruptly; none the less the response is acceptable (the maximal relative error is 2.8% at the second 2.15). The difficulty to overcome the load with the desired speed was the reason for adding 1s (in total 4s) to the time calculated for climbing the ramp.
The form of the waveforms of supply voltages depend strongly on the behavior of speed and the same in the inverse way. Compared to the 0-100km/h case, in 5.40 can be seen that due to the alternating character of the parking ramp test, the form of the voltage waveforms change constantly, that means that the voltage supply detects what are the needs of the systems and consequently applies the required controls.

The initial current in the stator presents a peak of 230A (figure 5.41), but due to the high load torque the peak in the constant speed of 18km/h is 380A (figure 5.42).
Finally, in the steady state of 5 km/h the amplitude of the current is again low (45A) because the demanded torque becomes really small in comparison to the torque in the other cases (only friction and aerodynamics for a very low speed are present), see 5.43.
A power balance was realized in order to calculate again the efficiencies of the system. In this case, the different stages of the simulation require their own calculation of the power balance in order to understand what is happening.

In figure 5.44 during the first seconds, where the speed grows practically linearly and the electric torque is constant and bigger than the load torque, the average electric power generated is 7.103 kW, and the peaks are around 80kW. Compared to the acceleration from 0 to 100km/h, the supplied power
is lower in this case (peaks of 80kW compared to 140kW). The power losses in the resistances (3.78kW) and the mechanical power (3.3kW) evidence that in this first "transitory" the efficiency is low:

\[
\eta = \frac{P_{mechanic}}{P_{electric}} = \frac{3.2966}{7.1030} \cdot 100 = 46.41\%
\]

In the second section of the figure the constant speed and the heavy load torque dominate. The big torque derives in higher mechanical power but also in higher currents and more losses in the resistances, the efficiency is now:

\[
\eta = \frac{P_{mechanic}}{P_{electric}} = \frac{10.965}{19.86} \cdot 100 = 55.21\%
\]

During the speed decrease, the torque and the q-current are reduced abruptly, and therefore all the powers decrease too:

\[
\eta = \frac{P_{mechanic}}{P_{electric}} = \frac{2.0819}{3.5714} \cdot 100 = 58.29\%
\]

And in the last part:

\[
\eta = \frac{P_{mechanic}}{P_{electric}} = \frac{273.66}{719.92} \cdot 100 = 38.01\%
\]

This last value seems negative, but was calculated like this with regard, to show the importance in all the previous tests of having chosen the real steady state to make the balances. In this case, the region took is part of the transitory, as it can be seen the currents and voltages that produce the supplied power are still adapting their values to the new conditions (values of Pelectric in second 6 are really higher than in second 8) while the electric torque (and the speed, and therefore the mechanical power, too) assumes the final value from the very beginning (time 6s). If we had used the last second of simulation (from 8s to 9s) where the steady state is closer, the result (of the powers and their efficiency which are not include here) would have been 73 %.

Just to conclude with this test, it should be underlined that the low efficiencies obtained in this test are due to the extreme changes during the drive, nevertheless the efficiency of the electric motor is still higher than the one developed by a combustion engine, and after the steady state is reached, the comparison is again like in the first test: the efficiency of the electric motor is around 90 % and it represents a clear advantage. The fact that the dynamic response with such changes is a little slow and later it works perfectly is to be found in the FOC disadvantages compared to DTC, as explained in the first chapters FOC priorizes steady state response over dynamic transitories, exactly the opposites is done in DTC. Nevertheless, the smooth response obtained with FOC and the high efficiencies in the steady state still make it a good option.

5.2.4 Usual situation nr3: New European Driving Cycle

The third and last simulation is related with the New European Driving Cycle which is originally a test designed for evaluating objectively the environmental impact of the cars. The information has been taken from [27]

Even if treating with electric cars the environmental impact is not the question, this driving cycle was chosen because of its precise description of the usual situations to find while driving a car in the city and its surroundings. It consists of four identical driving cycles for the inner city circuit (ECE-15) and one extra-urban cycle (EUDC: Extra-Urban driving cycle)
The first cycle (which is repeated four times) does not exceed the urban limit speed of 50km/h and represents reliably a possible driving cycle in the city. Its characteristics are exactly described in table 5.9. Only this part of the cycle was simulated.

<table>
<thead>
<tr>
<th>Operation</th>
<th>duration(s)</th>
<th>time(s)</th>
<th>V- (km/h)</th>
<th>V+ (km/h)</th>
<th>Distance (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>acceleration</td>
<td>11</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0.00</td>
</tr>
<tr>
<td>constant speed</td>
<td>8</td>
<td>23</td>
<td>15</td>
<td>15</td>
<td>41.67</td>
</tr>
<tr>
<td>slowing down</td>
<td>5</td>
<td>28</td>
<td>15</td>
<td>0</td>
<td>52.08</td>
</tr>
<tr>
<td>stop</td>
<td>21</td>
<td>49</td>
<td>0</td>
<td>0</td>
<td>52.08</td>
</tr>
<tr>
<td>acceleration</td>
<td>12</td>
<td>61</td>
<td>0</td>
<td>32</td>
<td>105.42</td>
</tr>
<tr>
<td>constant speed</td>
<td>24</td>
<td>85</td>
<td>32</td>
<td>32</td>
<td>318.75</td>
</tr>
<tr>
<td>slowing down</td>
<td>11</td>
<td>96</td>
<td>32</td>
<td>0</td>
<td>367.64</td>
</tr>
<tr>
<td>stop</td>
<td>21</td>
<td>117</td>
<td>0</td>
<td>0</td>
<td>367.64</td>
</tr>
<tr>
<td>acceleration</td>
<td>26</td>
<td>143</td>
<td>0</td>
<td>50</td>
<td>548.19</td>
</tr>
<tr>
<td>constant speed</td>
<td>12</td>
<td>155</td>
<td>50</td>
<td>50</td>
<td>714.86</td>
</tr>
<tr>
<td>slowing down</td>
<td>8</td>
<td>163</td>
<td>50</td>
<td>35</td>
<td>809.31</td>
</tr>
<tr>
<td>constant speed</td>
<td>13</td>
<td>176</td>
<td>35</td>
<td>35</td>
<td>935.69</td>
</tr>
<tr>
<td>slowing down</td>
<td>12</td>
<td>188</td>
<td>35</td>
<td>0</td>
<td>994.03</td>
</tr>
<tr>
<td>stop</td>
<td>7</td>
<td>195</td>
<td>0</td>
<td>0</td>
<td>994.03</td>
</tr>
</tbody>
</table>

Table 5.9: Different test stages

In the inner city cycle which is repeated four times, the average speed is 18.35km/h, the total time 13 min, the distance driven 3,98 km and the top speed 50 km/h. In the next figure, 5.45, the results for the same Kp, Kd and Ki found for the first test are presented in the next lines.

The aim of this test is not analyze every single detail like already done in the previous simulations, the idea behind it is to proof that the response of the system for a long simulation with different conditions changes still works properly. For this reason the innercity cycle of the mentioned european test was chosen. The selected frequency in this case was low due to the huge amount of time for processing a simulation of 180 seconds. For this reason, the results presented are only qualitative.

Nevertheless, the speed response is smooth enough. As it can be seen in 5.45 the currents and torque response presents overshoots that are undesired and while the speed remains constant id is zero or presents an average zero if it is not completely constant, while iq always presents a value depending on the load torque.
Important is to conclude that the speed response tracks the desired one and that the validation of a long simulation with a higher frequency could be done in this subsection. With more powerful computers and less time, the response would be more accurate, but the information acquired was enough.

5.3 Summary

In this section, the response of the system modeled was validated with the simulations of three different cases. In the first one different main characteristics of the motor such as stator currents, torque, speed and power balances were presented, comparing the results using different switching frequency. It was observed that, the higher the switching frequency, the smaller the ripple and smoother the response. In the second test, an extreme situation (sudden high torque) was examined evidencing that the dynamic result in this case is not very exact, but is acceptable and once the steady state is reached the response is again exact presenting high efficiency. This characteristics were exactly expected since the very beginning when the FOC controls were chosen instead of DTC.
The third test was just a last effort to evaluate the correctness obtained in the previous short times simulations. For this reason, the fact that the motor speed tracks perfectly the desired one during all the cycle finished with the conclusion that the model obtained works properly.
Kp = 0, Kd = 0, Ki = 0
INITIAL
Kp
Rise time 8s (0-98% reference)
“ss” Error < 0.3 %
Fall time < 1s
No Oscillation?

Kd
Overshoot < 30%
No Ripple?

Ki
“ss” Error < 0.01%

no
yes
Condition referred to speed response
Condition referred to torque response

yes
no
yes
no
yes
no
yes
no
yes

Kp = 150, Kd = 3, Ki = 15
END

Figure 5.6: PID tuning proceed
6 Conclusions

The vast challenge of modeling and simulating the controls of a PMSM resulted in some useful conclusions fruit of the experience at different levels. All this ideas, which will be presented separately according to their nature, can inspire the reader convinced of going one step further and implement the controls described in this thesis.

For this reason, the last section of the thesis will first summarize the important points to have in mind, observed during the realization of this work, when modeling and simulating the system and will finish with some advices for the way forward to synthetize the controller.

6.0.1 Conclusions

Modeling

The first problem that any person interested in simulating the PMSM will have to face is to find the right equations to describe the motor. In this thesis, it was proved that the d-q equations are the appropriate to be used, but furthermore it was experienced the need of separating the motor in three modules, one of them working only for the integration speed-rotation angle. This is the first recommendation, to use the equations of the motor described in the control section and to divide them the same way, using an integrator instead of derivatives for the speed-angle.

In the same line goes the selection of the control. The space vector modulation selected, which avoids burden calculations such as arctan, offered good results and was simple to implement. Another special aspect to remark is the importance of being skilled in the mechanic concepts of torque and transmission in order to handle constantly with torques and speeds related to the motor or the wheel reference. As described before, this thesis can be used as a good reference for setting these conditions.

Maybe the most important point of view is the tuning of the PID, but as it will be explained in further works, having the complete scheme with encoder, sensors and purely digital blocks, there are some papers on the net offering methods to calculate the PID. Nevertheless the particular solution used for this model and simulation, as it will be explained after these lines, solved the problem of the control.

Simulating

In this step the main challenge is represented by the PID control. For the reader interested in going further with this thesis and implement the controller, maybe the solution developed here won’t be very useful, only the different responses obtained will be of high interest to have some notions about how the motor should react in front of different situations such as the proposed in the cases 1, 2 and 3. The advice for those persons is discussed in the further works section.

If the aim is only to model and simulate the motor and its controls with VHDL AMS, the method used in this thesis is strongly recommended. First, the simulations were carried out always with Questa,
process which took incredible amounts of time using the environment described in the simulations chapter introduction. But afterwards the idea of linearizing the model and simulating this approach with Simulink reduced the time of simulation incredibly.

As fruit of the experience, for those who want to implement this design with VHDL AMS, is strongly recommend to approach the right solution using the simplified system and Simulink tool, afterwards having fenced the solution parameter, Questa can be used to simulate all the mixed system and finish to calibrate the valid response.

Using the VHDL AMS code developed in this thesis which is particular and represents a different way to model the system it is recommended to work with a high switching frequency (above 6kHz) in order to get the desired smooth response and to not meet the fatidic error that stops the simulation (already explained before, due to a negative wait statement).

To finish with the conclusions just underline the importance of the different advices described in this section, which will really suppose high time savings and will facilitate extremely the introduction in further works.

6.0.2 Recommendations for Further Work

As already mentioned during the rest of the thesis, the complexity and vast challenge to model, simulate and synthetize the controls of the PMSM were considered and the decision of focusing on the modeling and simulations was made opening the door to further works.

The reader interested in synthetizing the controls based on this thesis should find some help in the following points (To refresh the figures of the control section will help to understand):

- The first step is to include the current sensors and the encoder. The signal received from the encoder should be processed in order to estimate the rotation speed which is the feedback for the main loop. The introduction of the sensors require a short analysis of the sample times to use, in order to synchronize the inputs of the loops.

- Having the sensors modeled, the output of them will be a digital word that will arrive every x ns (depending on the sample times). In the case of the currents, the signal read will be the corresponding to ia and ib. This two currents have to be transformed to d-q axis in order to feedback their respective loops. This transformation requires trigonometric functions such as sinus and cosinus and to implement it, cordic algorithms based on floating point are necessary. In [28], also developed in the IES can be found the code to implement this transformation.

- The values read from the sensors will enter in the substractors and will be compared with the desired values. Since the sensors signals are digital words, the substractor.vhdl, PID.vhdl, SVM.vhdl and sequence.vhdl should be modified in order to work with bit words instead of real quantities.

- The voltage of the battery which is used not only in the driver to feed the motor, but also in the svm modul to calculate the switching times, should be read and converted to a digital word in order to process this information in the control unit designed.

- And last but not least, the PID control should be reconsidered due to the possible instabilities brought by the sensors and the trigonometric transformations. While this thesis was focused on simulation and modeling, the PID half-manual tuning was acceptable; now, the mentioned instabilities and an optimum
response require a deeper analysis. In this direction it should be interesting to take a look at [29], [30], [21]. All of them discuss the issue of computational method for tuning the PID such as swarm particle optimization and neural networks.

Another interesting branch to study could be "decoupling" which would help to the control performing transform from non-linear system to a linear one easy to control using advanced linear control theory. In the paper [31] the reader will find useful information for this purpose.

Finally, having already a useful model to be simulated using the VHDL AMS code developed in this thesis, different tests could be done changing some parameters in order to get to learn more about the system, for example changing the supply voltage or using another zero vectors scheme in the sequence modul instead of the right aligned scheme.

The complexity and extension of the PMSM control requires joint efforts of the scientific community and this thesis based on the theory presented in very different papers developed a model validating the results with simulations and it pretends to have contributed in this joint efforts doing its bit: providing a good reference for stepping into the synthesis of the controller which is the final goal of the engineer, to see the engine controlled.
Bibliography


