ELECTRIC MOTOR CONTROL SYSTEM WITH APPLICATION TO MARINE PROPULSION

by

Camilo Carlos Roa

A Thesis Submitted to the Faculty of

The College of Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degree of

Master of Science

Florida Atlantic University

Boca Raton, Florida

August 2010

ELECTRIC MOTOR CONTROL SYSTEM WITH APPLICATION TO MARINE PROPULSION

by

Camilo Carlos Roa

This thesis was prepared under the direction of the candidate's thesis advisor, Dr. Nikolaos Xiros, Department of Ocean and Mechanical Engineering, and has been approved by the members of his supervisory committee. It was submitted to the faculty of the College of Engineering and Computer Science and was accepted in partial fulfillment of the requirements for the degree of Master of Science.

SUPERVISORY COMMITTEE:

Nikolaos Xiros, Ph.D.
Thesis Advisor

Manhar Dhanak, Ph.D.

Pierre-Philippe Beaujean, Ph.D.

Mohammad Ilyas, Ph.D.
Chair, Department of Ocean and Mechanical Engineering

Karl K. Stevens, Ph.D., P.E.
Dean, College of Engineering and Computer Science

Barry T. Rosson, Ph.D.
Dean, Graduate College

July 20, 2010
Date

ACKNOWLEDGEMENTS

The author wishes to express his gratitude to his family for their unconditional support; to his friends for their time and warmth; to his professors for their dedication and patience; to the FAU staff for their impeccable wok; to the people of the Center for Ocean Energy Technology for their support and help with the physical part of this project; to his advisory committee for their guidance; to his advisor, Dr. Nikolaos Xiros, for his clarity and faith in him; and to his girl for being the one single light in the sky that never blinked.

ABSTRACT

Author: Camilo Carlos Roa

Title: Electric Motor Control System with Application to Marine Propulsion

Institution: Florida Atlantic University

Thesis Advisor: Dr. Nikolaos Xiros

Degree: Master of Science

Year: 2010

This thesis analyses the behavior of an induction motor based on a mathematical model created for its simulation. The model describes the interaction of its several non linear differential equations to present a simulated output of induced torque and mechanical speed. Considering the applications to marine propulsion it is also the goal of the project to design and test a control system for the speed of the motor by maintaining a specific cruse speed regardless the perturbations.

ELECTRIC MOTOR CONTROL SYSTEM WITH APPLICATION TO MARINE PROPULSION

LIST OF TABLES

TABLE OF FIGURES

1. INTRODUCTION

A diesel engine in marine vehicles is designed to deliver the desired speed, which in most of the cases is the cruise speed, at its maximum efficiency. The main disadvantage is that once the speed is set in the design it cannot be changed and if a variable speed is required it must be achieve at expenses of the efficiency. Same thing happens if the vehicle is faced with changing weather conditions; the perturbations introduced by the environment may affect the ideal performance of the motor. One possible solution to this problem is the usage of electrical motors since they can change their characteristics more easily than diesel motors. A different input power will represent a different speed and, if properly tuned, the motor will work at maximum possible efficiency within a larger range of speeds.

This thesis focuses on the control system required to regulate the input power on an electrical motor for changing load (perturbations) to achieve a better performance. This control system is required base on the assumption that the vehicle will change its speed during regular performance.

One motivation for this work is to make electrical motors more adaptative than diesel engines; adding a control system will represent an important advantage, making them more flexible and affordable. It will also improve ship's overall

efficiency and performance costs, considering that without a control system the common solution is to over-dimension the motor.

2. PROBLEM STATEMENT

Theoretically an electric motor can change its speed by changing its input, power if it is DC or frequency if it is AC; however, small perturbations on the input may greatly affect the result (this is especially true in the case of an electric AC motor where the model used plays a big role (Zaiping, 2003)). Since the motor has a load (even if no physical load is attached, the losses of the materials may act as one) no abrupt change in the input can be made: a fast change may cause the speed to oscillate around its equilibrium point, a very slow one may take too much time. The problem, then, is how to control the input with low oscillation and small time of answer? And, how to change the input soon enough when the motor faces unpredictable perturbations? A control system is evidently required.

The main challenge will be to integrate and test a suitable control system in a model that represents real conditions for a marine vehicle. This will require for the system to be flexible to adapt itself to different perturbations and robust enough to avoid compromising the integrity of the motor. The problem will be reduced to control one single AC 3-phased electrical motor for different. Most of electrical motors used to propel ships are AC since they are more efficient than DC; for this project an asynchronous AC motor is going to be used, this is due to its lower maintenance costs compared to a synchronous AC motor, however most of the calculations can apply to both (synchronous and asynchronous).

The questions I hope to answer are:

What control system (PID, neural networks, fuzzy logic, etc. or a combination of them) is better for this problem? Why?

How can this system be used in the best way to obtain the best performance?

Under what conditions or range of parameters can this system be used?

How can the system be extrapolated for a marine vehicle?

The solution, as suggested, may be a combination of different approaches (Machat, 2005), (Denal, 2002). The goal of this work will be to find an optimal control system for different speeds, stable, fast responsive in time when changing speeds, flexible and robust compared with respect to a manually tuned motor (open loop).

3. LITERATURE REVIEW AND STATE OF ART

Here is a compilation of the related subjects, from basic motor concepts to dynamic models for analysis and simulation.

3.1. AC MOTOR

An AC motor is an electric motor that is driven by an alternating current, consisting of 2 basic parts: an outside stationary stator (with coils that, supplied with an AC current, produce a rotating magnetic field), and an inside rotor (that rotates as a result of the torque given by the magnetic field) attached to a shaft.

Two types of AC motors are known based on the rotor used: synchronous motor and asynchronous motor. A synchronous motor rotates exactly at the supply frequency (or submultiples of it). The magnetic field on the rotor is either generated by current delivered through slip rings or by a permanent magnet. An asynchronous motor is known as induction motor too; it rotates slightly slower than the supply frequency. In this case the magnetic field is created by an induced current, hence its name (Bose, 2002).

3.1.1. Rotor Speed on Asynchronous Motors

In an asynchronous motor when a 3-phase voltage is applied to the stator and a 3-phase set of current flows; due to the stator windings arranged around the rotor

a rotating filed pattern is produced. This changing magnetic field induces current in the rotor conductors, the interaction between the induced current and the magnetic field creates the rotational motion of the rotor. The speed of the magnetic field's rotation is given by

$$\boldsymbol{n_{sync}} = \frac{\boldsymbol{120\, f_e}}{\boldsymbol{P}} \qquad (3.1.1)$$

where *fe* is the system frequency in hertz and P is the number of poles in the machine. The synchronous speed is the maximum speed that an AC motor can have. In an Induction motor it is the relative motion of the rotor compared to the stator magnetic field, hence when the rotor reaches the synchronous speed the relative motion with respect to the magnetic field is zero and the induction current stops (so the torque); with no induced current and no torque the friction losses slow down the rotor until its speed falls below the synchronous speed and current (and torque) is once again induced. This is the reason why an asynchronous machine rotates slightly below the synchronous speed never reaching it (Chapman, 2005). The difference between the synchronous (theoretical speed) speed and the asynchronous speed (real speed of the motor) is called slip and is calculated on equation 3.1.2

$$\boldsymbol{s} = \frac{\boldsymbol{n_s - n_r}}{\boldsymbol{n_s}} \qquad (3.1.2)$$

Therefore the rotor speed for an asynchronous machine is give by equation 3.1.3:

$$n_r = n_s(1 - s) \qquad (3.1.3)$$

where *s* is the *slip* and n_s is the synchronous speed. Usually slip is less than 5% so for practical purposes the asynchronous speed can be approximated to the synchronous speed. Evidently a synchronous motor always runs at synchronous speed with 0% slip (Fitzgerald, 2003).

3.1.2. Speed Control of an Induction Machine

As seen in equation 3.1.1 the synchronous rotational speed of the rotor (neglecting slip, i.e. approximating the asynchronous speed to the synchronous speed) is defined by the number of pole pairs (number of windings in the stator) and by the frequency of the supply voltage (Fitzgerald, 2003).

Before the development of cheap electronics devices, the only way to vary the speed of an induction machine was rewiring its poles, this usually implied stopping the motor and opening it. With recent advances in electronics it is possible to vary the supply frequency, to virtually any value of frequency; allowing, this way, a much wider range for the rotor speed. The tendency now is to control the rotor speed by changing the frequency of the voltage input.

3.1.3. Physical Model of an Induction Motor

As explained before an induction motor relies for its operation on the induction of voltages and currents in its rotor circuit from the stator circuit (transformer action). Because the induction of voltages and currents in the rotor circuit of an induction motor is essentially a transformer operation, the equivalent circuit of an

induction motor will turn out to be very similar to the equivalent circuit of a transformer.

A transformer per-phase equivalent circuit, representing the operation of an induction motor, is shown in figure 3.1.1.

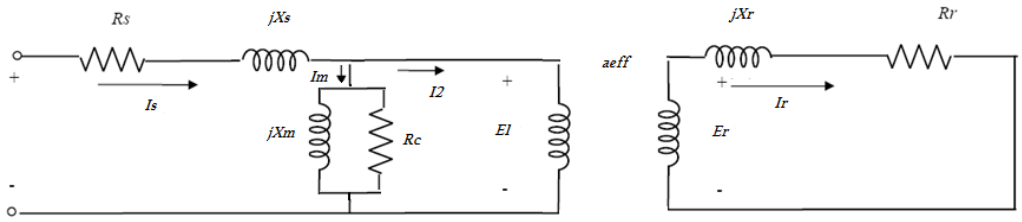

Figure 3.1.1: per-phase equivalent circuit

As in any transformer, there is a certain resistance and self-inductance in the primary (stator) windings, which must be represented in the equivalent circuit of the machine. The stator resistance will be called Rs, and the stator leakage reactance will be called Xs. These two components appear right at the input of the machine model.

Also, like any transformer with an iron core, the flux in the machine is related to the integral of the applied voltage *E1*. As a result of an air-gap between the primary and secondary windings there is a magnetizing reactance called Xm in the equivalent circuit (this reactance will have a much smaller value than in an ordinary transformer).

The primary internal stator voltage *E1* is coupled to the secondary *Er* by an ideal transformer with an effective turns ratio a_{eff}. The primary impedance and magnetization current of the induction motor are very similar to the corresponding

components in a transformer equivalent circuit. An induction motor equivalent circuit differs from a transformer equivalent circuit primarily in effects of varying rotor frequency on the rotor voltage *Er* and the rotor impedances Rr and jXr.

To produce the final equivalent per-phase circuit for an induction motor, it is necessary to refer the rotor part of the model over the stator side. The rotor circuit model that will be referred to the stator in shown in figure 3.1.2, which has all the speed variation effects concentrated in the impedance term. The rotor resistance is a constant Rr while the reactance is affected in a more complicated way. The reactance of an induction motor rotor depends on the inductance of the rotor and the frequency of the voltage and current in the rotor. Xro is the blocked-rotor rotor reactance that comes out after considering the effects of frequency (Chapman, 2005).

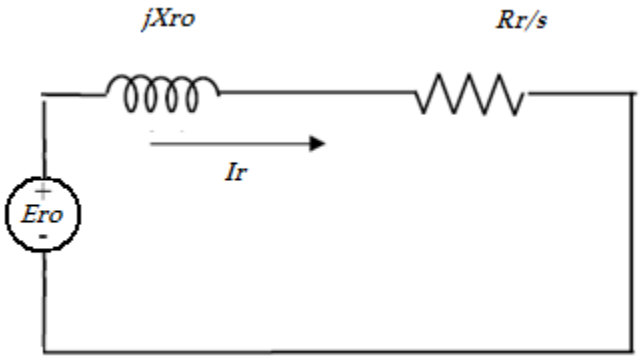

Figure 3.1.2: rotor circuit model with all frequency effects contained in resistor Rr

Considering that $f_r = sf_e$, then $X_r = sX_{ro}$. The rotor current flow can be found as

$$I_r = \frac{E_r}{R_r + j\,X_r}$$

$$I_r = \frac{E_r}{R_r + j\,sX_{ro}}$$

$$\boldsymbol{I_r = \frac{E_{ro}}{R_r/s + jX_{ro}}} \qquad (3.1.4)$$

Notice from equation 3.1.4 that it is possible to treat all of the rotor effects due to varying speed as being caused by varying impedance supplied with power from a constant-voltage source E_{ro}.

In an ordinary transformer, the voltages, currents, and impedances on the secondary side of the device can be referred to the primary side by means of the turns ratio of the transformer. The same applies for the induction motor, the final equivalent circuit can be seeing figure 3.1.3.

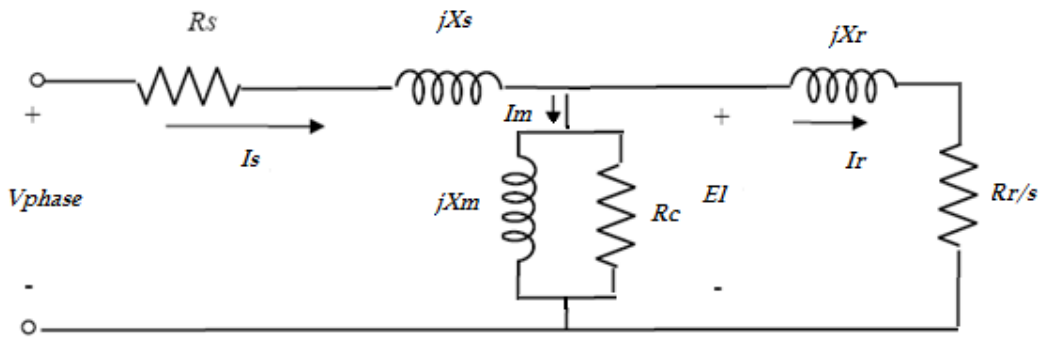

Figure 3.1.3: per-phase equivalent circuit of an induction motor

The rotor slip is represented by s and the resistance Rc, for practical purposes, can be neglected since the equivalent impedance of Rr and jXr is usually much smaller than it (Rc and Rr+jXr are in parallel) (Chapman, 2005).

3.1.4. Torque in an Induction Motor

Usually in analysis and design of asynchronous motors, the “per-phase equivalent circuit” of induction motors shown in Fig. 3.1.4 is widely used (it is the same circuit that in figure 3.1.3. but neglecting Rc). In the figure, Rs and Rr are the stator and rotor resistances respectively, Lm is the magnetization inductance and Ls and Lr are the stator and rotor inductances. Such inductances Ls (Lr) are defined by

$$\boldsymbol{L_s} = \boldsymbol{Ll_s} + \mathbf{L_m}, \mathbf{L_r} = \boldsymbol{Ll_r} + \mathbf{L_m} \qquad (3.1.6)$$

where Lls (Llr) is the stator (rotor) leakage inductance. In this equivalent circuit all rotor parameters are quantities referred to the stator.

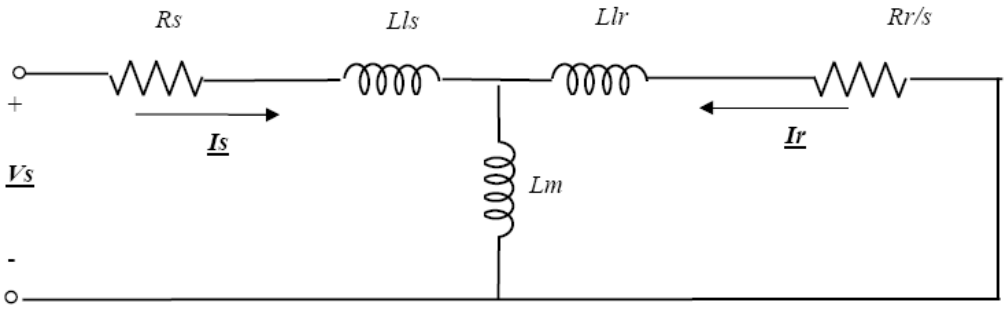

Figure 3.1.4: per-phase equivalent circuit

As seen before asynchronous motors do not rotate synchronously to the excitation frequency, the speed of induction motors are slightly (about 2 - 7% slip in many cases) less than the synchronous speed, at rate load. Therefore the slip can be defined as

$$\boldsymbol{s} = \frac{(\boldsymbol{\omega_e} - \boldsymbol{\omega_o})}{\boldsymbol{\omega_e}} = \frac{\boldsymbol{\omega_{slip}}}{\boldsymbol{\omega_e}} \qquad (3.1.6)$$

where ωe is the excitation frequency injected into the stator, ωo is the actual speed converted into electrical frequency unit, and ω*slip* is called the slip frequency which is the frequency of the actual rotor current. For steady-state of the AC circuit, voltage and current phasors are denoted by ~ on the top of it.

Power consumption in the stator is interpreted as:

$\tilde{I}_S^2 R_S$, while

$\tilde{I}_S^2 R_r/_s$

represents both power consumption in the rotor and the mechanical output (torque). By subtracting these two quantities produced torque (mechanical power divided by the shaft speed) can be obtained

$$T_e = \frac{\tilde{I}_r^2 R_r \left(P/_2\right)(1-s)}{(s\ \omega_o)} = \tilde{I}_r^2 R_r \left(P/_{2\omega_e}\right) \qquad (3.1.7)$$

where *P* is the number of poles. However, the per-phase equivalent circuit is not applicable to explain dynamic performance of the induction motor; a dynamic model is needed (Ohm, 2000).

3.1.5. Dynamic Equivalent Circuit for Arbitrary Rotating Reference Frame

In this analysis vectors are denoted **in red**. Quantities on a rotating reference frame are followed by a superscript letter designating the frame used V_S^S (Vs in

stationary frame). The derivative operator is denoted by *p* while *P* is the number of poles. Real and Imaginary values of a space vector ***Y*** are denoted by *Re*(***Y***) and *Im*(***Y***), respectively. Zero vectors are denoted by ***0*** regardless of the reference frame used (Ohm, 2000).

Assuming uniform air-gap and neglecting the effects of slot harmonics, the distribution of magnetic flux is sinusoidal. The neutral connection of the machine is open so that phase voltages, currents and flux linkages are always balanced and there is no zero phase sequence component in the system. For 3 phase induction motors, the space vector Y_S^s of the stator voltage, current and flux linkage is defined from its phase quantities by

$$\mathbf{\frac{s}{s}} = \left(\frac{2}{3}\right)(\mathbf{Y_a} + \boldsymbol{\alpha}\mathbf{Y_b} + \boldsymbol{\alpha}^2\mathbf{Y_c}) \qquad (3.1.8)$$

where α = exp(j 2π/3). Each phase quantities can be calculated from the space vector by,

$$\mathbf{Ya} = \mathbf{Re}(\mathbf{Y_s^s})$$

$$\mathbf{Ib} = \mathbf{Re}(\boldsymbol{\alpha}^2\mathbf{Y_s^s})$$

$$\mathbf{Ic} = \mathbf{Re}(\boldsymbol{\alpha}\mathbf{Y_s^s}) \qquad (3.1.9)$$

The corresponding space vector, for a sinusoidal 3-phase quantity of constant rms value, is a constant-magnitude vector rotating at the frequency of the sinusoid with respect to the fixed (stationary) reference frame. The space vector is at vector angle 0 when A-phase signal (Ya, first phase) is at its sinusoidal peak

value in steady-state. Voltage equations on the stator and rotor circuits of induction motors are,

$$\mathbf{V_s^s} = \mathbf{R_s I_s^s} + \mathbf{p\lambda_s^s} \qquad (3.1.10)$$

$$\mathbf{V_{r'}} = \mathbf{R_{r'} I_{r'}} + \mathbf{p\lambda_{r'}} = \mathbf{0} \qquad (3.1.11)$$

Rotor variables (**Vr', Ir', λr'**) from eq. 3.1.11 on a rotor reference frame must be transformed to a stator reference frame ($\boldsymbol{V_r^s}$, $\boldsymbol{I_r^s}$, $\boldsymbol{\lambda_r^s}$). Define the stator to rotor winding turn ratio as *n* and the angular position of the rotor as *θ*

$$\mathbf{I_r^s} = \left(\frac{\mathbf{1}}{\mathbf{n}}\right) \mathbf{e^{j\theta} I_{r'}} \qquad (3.1.12)$$

$$\boldsymbol{\lambda_r^s} = \boldsymbol{n e^{j\theta} \lambda_{r'}} \qquad (3.1.13)$$

Also, by defining referred rotor impedances as Rr = n^2Rr', etc., and replacing equations 3.1.12 and 3.1.13 into 3.1.11 the following is obtained

$$\mathbf{0} = \boldsymbol{R_r} \mathbf{I_r^s} + (\boldsymbol{p} - \boldsymbol{j\omega_o}) \boldsymbol{\lambda_r^s} \qquad (3.1.14)$$

$$\boldsymbol{\lambda_r^s} = \mathbf{L_s I_s^s} + \mathbf{L_m I_r^s} \qquad (3.1.15)$$

$$\boldsymbol{\lambda_r^s} = \boldsymbol{L_m I_s^s} + \boldsymbol{L_r I_r^s} \qquad (3.1.16)$$

where ωo = p θo, is the speed of the motor in electrical frequency unit.

The equations 3.1.10, 3.1.14-3.1.16 constitute a dynamic model of the induction motor on a stationary (stator) reference frame in space vector form. Eliminating flux linkages the model is simplified as

$$\mathbf{V_s^s} = (\mathbf{R_s} + \mathbf{L_s p})\mathbf{I_s^s} + \mathbf{L_m p I_r^s} \quad (3.1.17)$$

$$\mathbf{0} = \left(\boldsymbol{R_r} + \boldsymbol{L_r}(\boldsymbol{p} - \boldsymbol{j\omega_o})\right)\mathbf{I_r^s} + \boldsymbol{L_m}(\boldsymbol{p} - \boldsymbol{j\omega_o})\mathbf{I_s^s} \quad (3.1.18)$$

Based on Eqs. 3.1.17-3.1.18 a dynamic equivalent circuit model on a stationary reference frame is drawn (Fig. 3.1.5).

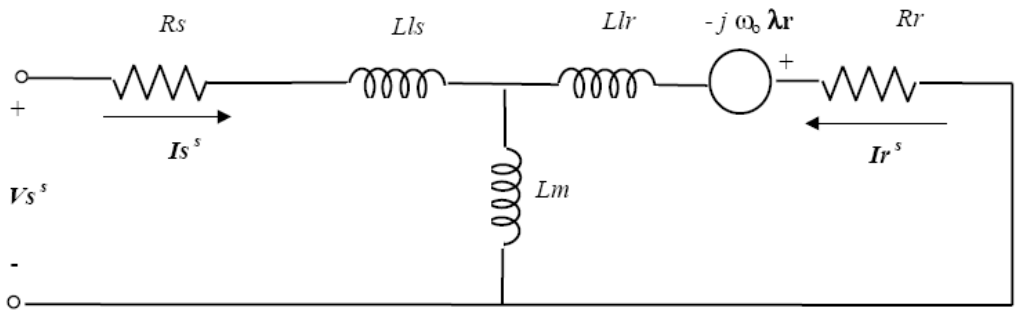

Figure 3.1.5: dynamic equivalent circuit on a stationary reference frame

Considering excitation frequency as ωe for steady-state and p in Eq. 3.1.17-3.1.18 being replaced by jωe, the equations are now

$$\mathbf{V_s^s} = (\mathbf{R_s} + \boldsymbol{j\omega_e}\mathbf{L_s})\mathbf{I_s^s} + \mathbf{L_m}\boldsymbol{j\omega_e}\mathbf{I_r^s} \quad (3.1.19)$$

$$\mathbf{0} = \left(\frac{\boldsymbol{R_r}}{\boldsymbol{s}} + \boldsymbol{j\omega_e L_r}\right)\mathbf{I_r^s} + \boldsymbol{j\omega_e L_m}\mathbf{I_s^s} \quad (3.1.20)$$

The procedure can be generalized so that the dynamic model is described on an arbitrary reference frame rotating at a speed ωa, where Eq. 3.1.10, 3.1.14 - 3.1.20 are a special case with ωa= 0 (Trzynadlowski, 1994) - (Novotney, 1986). The new space vector on the arbitrary frame is defined as

$$\mathbf{Y^a} = \mathbf{e}^{-\mathbf{j}\theta_\mathbf{a}}\boldsymbol{Y^s} \quad (3.1.21)$$

In the arbitrary reference frame, eqs. 3.1.10, 3.1.14 are modified to

$$V_s^a = (R_s + L_s p)\mathbf{I}_s^a + L_m p\mathbf{I}_r^a + \mathbf{j}\omega_a \lambda_s^a \quad (3.1.22)$$

$$\mathbf{0} = (R_r + L_r p)\mathbf{I}_r^a + L_m \mathbf{I}_s^a p + j(\omega_a - \omega_o)\lambda_r^a \quad (3.1.23)$$

The flux linkage equations are defined as,

$$\lambda_s^a = \mathbf{L}_s \mathbf{I}_s^a + \mathbf{L}_m \mathbf{I}_r^a \quad (3.1.24)$$

$$\lambda_r^a = \mathbf{L}_m \mathbf{I}_s^a + \mathbf{L}_r \mathbf{I}_r^a \quad (3.1.25)$$

Applying the same substitution flux linkage variables are eliminated.

$$V_s^a = \big(R_s + L_s(p + j\omega_a)\big)\mathbf{I}_s^a + L_m(p + j\omega_a)\mathbf{I}_r^a \quad (3.1.26)$$

$$\mathbf{0} = \big(R_r + L_r(p + j\omega_a - j\omega_o)\big)\mathbf{I}_r^a + L_m \mathbf{I}_s^a (p + j\omega_a - j\omega_o) \quad (3.1.27)$$

The generalized equivalent circuit on an arbitrarily rotating frame based on Eq. 3.1.26-3.1.27 is shown in Fig. 3.1.6. Many forms of dynamic equivalent circuit can be established depending on ωa. The synchronous frame form can be obtained by choosing ωa = ωe (Ohm, 2000).

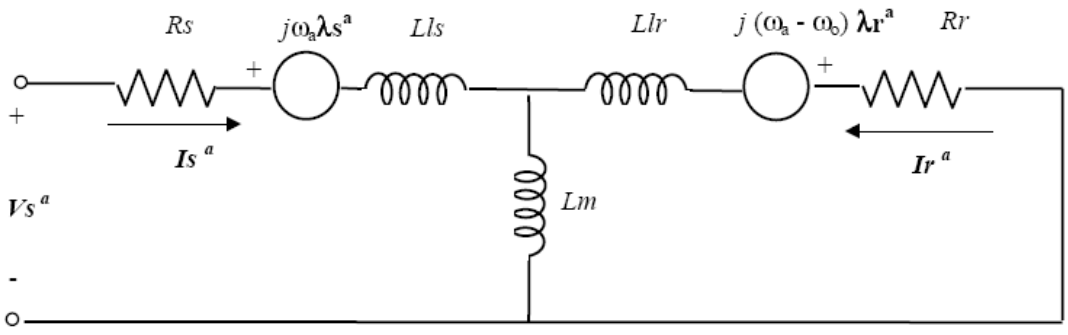

Figure 3.1.6: dynamic equivalent circuit on an arbitrary reference frame rotating at ωa

3.1.6. D-Q Equivalent Circuit

The variables of complex numbers that arise very often in the analysis of induction motors complicate the space vector model, D-Q analysis is then useful. For any space vector Y, define two real quantities Sq and Sd as,

$$\mathbf{S} = \mathbf{S_q} - \mathbf{jS_d} \qquad (3.1.28)$$

This is, Sq = *Re* (S) and Sd = - *Im* (S). Fig. 3.1.7 shows the relationship between d-q axis and complex plane on a rotating frame with respect to stationary a-b-c frame. Axes d- and q are defined on a rotating reference frame, where ωa = *p* θa, with respect to fixed a-b-c frame.

Eq. 3.1.26 – 3.1.27 can be expressed in a matrix form as follows

$$\begin{bmatrix} Vq_s^a \\ Vd_s^a \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + L_s p & \omega_a L_s & L_m p & \omega_a L_m \\ -\omega_a L_s & R_s + L_s p & -\omega_a L_m & L_m p \\ L_m p & (\omega_a - \omega_o) L_m & R_r + L_r p & (\omega_a - \omega_o) L_r \\ -(\omega_a - \omega_o) L_m & L_m p & -(\omega_a - \omega_o) L_r & R_r + L_r p \end{bmatrix} * \begin{bmatrix} Iq_s^a \\ Id_s^a \\ Iq_r^a \\ Id_r^a \end{bmatrix}$$

(3.1.29)

Substituting ωa = 0 (stationary reference frame), the above equation is reduced to

$$\begin{bmatrix} Vq_s^s \\ Vd_s \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + L_s p & 0 & L_m p & 0 \\ 0 & R_s + L_s p & 0 & L_m p \\ L_m p & (-\omega_o) L_m & R_r + L_r p & (-\omega_o) L_r \\ (\omega_o) L_m & L_m p & (\omega_o) L_r & R_r + L_r p \end{bmatrix} * \begin{bmatrix} Iq_s^s \\ Id_s^s \\ Iq_r^s \\ Id_r^s \end{bmatrix}$$

(3.1.30)

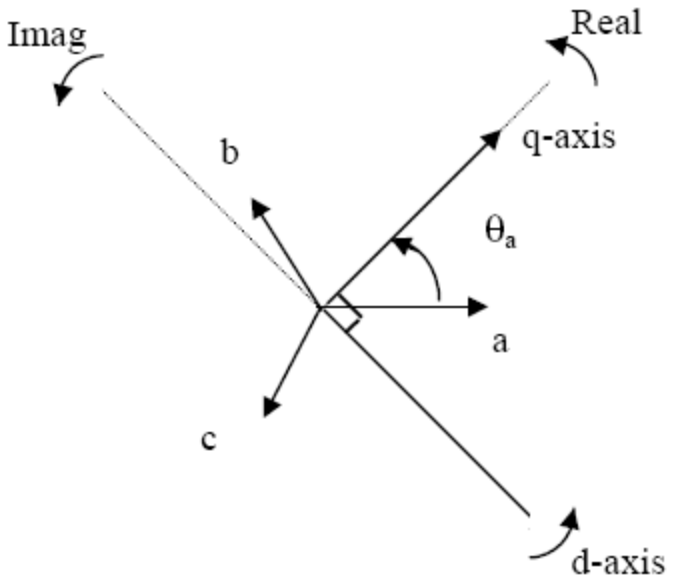

Figure 3.1.7: definition of d-axis and q-axis on an arbitrary reference frame

If calculation on rotor reference frame is needed (rotor rotating at ωo), all ωa in equation 3.1.29 must be substituted by ωo. Equations simplified for synchronous frame are:

$$\begin{bmatrix} Vq_s \\ Vd_s \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + L_s p & \omega_e L_s & L_m p & \omega_e L_m \\ -\omega_e L_s & R_s + L_s p & -\omega_e L_m & L_m p \\ L_m p & (\omega_{slip}) L_m & R_r + L_r p & (\omega_{slip}) L_r \\ -(\omega_e) L_m & L_m p & -(\omega_{slip}) L_r & R_r + L_r p \end{bmatrix} * \begin{bmatrix} Iq_s \\ Id_s \\ Iq_r \\ Id_r \end{bmatrix} \quad (3.1.31)$$

Each variable (voltage, current or flux linkage) on the synchronous frame is stationary and fixed to a constant magnitude in steady-state. Dynamic d-q equivalent circuit is shown in Fig. 3.1.8 (Ohm, 2000).

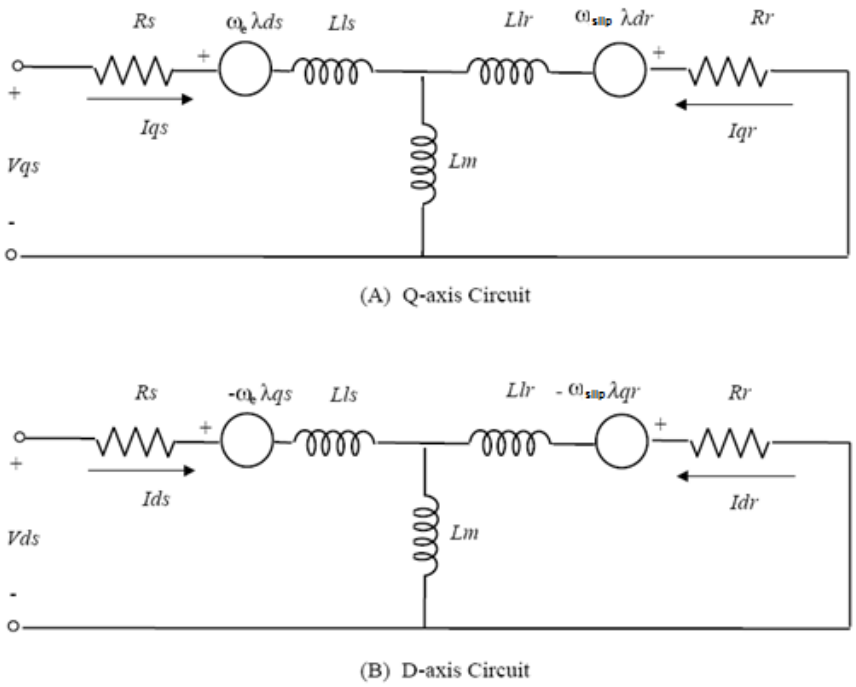

Figure 3.1.8: D-Q Equivalent Circuit on a Synchronous Frame

3.1.7. State-Space Standard Form In D-Q Plane

Although equations 3.1.30 and 3.1.31 can be used for dynamic simulations of the motors, a standard differential state-space form is preferred.

$$\dot{X} = AX + BU \qquad (3.1.32)$$

Therefore, for a synchronous frame, equation 3.1.31 is transformed to (Ohm, 2000):

$$\begin{bmatrix} \dot{Iq_s} \\ Id_s \\ Iq_r \\ Id_r \end{bmatrix} =$$

$$\frac{1}{\Delta}\begin{bmatrix} R_s L_r & \omega_e L_s L_r - \omega_{slip} L_m^2 & -R_r L_m & \omega_o L_r L_m \\ \omega_{slip} L_m^2 - \omega_e L_s L_r & R_s L_r & -\omega_o L_r L_m & -R_r L_m \\ -R_s L_m & -\omega_o L_s L_m & R_r L_s & \omega_{slip} L_s L_r - \omega_e L_m^2 \\ \omega_o L_s L_m & -R_s L_m & \omega_e L_m^2 - \omega_{slip} L_s L_r & R_r L_s \end{bmatrix} *$$

$$\begin{bmatrix} Iq_s \\ Id_s \\ Iq_r \\ Id_r \end{bmatrix} + \frac{1}{\Delta}\begin{bmatrix} L_r & 0 & -L_m & 0 \\ 0 & L_r & 0 & -L_m \\ -L_m & 0 & L_s & 0 \\ 0 & -L_m & 0 & L_s \end{bmatrix} * \begin{bmatrix} Vq_s \\ Vd_s \\ 0 \\ 0 \end{bmatrix} \qquad (3.1.33)$$

where $\Delta = \mathrm{L_s L_r} - L_m^2$. Matrices and variables can be defined as follows.

$$X = \begin{bmatrix} Iq_s \\ Id_s \\ Iq_r \\ Id_r \end{bmatrix} \qquad (3.1.34)$$

$$A =$$

$$\frac{1}{\Delta}\begin{bmatrix} R_s L_r & \omega_e L_s L_r - \omega_{slip} L_m^2 & -R_r L_m & \omega_o L_r L_m \\ \omega_{slip} L_m^2 - \omega_e L_s L_r & R_s L_r & -\omega_o L_r L_m & -R_r L_m \\ -R_s L_m & -\omega_o L_s L_m & R_r L_s & \omega_{slip} L_s L_r - \omega_e L_m^2 \\ \omega_o L_s L_m & -R_s L_m & \omega_e L_m^2 - \omega_{slip} L_s L_r & R_r L_s \end{bmatrix}$$

(3.1.35)

$$B = \frac{1}{\Delta}\begin{bmatrix} L_r & 0 & -L_m & 0 \\ 0 & L_r & 0 & -L_m \\ -L_m & 0 & L_s & 0 \\ 0 & -L_m & 0 & L_s \end{bmatrix} \qquad (3.1.36)$$

$$U=\begin{bmatrix} Vq_s \\ Vd_s \\ 0 \\ 0 \end{bmatrix} \quad (3.1.37)$$

Additional to equations 3.1.34-3.1.37, equations for flux linkage are needed to complete the dynamic description of the motor. Equations 3.1.25 and 3.1.22 translated to d-q plane on synchronous frame result in ($\omega_e = \omega_s;\ \omega_o = \omega_r$):

$$Vq_s = R_s Iq_s + \mathrm{p}\lambda q_s + \omega_s \lambda d_s \quad (3.1.38)$$

$$Vd_s = R_s Id_s + \mathrm{p}\lambda d_s - \omega_s \lambda q_s \quad (3.1.39)$$

$$0 = R_r Iq_r + \mathrm{p}\lambda q_r + \omega_{slip} \lambda d_r \quad (3.1.40)$$

$$0 = R_r Id_r + \mathrm{p}\lambda d_r + \omega_{slip} \lambda q_r \quad (3.1.41)$$

where flux linkages are:

$$\lambda q_s = L_s Iq_s + L_m Iq_r \quad (3.1.42)$$

$$\lambda d_s = L_s Id_s + L_m Id_r \quad (3.1.43)$$

$$\lambda q_r = L_m Iq_s + L_r q_r \quad (3.1.44)$$

$$\lambda d_r = L_m Id_s + L_r Id_r \quad (3.1.45)$$

3.1.8. Torque and Power In D-Q Plane

Instantaneous input power can be expressed in terms of space vectors:

$$P_i = (3/2)Re(V_s I_s^{@}) \quad (3.1.46)$$

where @ states for the conjugate of the vector. In d-q plane:

$$P_i = (3/2)[Vd_s Id_s + Vq_s Iq_s] \quad (3.1.47)$$

The reactive power can be defined as:

$$Q_i = (3/2)Im(V_s I_s^{@}) \quad (3.1.48)$$

and in d-q plane:

$$Q_i = (3/2)[Vq_s Id_s - Vd_s Iq_s] \quad (3.1.49)$$

Considering the power Po associated with speed voltage

$$P_o = (3/2)\omega_o Im(\lambda_r I_r^{@}) \quad (3.1.50)$$

the torque can be found (Po/wo)

$$T_e = (3/2)P\ Im(\lambda_r I_r^{@}) \quad (3.1.51)$$

where P is the number of poles. In d-q plane torque is:

$$T_e = (3/2)P[\lambda q_r Id_r - \lambda d_r Iq_r] \quad (3.1.52)$$

This torque equation is true for any reference frame. Substituting flux linkage of equation 3.1.15 into equation 3.1.51 other useful equations of torque are found (Ohm, 2000).

$$T_e = (3/2)P\ L_m\ Im(I_s I_r^{@}) \quad (3.1.53)$$

$$T_e = (3/2)P\ L_m[Iq_s Id_r - Id_s Iq_r] \quad (3.1.54)$$

3.1.9. Dynamic Block Diagram for an Induction Motor In D-Q Plane.

Remember that for a synchronous frame, the references for synchronous speed and electrical angular velocity are ω_s and ω_r respectively. Summary of induction motor equations in d-q plane is shown below (Matlab, 2008).

$$Vq_s = R_s Iq_s + \dot{\lambda q}_s + \omega_s \lambda d_s \quad (3.1.55)$$

$$Vd_s = R_s Id_s + \dot{\lambda d}_s - \omega_s \lambda q_s \quad (3.1.56)$$

$$Vq_r = R_r Iq_r + \dot{\lambda q}_r + (\omega_s - \omega_r)\lambda d_r \quad (3.1.57)$$

$$Vd_r = R_r Id_r + \dot{\lambda d}_r - (\omega_s - \omega_r)\lambda q_r \quad (3.1.58)$$

$$\lambda q_s = L_s Iq_s + L_m Iq_r \quad (3.1.59)$$

$$\lambda d_s = L_s Id_s + L_m Id_r \quad (3.1.60)$$

$$\lambda q_r = L_r Iq_r + L_m Iq_s \quad (3.1.61)$$

$$\lambda d_r = L_r Id_r + L_m Id_s \quad (3.1.62)$$

$$L_s = Ll_s + L_m \quad (3.1.63)$$

$$L_r = Ll_r + L_m \quad (3.1.64)$$

Adding some terms to equation 3.1.54 a relation between d-q stator currents and d-q stator fluxes can be found.

$$T_e = (1.5)P\ L_m[Iq_s Id_r - Id_s Iq_r]$$

$$T_e = (1.5)P\ [L_m Iq_s Id_r - L_m Id_s q_r + L_s Id_s Iq_s - L_s Id_s Iq_s]$$

$$T_e = (1.5)P\ [(L_s Id_s + L_m Id_r)Iq_s - (L_s Iq_s + L_m Iq_r)Id_s]$$

From equations 3.1.59 and 3.1.60 the stator fluxes can be replaced

$$T_e = (1.5)P[Iq_s \lambda d_s - Id_s \lambda q_s] \quad (3.1.65)$$

Solving Iq_r and Id_r from 3.1.59 and 3.1.61 yields to

$$Iq_r = \frac{\lambda q_s - L_s Iq_s}{L_m} \quad (3.1.66)$$

$$Id_r = \frac{\lambda d_s - L_s Id_s}{L_m} \quad (3.1.67)$$

Similarly, solving Iq_s and Id_s from 3.1.61 and 3.1.62 yields to

$$Iq_s = \frac{\lambda q_r - L_r Iq_r}{L_m} \quad (3.1.68)$$

$$Id_s = \frac{\lambda d_r - L_r Id_r}{L_m} \quad (3.1.69)$$

Replacing 3.1.66 and 3.1.67 in 3.1.68 and 3.1.69

$$Iq_s = \frac{\lambda q_r - L_r \frac{\lambda q_s - L_s Iq_s}{L_m}}{L_m}$$

$$Iq_s = [L_m \lambda q_r - L_r \lambda q_s]\left(\frac{1}{{L_m}^2 - L_r L_s}\right)$$

$$Iq_s = [L_o \lambda q_r - L_c \lambda q_s] \quad (3.1.70)$$

where

$$L_o = \left(\frac{L_m}{{L_m}^2 - L_r L_s}\right) \quad (3.1.71)$$

$$L_c = \left(\frac{L_r}{{L_m}^2 - L_r L_s}\right) \quad (3.1.72)$$

Similarly

$$Id_s = \frac{\lambda d_r - L_r \frac{\lambda d_s - L_s Id_s}{L_m}}{L_m}$$

$$Id_s = [L_o \lambda d_r - L \quad \lambda d_s] \quad (3.1.73)$$

Replacing stator currents in equation 3.1.65 lead to

$$T_e = (1.5)P\left(\frac{1}{{L_m}^2 - L_r L_s}\right)[(L_m \lambda q_r - L_r \lambda q_s)\lambda d_s - (L_m \lambda d_r - L_r \lambda d_s)\lambda q_s]$$

$$T_e = (1.5)PL_o[\lambda q_r \lambda d_s - \lambda d_r \lambda q_s] \quad (3.1.73)$$

In this analysis making the torque depending of the fluxes is useful because the fluxes can be found solving the equations 3.1.55 to 3.1.58 and replacing the respective currents; therefore

$$\dot{\lambda q_s} = Vq_s - R_s Iq_s - \omega_s \lambda d_s$$

Replacing the q stator current

$$\dot{\lambda q_s} = Vq_s - R_s[L_o \lambda q_r - L_c \lambda q_s] - \omega_s \lambda d_s \quad (3.1.74)$$

Similarly

$$\dot{\lambda d} \quad = Vd_s - R_s[L_o \lambda d_r - L_c \lambda d_s] + \omega_s \lambda q_s \qquad (3.1.75)$$

The voltages in the rotor (Vq_r and Vd_r) side are zero and the currents in the rotor can be found following the same procedure than for the stator currents.

$$Iq_r = \frac{\lambda q_s - L_s Iq_s}{L_m}$$

$$Iq_r = \frac{\lambda q_s - L_s\left(\frac{\lambda q_r - L_r \frac{\lambda q_s - L_s Iq_s}{L_m}}{L_m}\right)}{L_m}$$

$$Iq_r = [L_o \lambda q_s - L_c \lambda q_r] \qquad (3.1.76)$$

$$Id_r = [L_o \lambda d_s - L_c \lambda d_r] \qquad (3.1.76)$$

Replacing rotor currents into the fluxes differential equations yields to:

$$\dot{\lambda q_r} = -R_r Iq_r - (\omega_s - \omega_r)\lambda d_r$$

$$\dot{\lambda q_r} = -R_r[L_o \lambda q_s - L_c \lambda q_r] - (\omega_s - \omega_r)\lambda d_r \qquad (3.1.77)$$

$$\dot{\lambda d_r} = -R_r Id_r + (\omega - \omega_r)\lambda q_r$$

$$\dot{\lambda d_r} = -R_r[L_o \lambda d_s - L_c \lambda d_r] + (\omega_s - \omega_r)\lambda q_r \qquad (3.1.78)$$

This is a system of 4 first order coupled differential equations, along with the torque form a system of 5 unknowns and 5 equations, depending only on the d-q voltages. The block diagram of the dynamic model is shown in figure 3.1.9.

Mechanical system dynamics can be described by (Matlab, 2008), (Krause, 2002):

$$\dot{\omega}_m = \frac{1}{J}(T_e - F\omega_m - T_l) \quad (3.1.79)$$

$$\dot{\theta}_m = \omega_m \quad (3.1.80)$$

$$\omega_r = P\omega_m \quad (3.1.81)$$

where J is the combined rotor and load inertial coefficient, F is the friction coefficient, Tl is the mechanical load, θ_m is the rotor angular position, P is the number of poles and, ω_r is the electrical angular velocity. The mechanical system is represented in the block diagram of figure 3.1.11

A complete dynamic block diagram can be seen in figure 3.1.10. Notice that the inputs for the system are the 3-phase voltage input (voltage d-q can be found applying a transformation to the 3-phase abc voltage) and its frequency. The system should simulate any change on the input by giving the corresponding Te and wm. The error between wm simulated and wm real should be as close as possible to zero in order to achieve a robust and stable control.

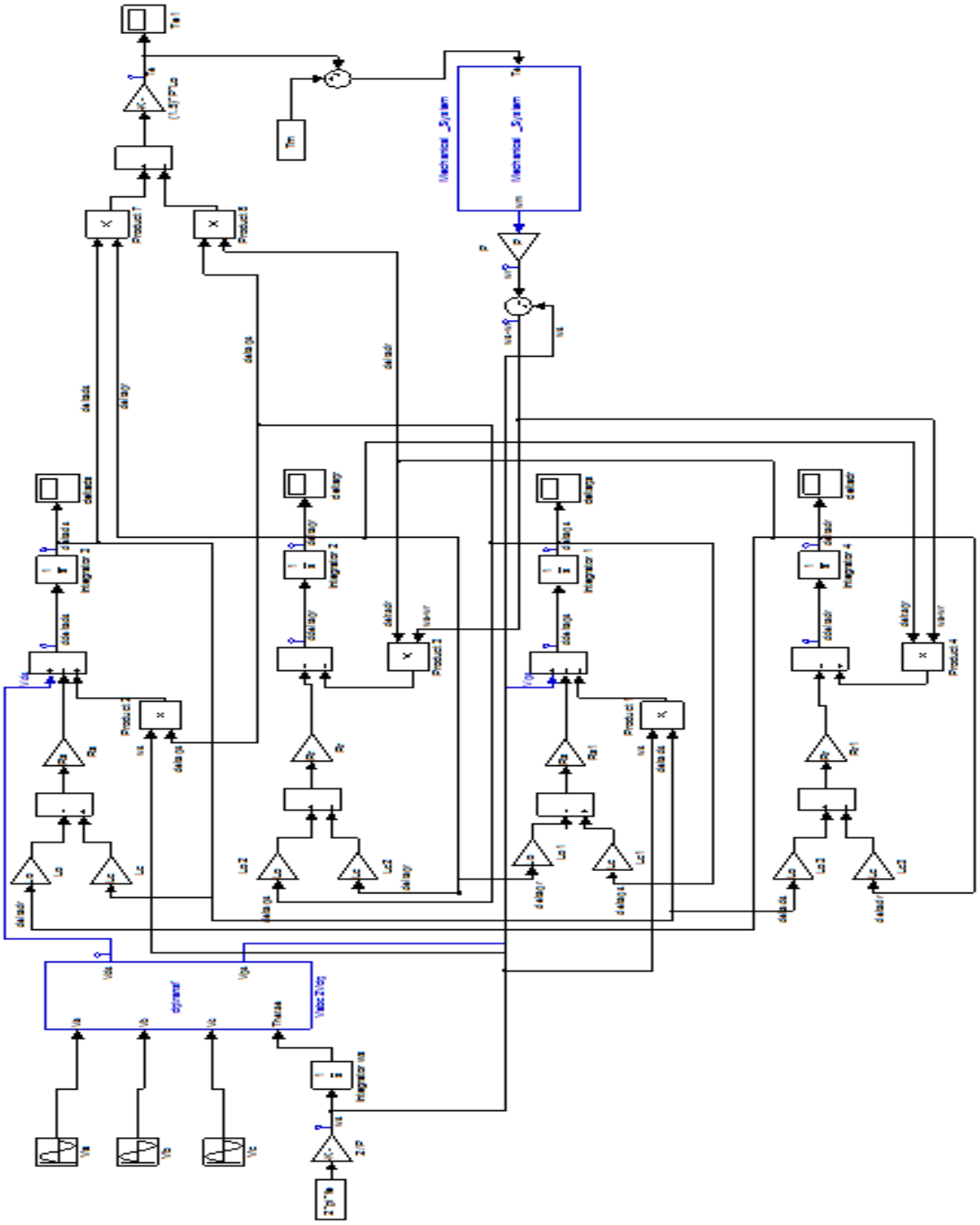

Figure 3.1.9: dynamic block diagram of an induction motor in d-q plane

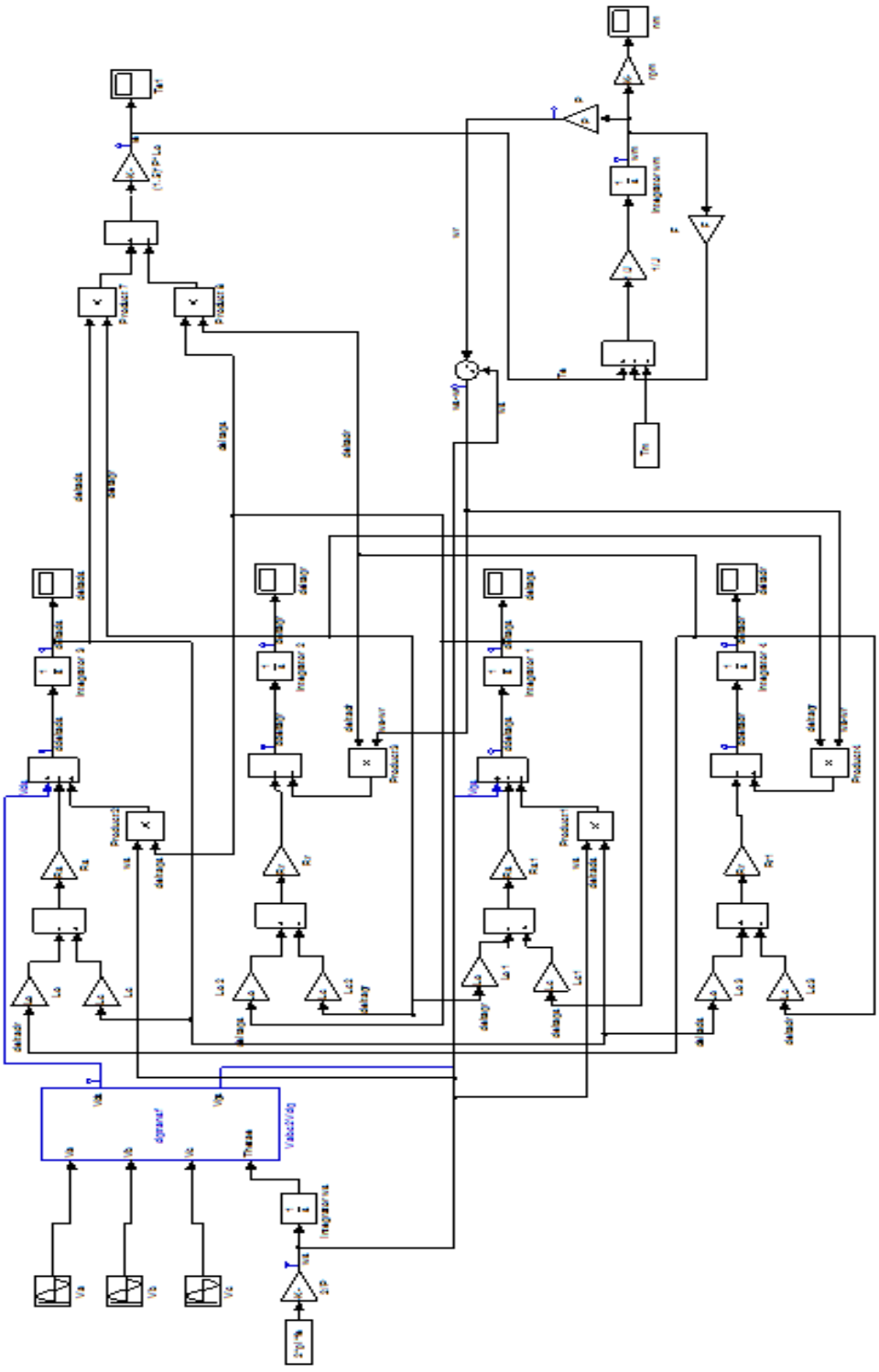

Figure 3.1.10: dynamic block diagram of an induction motor in d-q plane

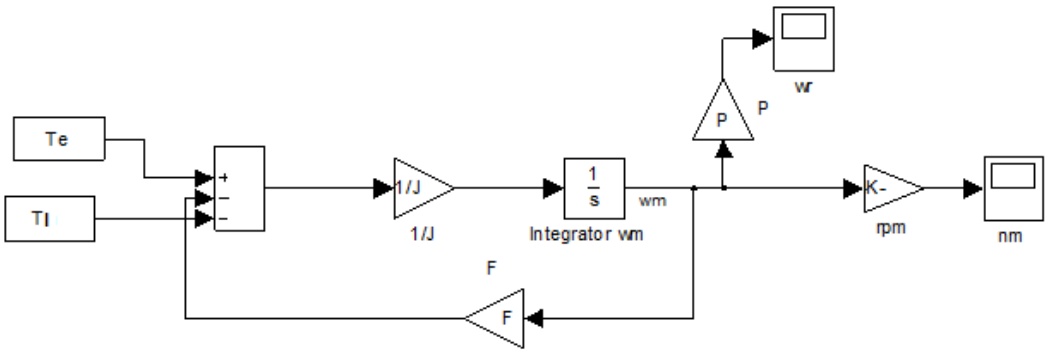

Figure 3.1.11: dynamic block diagram of mechanical system of an induction motor

3.1.10. Transformation Matrix

Equations 3.1.73 and 3.1.79 describe the torque and the rotor speed with respect to the input in d-q plane; however a transformation matrix is needed to convert d-q values to stationary a-b-c frame.

$$Sq_s = \left(2/3\right) Re\left\{e^{-j\,\theta_a}\left(S_a + \alpha S_b + \alpha^2 S_c\right)\right\} \qquad (3.1.82)$$

$$Sd_s = -\left(2/3\right) Im\left\{e^{-j\,\theta_a}\left(S_a + \alpha S_b + \alpha^2 S_c\right)\right\} \qquad (3.1.83)$$

where, recalling from equation 3.1.8 and 3.1.9, $\alpha = e^{j\frac{2\pi}{3}}$

Or in a matrix form:

$$\begin{bmatrix} Yq \\ Yd \\ 0 \end{bmatrix} = \left(2/3\right) \begin{bmatrix} cos\,\theta & cos\left(\theta - 2\pi/3\right) & cos\left(\theta + 2\pi/3\right) \\ sin\,\theta & sin\left(\theta - 2\pi/3\right) & sin\left(\theta + 2\pi/3\right) \\ 0.5 & 0.5 & 0.5 \end{bmatrix} \begin{bmatrix} Ya \\ Yb \\ Yc \end{bmatrix} \qquad (3.1.84)$$

The inverse transform is:

$$\begin{bmatrix} Ya \\ Yb \\ Yc \end{bmatrix} = \begin{bmatrix} cos\,\theta & sin\,\theta & 1 \\ cos\left(\theta - {}^{2\pi}/_{3}\right) & sin\left(\theta - {}^{2\pi}/_{3}\right) & 1 \\ cos\left(\theta + {}^{2\pi}/_{3}\right) & sin\left(\theta + {}^{2\pi}/_{3}\right) & 1 \end{bmatrix} \begin{bmatrix} Yq \\ Yd \\ 0 \end{bmatrix} \qquad (3.1.85)$$

Equation 3.1.84 is generally used to convert measured currents and voltages to d-q plane while equation 3.1.85 is used to feed command signals to the system.

In a synchronous frame $\boldsymbol{\theta}$ is the integral of the synchronous speed, i.e.

$$\dot{\theta} = \dot{\theta}_e = \dot{\theta}_s = \omega_e = \omega_s \qquad (3.1.86)$$

3.2. STABILITY OF THE SYSTEM ON DIFFERENTIAL STATE-SPACE FORM

In order to identify if the system is stable or not the standard state-space form (Equation 3.1.33) is needed. The eigenvalues of matrix A (equation 3.1.35) must have all a negative real part; if only one of them has a positive real part the system then is unstable (Figari, 2007).

For this system, values shown in table 4.1.1. are used; the system is assumed to be feed at 60Hz and its maximum rotor speed is 1755 rpm.

The eigenvalues for matrix A are:

EIG=

-0.7352 +26.3554i

-0.7352 -26.3554i

-0.1267 + 1.6572i

-0.1267 - 1.6572i

It can be seen that all real parts are negative thus the system is stable in principle.

3.3. CONTROLLABILITY OF MECHANICAL SYSTEM

Prior of deciding what control strategy is the best an analysis of controllability must be done. Controllability tells the possibility of forcing the system into a particular state by using the control variables. If a state or system is not controllable no input signal will be able to control the system. Every state of a system must be controllable for the system to have a good behavior in a closed-loop system.

3.3.1. Analysis of Controllability of an Inductor Motor with a General Dynamic Model

Generally, the mechanical system dynamics can be simplified as

$$J\ddot{\theta} + F\dot{\theta} + T_l = T_e \qquad (3.3.1)$$

This is a simple dynamic model based on the controllable variable which is the rotation of the motor. Here, *J* is the rotary inertia, *F* is a damping coefficient, θ is the motor's rotation angle, and *Tl* is additional load interference. *Te* is the electromagnetic torque and can be defined as below:

$$T_e = K_t i_{qs} \qquad (3.3.2)$$

$$K_t = \left(\frac{3P}{2}\right)\left(\frac{L_m^2}{L_p}\right) i_{ds} \quad (3.3.3)$$

Where *Kt* is the torque constant, i_{qs} is the torque current command, i_{ds} is the outflow current command, *P* is the pole pair, *Lm* is the air-gap magnetic flux, and *Lr* is the rotor inductance. The dynamic equation of the induction motor can be rewritten as:

$$\ddot{\Theta} = -\frac{F}{J}\dot{\Theta} + \frac{K_t}{J} i_{qs} - \frac{1}{J} T_l = A_p \dot{\Theta} + F_p u(t) - D_p T_l \quad (3.3.4)$$

Here, *Ap* = -*B/J*, *Bp* = *Kt/J* > 0, *Dp* = -1*/J*, and u(t)=$i_{qs}(t)$ is the control command. The variable to control is the rotor angle, therefore the error is:

$$\mathbf{e} = \boldsymbol{\Theta_c} - \boldsymbol{\Theta} \quad (3.3.5)$$

First, provided that both system dynamic functions *Ap* and *Bp* of the induction motor and external interference *Tl* can be acquired, an ideal control rule can be obtained using the feedback control theory.

$$u^* = F_p^{-1}\left[-A_p \dot{\Theta} - D_p T_l + \ddot{\Theta}_c + K_1 \dot{e} + K_2 e\right] \quad (3.3.6)$$

Insert equation 3.3.6 of the ideal control rule into system dynamic equation 3.3.4 and obtain:

$$\ddot{e} + K_1 \dot{e} + K_2 e = 0 \quad (3.3.7)$$

It can be seen that with the proper values of K1 and K2, the roots of the polynomial fall on the left hand side of the real-complex plane, making the system controllable (Zhaoming, 2007).

3.4. AC ANALYSIS

AC motor equations (Chapman, 2005).

$$n_{syn} = \frac{120 * f_e}{P} \quad (3.4.1)$$

$$\omega_s = n_{syn} * {}^{2\pi}/_{60} \quad (3.4.2)$$

$$s = \frac{n_{syn} - n_n}{n_{syn}} \quad (3.4.3)$$

$$P_{AG} = 3I_r^2 \, {}^{R_r}/_{s} \quad (3.4.4)$$

$$T_e = \frac{P_{AG}}{\omega_s} \quad (3.4.5)$$

In steady-state the induce torque, then, is a relation between the air-gap power and the synchronous speed, since the synchronous speed is a constant and the air-gap power can be calculated, the induce torque can be found. The air-gap power is the power crossing the gap from the stator circuit to the rotor circuit. It is equal to the power absorbed by the resistance Rr/s (Chapman, 2005). This power can be expressed in terms of Rr, Xr, Rth, Xth and Vth, where Rth, Xth and Vth are the equivalent Thevenin's resistance, reactance and voltage respectively.

3.4.1. Thevenin's of Equivalent Circuit

Using the model already seen in section 3.1.3. (Rc is neglected) and equivalent circuit can be found. (figures 3.4.1 – 3.4.3)

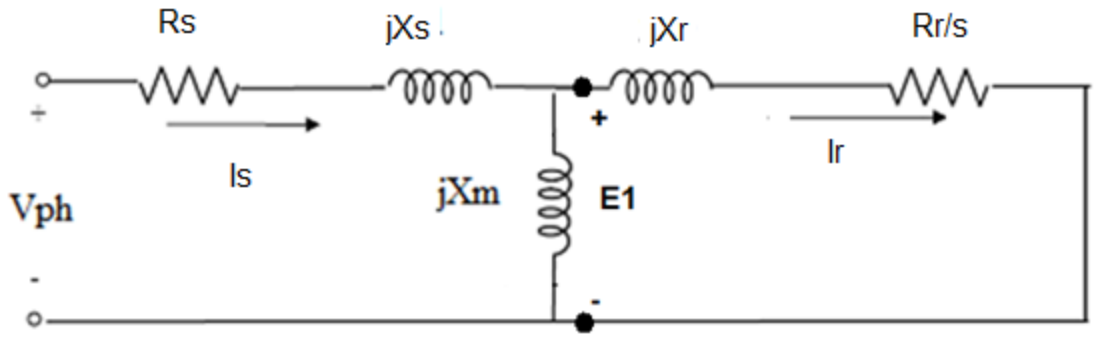

Figure 3.4.1: equivalent circuit

Applying Thevenin to the equivalent circuit, the voltage (E1) is:

$$V_{TH} = V_{\phi} \frac{X_m}{\sqrt{R_s^2 + (X_s + X_m)^2}} \quad (3.4.6)$$

Since the magnetization reactance Xm>>Xs and Xm>>Rs the Thevenin voltage Figure 3.4.2) can be approximated to:

$$V_{TH} \approx V_{\phi} \frac{X_m}{X_s + X_m} \quad (3.4.7)$$

Also the Thevenin impedance can be found (figure 3.4.3) this impedance is:

$$Z_{TH} = R_{TH} + jX_{TH} = \frac{jX_m(R_s + jX_s)}{R_s + j(X_s + X_m)} \quad (3.4.8)$$

Because Xm>>Xs and Xm+Xs>>Rs, the Thevenin impedance is given by:

$$R_{TH} \approx R_s \left(\frac{X_m}{X_s + X_m}\right)^2 \quad (3.4.9)$$

$$X_{TH} \approx X_s \quad (3.4.10)$$

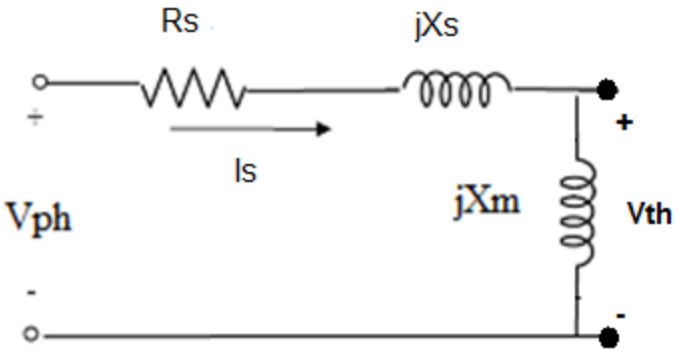

Figure 3.4.2: Thevenin voltage

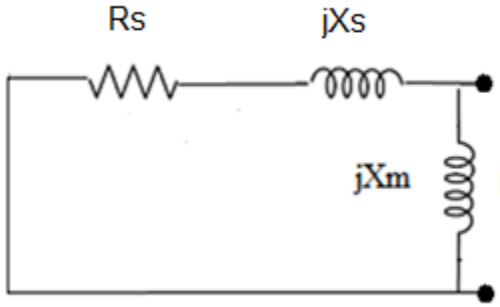

Figure 3.4.3: Thevenin impedance

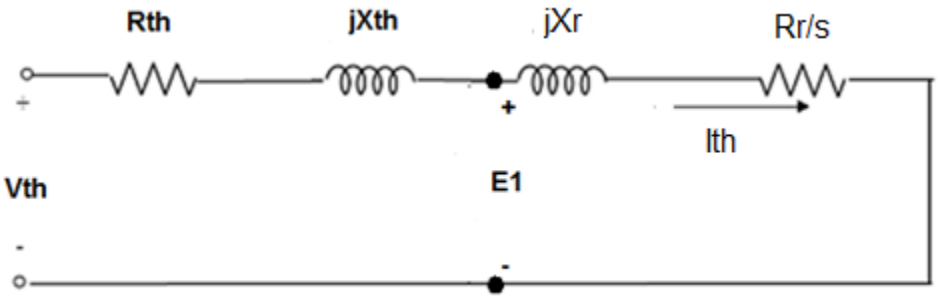

Figure 3.4.4: equivalent circuit after Thevenin

Therefore the equivalent current for the Thevenin circuit (figure 3.4.4) is (which is the current needed in section 3.4.2):

$$I_{TH} = \frac{V_{TH}}{Z_{TH} + Z_2} \qquad (3.4.11)$$

The magnitude of this current is:

$$I_{TH} = \frac{V_{TH}}{\sqrt{\left(R_{TH} + {}^{R_r}/_{s}\right)^2 + (X_{TH} + X_r)^2}} \quad (3.4.12)$$

Replacing this current in the air-gap power equation (equation 3.4.4) and this power in the induced torque equation (equation 3.4.5) yields to:

$$T_e = \frac{3{V_{TH}}^2 {}^{R_r}/_{s}}{\omega_s\left[\left(R_{TH} + {}^{R_r}/_{s}\right)^2 + (X_{TH} + X_r)^2\right]} \quad (3.4.13)$$

It can be seen that the induced torque is function of the Thevenin voltage, the synchronous speed and the slip. Thevenin voltage is function of the phase voltage, which is known. The slip can be calculated having the rotor speed measured (actually the rotor speed is assumed to be another input for the control system) and the synchronous speed. The synchronous speed depends only of the input frequency (equation 3.4.1). So the torque depends on the stator frequency, the rotor speed and the phase voltage.

3.4.2. Torque Vs. Speed Curves

As seen in equation 3.4.13, induced torque depends on synchronous velocity, slip and phase voltage.

Given a 3 phase sinusoidal voltage (230V for this example), several curves torque vs. speed can be found. The first one (figure 2.4.5) would be for an input frequency of 60 Hz.

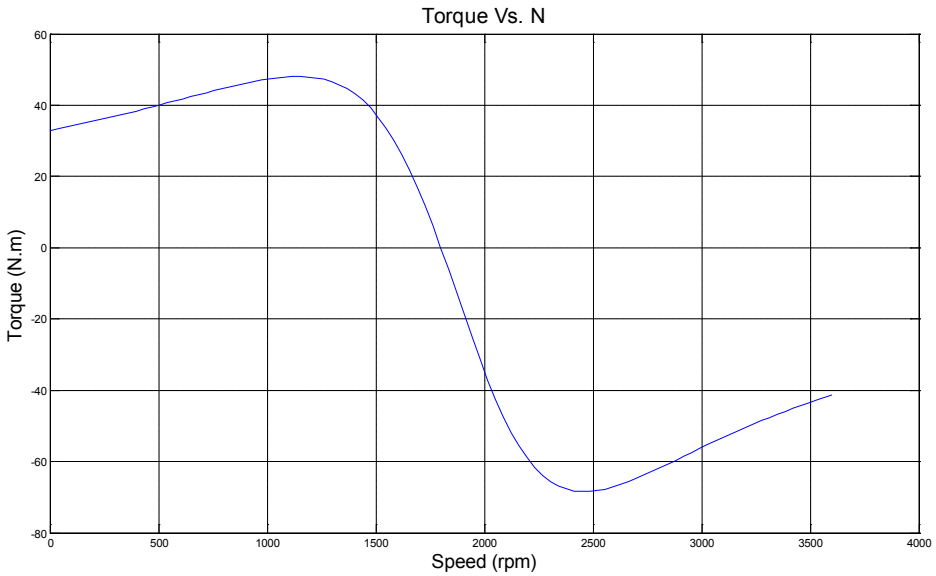

Figure 3.4.5: torque vs. Speed for fe=60 Hz and Vf=230 V

Changing the input frequency (fe) a family of curves is obtained (figure 3.4.6)

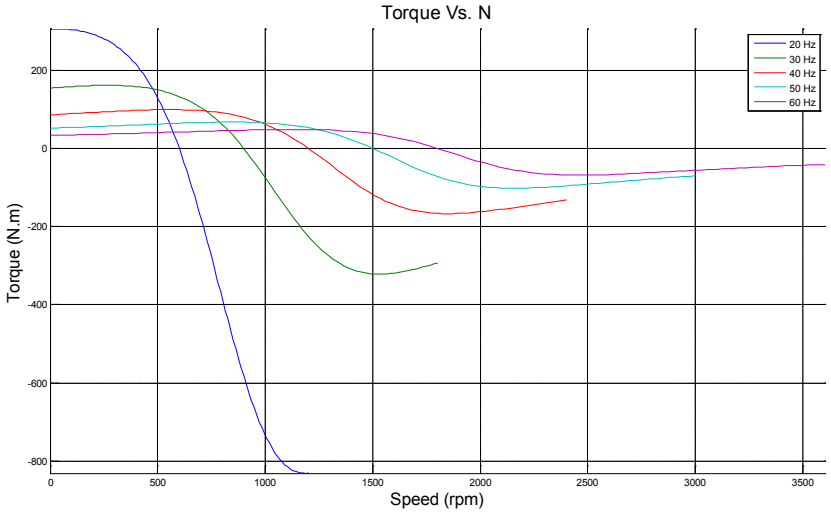

Figure 3.4.6: family curves for fe varying from 20 Hz to 60 Hz, Vf=230 V

The variation of fe, all other things equal, represents a different torque vs. speed curve, the control of the speed will be based on that (detailed curves in appendix 8.4).

3.5. ANALYSIS OF DISTURBANCES IN STEADY-STATE

Rearranging equation 3.4.13, replacing the slip and impedances leads to:

$$T_e(\omega_{el}, \omega_m) = \frac{3{V_\phi}^2\left(\frac{L_m}{L_s+L_m}\right)^2 R_r}{(\omega_s-\omega_m)\left[\left(R_s\left(\frac{L_m}{L_s+L_m}\right)^2+{R_r}/{s}\right)^2+\omega_{el}^2(L_s+L_r)^2\right]} \qquad (3.51)$$

where $\omega_s = \frac{2\omega_{el}}{P}$ and $\omega_{el} = 2\pi f_e$

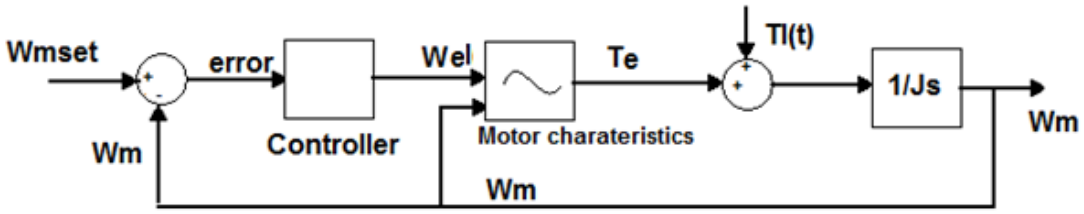

Figure 3.5.1: block diagram of control system

The equation of the system is then:

$$T_e(\omega_{el}, \omega_m) - T_L(t) = J\dot{\omega}_m \qquad (3.5.2)$$

where Tm is the torque applied to the rotor (Te) and TL(t) is the load defined as (Carlton, 1994):

$$T_L(t) = k{\omega_m}^2 \qquad (3.5.3)$$

Linearizing around an equilibrium point for the mechanical torque gives:

$$T_e(\omega_{el}, \omega_m) = T_{eo} + \frac{\partial T_e}{\partial \omega_{el}}\Big|_{\omega_{elo},\omega_{mo}} \delta\omega_{el} + \frac{\partial T_e}{\partial \omega_m}\Big|_{\omega_{elo},\omega_{mo}} \delta\omega_m \qquad (3.5.4)$$

where T_{eo} is the torque evaluated at the equilibrium point, i.e. $T_e(\omega_{elo}, \omega_{mo})$

The perturbation and equilibrium point for the variables ω_m, ω_{el} and k are defined as:

$$\Omega_m = \omega_{mo} + \delta\omega_m = \Omega_{mo} + \omega_m \quad (3.5.5)$$

$$\Omega_{el} = \omega_{elo} + \delta\omega_{el} = \Omega_{elo} + \omega_{el} \quad (3.5.6)$$

$$K = k_o + \delta k = K_o + k \quad (3.5.7)$$

where the capital letter with subindex “o” stands for the equilibrium point and the lower case letter for the perturbation.

The load can be linearized too:

$$T_L(t) = K_o {\Omega_{mo}}^2 + 2K_o\Omega_{mo}\,\omega_m + {\Omega_{mo}}^2 k \quad (3.5.8)$$

Similarly for the acceleration component:

$$J\dot{\Omega}_m = J\dot{\Omega}_{mo} + J\dot{\omega}_m \quad (3.5.9)$$

and for the induced torque:

$$T_e(\omega_{el}, \omega_m) = T_{eo} + \left.\frac{\partial T_e}{\partial \omega_{el}}\right|_{\Omega_{elo}, \Omega_{mo}} \omega_{el} + \left.\frac{\partial T_e}{\partial \omega_m}\right|_{\Omega elo, \Omega_{mo}} \omega_m \quad (3.5.10)$$

Notice that $\dot{\Omega}_{mo} = 0$ since the mechanical speed is assumed constant at the equilibrium point and $T_{eo} = K_o {\Omega_{mo}}^2$ at the equilibrium point since it is steady state.

Replacing all the components in the main equation (3.5.1) yields to:

$$T_{eo} + \left.\frac{\partial T_e}{\partial \omega_{el}}\right|_{\Omega_{elo},\Omega_{mo}} \omega_{el} + \left.\frac{\partial T_e}{\partial \omega_m}\right|_{\Omega o,\Omega_{mo}} \omega_m - K_o {\Omega_{mo}}^2 - 2K_o \Omega_{mo}\,\omega_m - {\Omega_{mo}}^2 k = J\dot{\omega}_m \tag{3.5.11}$$

Rearranging the terms

$$J\dot{\omega}_m + \omega_m \left(2K_o\Omega_{mo} - \left.\frac{\partial T_e}{\partial \omega_m}\right|_{\Omega o,\Omega_{mo}}\right) - \left.\frac{\partial T_e}{\partial \omega_{el}}\right|_{\Omega_{elo}\,\Omega_{mo}} \omega_{el} + {\Omega_{mo}}^2 k = 0 \tag{3.5.12}$$

$\left.\frac{\partial T_e}{\partial \omega_m}\right|_{\Omega o,\Omega_{mo}}$ and $\left.\frac{\partial T_e}{\partial \omega_{el}}\right|_{\Omega_{elo},\Omega_{mo}}$ are constants once the partial derivatives are evaluated, for simplicity are going to be called $T_{\Omega mo}$ and $T_{\Omega elo}$ (the step by step procedure to find them can be seen in appendix 8.5) The solution for ω_m after being transformed to Laplace domain is:

$$\omega_m(s) = \frac{-T_{\Omega elo}\,\omega_{el}(s) + {\Omega_{mo}}^2 k(s)}{Js + 2K_o\Omega_{mo} - T_{\Omega m}} \tag{3.5.13}$$

The control term, therefore, must change $\omega_{el}(s)$ to affect the result.

3.5.1. Plant Dynamics

Equation 3.5.13 can be expressed as

$$\omega_m(s) = \frac{-T_{\Omega elo}\,\omega_{el}(s)}{Js + 2K_o\Omega_{mo} - T_{\Omega mo}} + \frac{{\Omega_{mo}}^2 k(s)}{Js + 2K_o\Omega_{mo} - T_{\Omega mo}} \tag{3.5.14}$$

or in compact form

$$\omega_m(s) = G_1(s)\omega_e(s) + G_2(s)k(s) \tag{3.5.15}$$

Note that

$$G_1(s) = \frac{\omega_m(s)}{\omega_{el}(s)} = \frac{-T_{\Omega elo}}{Js + 2K_o\Omega_{mo} - T_{\Omega mo}} \quad (3.5.16)$$

The poles on $G_1(s)$ and $G_2(s)$ must be stable; in order to have that, the real part of the roots of the characteristic polynomial must be negative. Analyzing the roots of the characteristic polynomial:

$$Js + 2K_o\Omega_{mo} - T_{\Omega mo} = 0 \quad (3.5.17)$$

$$s = \frac{-(2K_o\Omega_{mo} - T_{\Omega mo})}{J} \quad (3.5.18)$$

The root is real; for the system to be stable s<0. Since J>0, $K_o > 0$, and $\Omega_{mo} > 0$ it is then needed that

$$-2K_o\Omega_{mo} + T_{\Omega mo} < 0 \quad (3.5.19)$$

$$T_{\Omega mo} < 2K_o\Omega_{mo} \quad (3.5.20)$$

So, if $T_{\Omega mo}$ is negative, the plant is stable, if $T_{\Omega mo}$ is positive it should fulfill condition 3.5.20. For the values of the motor used, the system is stable and the pole has its real part negative; the condition is fulfilled.

The plant transfer function is:

$$\omega_m(s) = \frac{-T_{\Omega elo}\,\omega_e(s) + \Omega_{mo}\,k(s)}{Js + 2K_o\Omega_{mo} - T_{\Omega mo}} \quad (3.5.21)$$

That obtains the single, stable real dominant pole determined by the shaft inertia and the speed propeller loading similar to the analysis of Diesel engines (Xiros, 2004).

Speed regulation through feedback is required if the induction motor is operating near its synchronous speed when significant propeller torque coefficient fluctuation occurs.

3.5.2. Pi Design

Today's merchant ship Diesel propulsion plants are controlled by PI speed (rpm) regulators (Xiros, 2004). Usually D gain is set to zero to avoid the effects of the noise being amplified through the derivative gain (sometimes a filter is required) and to reduce complexity of the system. The same approach can be used for induction motors as a starting point. The standard form of a PID control law is:

$$y(t) = K_P n(t) + K_I \int n(t)dt + K_D \dot{n}(t) \underset{L}{\Leftrightarrow} G_C(s)\omega_m(s) = \left(K_P + \frac{K_I}{s} + K_D s\right)\omega_m(s) \quad (3.5.22)$$

Incorporating this into the plant transfer function, the following closed-loop transfer function from $\omega_e(s)$ to $\omega_m(s)$ is obtained:

$$G(s) = \frac{\omega_m(s)}{k(s)} = \frac{\Omega_{mo}\, S}{(J + T_{\Omega elo}\, K_D)s^2 + (2K_o\Omega_{mo} - T_{\Omega mo} + T_{\Omega elo}\, K_P)s + T_{\Omega elo}\, K_I} \quad (3.5.23)$$

For the closed-loop system to be stable the denominator of 3.5.23 must have all its roots < 0, this is, their real part must be < 0. With this motor values the roots are (the gains are obtained in next section, equations 3.5.41, 3.5.42 and 3.5.43):

$s_1 = -0.55 - 0.0000000079i$

$s_2 = -0.55 + 0.0000000079i$

It has a complex pole located in the left-half plane on the real-imaginary axis, it is stable then.

For design purposes, the following standard form of second order transfer function will be used (Kuo, 1987):

$$G(s) = \frac{A{\omega_n}^2 s}{s^2 + 2\zeta\omega_n s + {\omega_n}^2} \quad (3.5.24)$$

where the following design equations determine the relationship between the closed-loop transfer function parameters and the gains of the PID control law:

$${A\omega_n}^2 = \frac{\Omega_{mo}}{(J + T_{\Omega elo} K_D)} \quad (3.5.25)$$

$$2\zeta\omega_n = \frac{(2K_o\Omega_{mo} - T_{\Omega mo} + T_{\Omega elo} K_P)}{(J + T_{\Omega elo} K_D)} \quad (3.5.26)$$

$${\omega_n}^2 = \frac{T_{\Omega elo} K_I}{(J + T_{\Omega elo} K_D)} \quad (3.5.27)$$

Determining gain a, damping factor ζ and natural underdamped frequency ω_n of closed-loop transfer function result in the PID controller gains (K_P, K_I, K_D).

3.5.3. H∞ PID Speed Regulator Synthesis

The design specification for the controller is that the H_∞-norm of the transfer function of the compensated plant G(s):

$$\|G(s)\|_\infty = \sup_{\omega \geq 0} |G(j\omega)| \quad (3.5.28)$$

is equal or smaller to a bound provided that stability is maintained (Tempo, 1996). Is easy to see that, in general, $\|G(s)\|_\infty$ is a decreasing function with damping factor ζ.

It is required that the closed-loop system does not present oscillatory response. This requirement is achieved by assigning values to the damping factor larger or equal to unity, i.e. $\zeta \geq 1$ (Kuo, 1987). Robustness of design is simpler to achieve if $\zeta = 1$. When $\zeta > 1$ one has to compromise stability (as slow closed-loop pole s_1 moves closer to the right-half plane) and, robustness against neglected dynamics (as fast closed-loop pole s_2 moves closer to the region of neglected open-loop poles) (Xiros, 2004).

$$s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1} \quad (3.5.29)$$

For $\zeta > 1$ holds:

$$s_1 = -\zeta\omega_n + \omega_n\sqrt{\zeta^2 - 1} > \omega_n > -\zeta\omega_n - \omega_n\sqrt{\zeta^2 - 1} = s_2 \quad (3.5.30)$$

If $\zeta = 1$ a double real pole is obtained $s_{1,2} = -\omega_n$.

An appropriate selection of the natural underdamped frequency can maintain the above conditions. Furthermore, the peak of the magnitude Bode plot occurs at $\omega = \omega_n$ (Xiros, 2004).

The value of the H∞-norm of G(s), assuming $\zeta = 1$, is:

$$\|G(s)\|_\infty = \frac{A\omega_n}{2\zeta} = \frac{A\omega_n}{2} \quad (3.5.31)$$

which can be used for tuning a PI or a PID controller.

Considering a fluctuation of 80% of the load (extreme conditions, load assumed to be 11.9 N.m) and a maximum allowed change of 1% of the nominal speed (17 rpm) when PI acts (see section 3.5.5), the upper bound value for $\|G(s)\|_\infty$ is set to:

$$20\log_{10}(G_o) = 20\log_{10}\left(\frac{17\ rpm}{0.0095\ kNm/rpm\ 2}\right) = 65.05\ dB \quad (3.5.32)$$

1% may be a too tight limit so an error of 3% (51 rpm) is acceptable but the calculations are going to be done with 1% as basis.

3.5.4 Hinf PI

Based on the requirement for $\|G(s)\|_\infty$ the setting is calculated.

$$\|G(s)\|_\infty = 20\log_{10}\left(\frac{A\omega_n}{2}\right) = 65.05dB \quad (3.5.33)$$

As no D-term is considered, the following holds for A gain and underdamped frequency ω_n.

$$A{\omega_n}^2 = \frac{\Omega_{mo}}{J} \quad (3.5.34)$$

Solving equations 3.5.33 and 3.5.34 yields to:

$$\omega_n = \frac{\Omega_{mo}}{2J10^{3.25}} \quad (3.5.35)$$

$$A = \frac{\Omega_{mo}}{J {\omega_n}^2} \quad (3.5.36)$$

Therefore

$$\omega_n = 0.5589 \text{ rad} \quad (3.5.37)$$

$$A = 6.4036e + 3 \quad (3.5.38)$$

The values of the PI gains are obtained from equations 3.5.26 and 3.5.27:

$$K_P = \frac{2J\omega_n - 2K_o\Omega_{mo} + T_{\Omega mo}}{T_{\Omega elo}} \quad (3.5.39)$$

$$K_I = \frac{{\omega_n}^2 J}{T_{\Omega elo}} \quad (3.5.40)$$

$$K_P = 0.0106 \quad (3.5.41)$$

$$K_I = 0.1959 \quad (3.5.42)$$

$$K_D = 0 \quad (3.5.43)$$

The proportional gain is the same than the one obtained by manual tuning; larger proportional gains drive the system to instability. However the integral gain is smaller than the one obtained manually (see section 4.8.1). A larger Integral gain is possible:

$$K_P = 0.0106 \quad (3.5.44)$$

$$K_I = 4.5 \quad (3.5.45)$$

$$K_D = 0 \quad (3.5.46)$$

Simulations with 3.5.44 and 3.5.45 gains are shown in section 4.8.2. It is seen that the error is reduce and stability is not compromised, the poles of the closed loop system with these new gains are still stable.

$s_1 = -0.5589 + 2.6201\mathrm{i}$

$s_2 = -0.5589 - 2.6201\mathrm{i}$

the maximum error for these gains is 24 rpm which corresponds to 1.4% of the nominal speed.

3.5.5 Computing Upper Limit for Pi Hinf

Equation 3.5.32 is obtained based on (Xiros, 2004). The numerator of the logarithm is the limit of rpm allowed to differ from the set point once the disturbance and the PI acts. It was selected to be 1% of the nominal speed; if simulations with manual calibration of PI are checked, they show that the speed oscillates less than 17 rpm when both disturbances and PI are present.

The denominator is the maximum change of load allowed; this is the maximum value of the disturbances. It was selected to be 80% of the nominal load (11.9 N.m) which is 9.5 N.m.

3.6. VECTOR FIELD CONTROL

Another strategy to control an induction motor is Vector Field control. Usually is very useful for vector control to choose the rotating d-axis to be the angle of the rotor flux linkage. This is

$$\lambda_r = -j\lambda d_r \quad (3.6.1)$$

or, in other words,

$$\lambda q_r = 0 \text{ and } \lambda \dot{q}_r = 0 \quad (3.6.2)$$

Equations 3.1.73-3.1.75, 3.1.77 1nd 3.1.78 change as following:

$$T_e = (1.5)PL_o[-\lambda d_r \lambda q_s] \quad (3.6.3)$$

$$\lambda \dot{q}_s = Vq_s + R_s[L_c \lambda q_s] - \omega_s \lambda d_s \quad (3.6.4)$$

$$\lambda \dot{d}_s = Vd_s - R_s[L_o \lambda d_r - L_c \lambda d_s] + \omega_s \lambda q_s \quad (3.6.5)$$

$$0 = -R_r[L_o \lambda q_s] - (\omega_s - \omega_r)\lambda d_r \quad (3.6.6)$$

$$\lambda \dot{d}_r = -R_r[L_o \lambda d_s - L_c \lambda d_r] \quad (3.6.7)$$

$$\dot{\omega}_m = \frac{1}{J}(T_e - F\omega_m - T_l) \quad (3.6.8)$$

From equation 3.6.6 a relation for $(\omega_s - \omega_r) = \omega_{slip}$ can be found.

$$(\omega_s - \omega_r) = -R_r L_o \, {}^{\lambda q_s}/_{\lambda d_r} \quad (3.6.9)$$

There are now 6 equations and 6 unknowns, but one of the unknowns is ω_s. It is possible to find the corresponding synchronous speed (and therefore the input frequency) for certain value of torque.

The block diagram can be seen on figure 3.6.1. It's similar than figure 3.1.11 except that the angle Thetae is not given but found. This kind of control requires

a good accuracy of the rotor flux angle. There are two methods to determine the rotor flux angle. The first one called Indirect Vector Control (IVC) calculates it from equations 3.6.9 and

$$\theta_e = \theta_r + \int(\omega_s - \omega_r)dt \Rightarrow \theta_e = \theta_s \qquad (3.6.10)$$

easily deduced from section 3.1.9. For IVC the knowledge of the internal parameters of the motor (Lm, Lr, etc.) is fundamental; since these parameters change due to operating conditions some error may appear. The second method is called Direct Vector Control (DVC) and determines $\boldsymbol{\theta_e}$ either from the measurement of air-gap flux, or from terminal voltages and currents. For DVC the angle can be found using:

$$\lambda_s = \int(V_s - R_s I_s)dt$$

$$\lambda_r = \frac{L_m}{L_r}\left[\lambda_s + \left(\frac{L_s - L_m{}^2}{L_r}\right) I_s\right] \qquad (3.6.11)$$

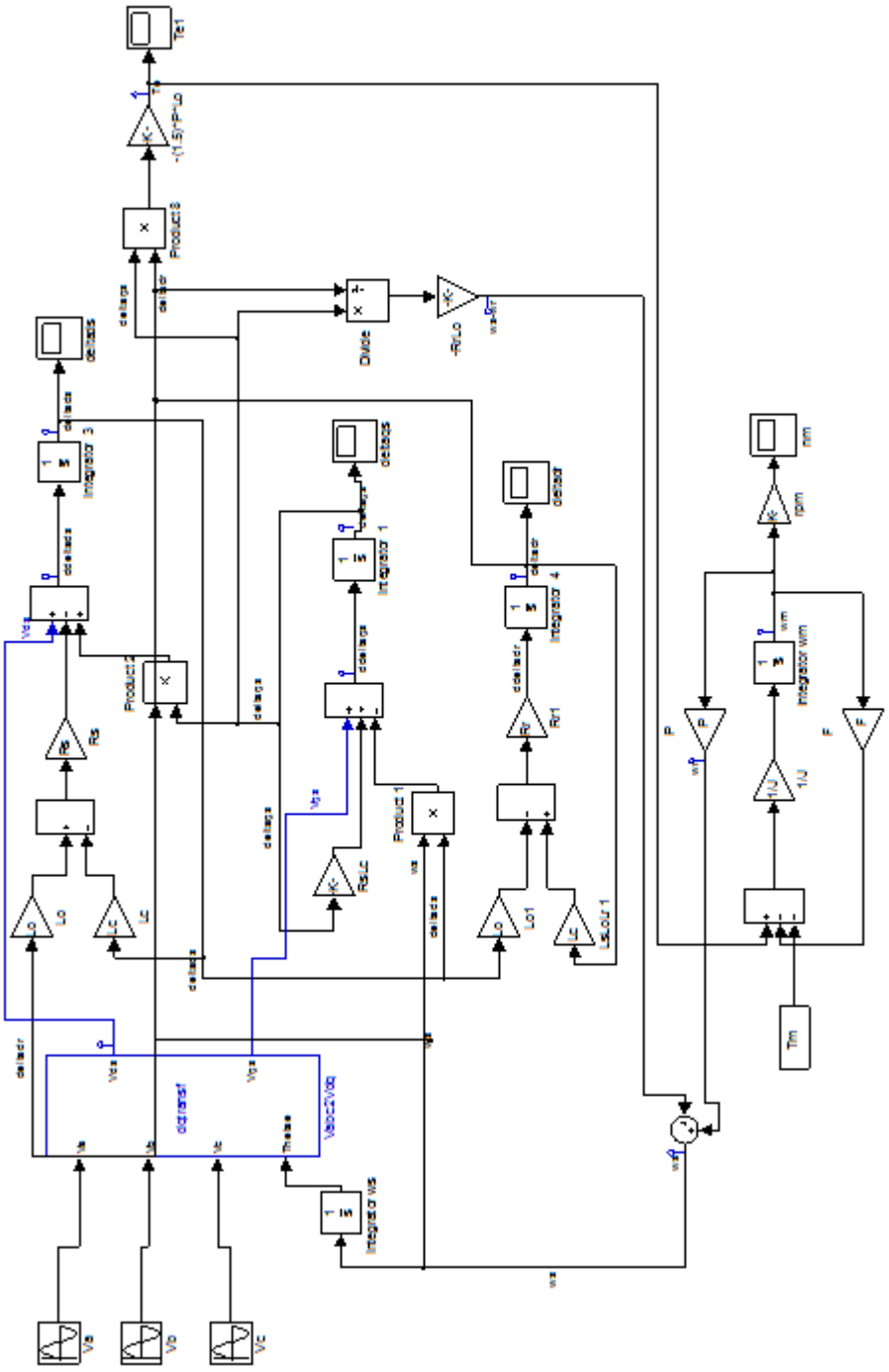

Figure 3.6.1: dynamic block diagram of an induction motor in d-q plane with vector field control

DVC is relatively insensitive to variations of rotor parameters; however performance of DVC may be sluggish at low speed due to inaccurate knowledge

of stator resistance, integration drift, etc (Ohm, 2000). For purposes of this project DVC is not used; the models developed were done using the IVC approach. For further information refer to (De Doncker, 1989).

3.7. COMPREHENSIVE MODEL

So far the project has a d-q model to predict the behavior of an induction motor (section 3.1.9), an analysis of steady state and disturbances (sections 3.4 and 3.5), a PI controller (section 3.5.2-3.5.4), and a vector field control model (section 3.6).

The PI controller assumes that the system already reached steady state (simulation section 4.6) and that only the disturbances are affecting the system; this may be true in some cases but one of the advantages of electric motors is that they can change their speed easily, once the PI is activated if the reference speed of the motor is changed the PI will see this change as a big perturbation and will affect the output; therefore a method to avoid this is needed.

In order to change the reference speed an open loop control is created (section 4.5) with a lookup table.

If the d-q model is integrated with the PI and the lookup table a system capable to deal with perturbations and changes of speed is possible. Block diagram of figure 3.6.1 shows such system.

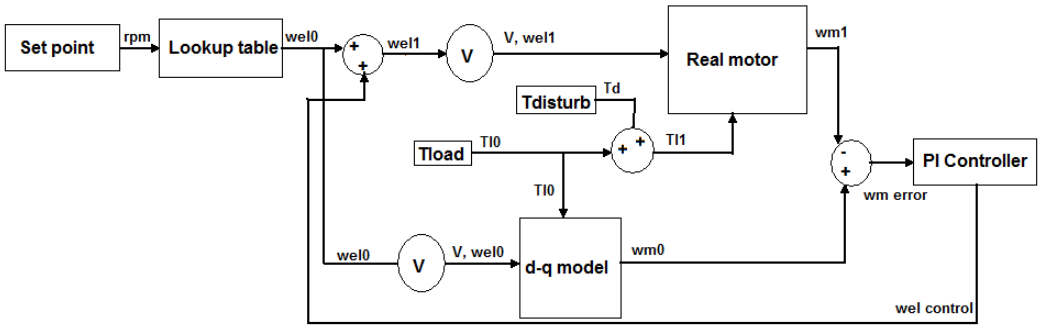

Figure 3.7.1: block diagram comprehensive model

The block diagram shows that when no perturbations are present, the real motor and the d-q model have the same behavior so the input for the PI controller is zero; this applies also when the reference speed changes. When a disturbance appears (Td) the difference between the speed of the real motor (wm1) and the speed predicted by the d-q model (wm0) acts as input for the PI controller; the PI controller responds with a frequency signal (wel control) that is added to the frequency input signal of the real motor (wel1) thus correcting its speed.

A different plant model and a more complex control system may be implemented using Vector Field Control (VFC) (section 3.6 for preliminary simulations). This approach has the advantage that 2 different outputs can be controlled with the same input. The control for this system goes beyond the scope of this work and is left as future work; however, the plant model obtained with VFC is simulated and tested.

4. ANALYSIS AND SIMULATIONS

Based on the state of art, the models developed and Matlab own model, a series of simulations can be done to analyze the viability of the system and get a first approach to its control.

First the Matlab asynchronous machine model is tested (section 4.1 and 4.2); then an open loop control (look up table) is applied directly to the Matlab block (section 4.5); after that, a PI controller tuned using Ziegler & Nichols is added (section 4.3 and 4.6).

Once checked that the Matlab block behaves as a real motor and that PI and open loop control are possible, the d-q model is added (section 4.7 and 4.8). The Vector Field Control is also tested in section 4.9.

In section 4.10 the test bench that is used to physically test the analysis is described.

4.1. MATLAB SIMULATION OF AN INDUCTION MACHINE

Using the a D-Q model a little more complicated than in section 3.1.6, Matlab/Simulink has a block that represents an induction motor called “Asynchronous machine” (Matlab, 2008). Referenced to the stator and setting the proper parameter a 3-phase AC motor can be simulated.

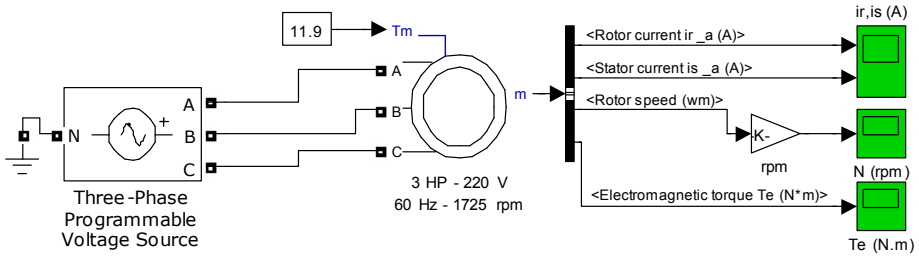

Figure 4.1.1: 3-phase synchronous machine

The system is connected to a 3-phase voltage input, Y connection, 220V, 60 Hz. The motor is set as a 3 HP, 220 V, 60Hz, 1725 rpm, with the values in table 4.1.1 and the stator as reference frame. These values are arbitrary values since the real values of the motor that is going to be simulated are unknown at this stage (see section 4.7); however the values were selected based on the closest predefined Matlab standard motor. The output was set to be rotor speed (units rpm, using the rpm block seen in the figure that multiplies the rotor speed *wm* by $30/\pi$, this block is for now on used in all the simulations) , rotor torque (N.M) and rotor angle (rad).

Parameter	Value
Load	11.9 N.m
Nominal power	3HP/ 2238 VA
Nominal Voltage	220 Vrms
Rs	0.435 ohms
Lls	4 mH
Lm	69.31 mH
Rr	0.816 ohms
Llr	2 mH
Pole pairs	2
Inertia (J)	0.089 kgm^2
Friction coefficient (F)	0

Table 4.1.1: AC motor parameters

The load (TM) appears as an input in the Matlab block but is really a fixed value defined by the propeller and the tests made in calm water, it is the mechanical load torque; the load was fixed arbitrarily at 11.9 N.m, that value cannot be change except to simulate disturbances (see section 4.6). Figure 4.1.2 shows the results for a lapse of 1 second.

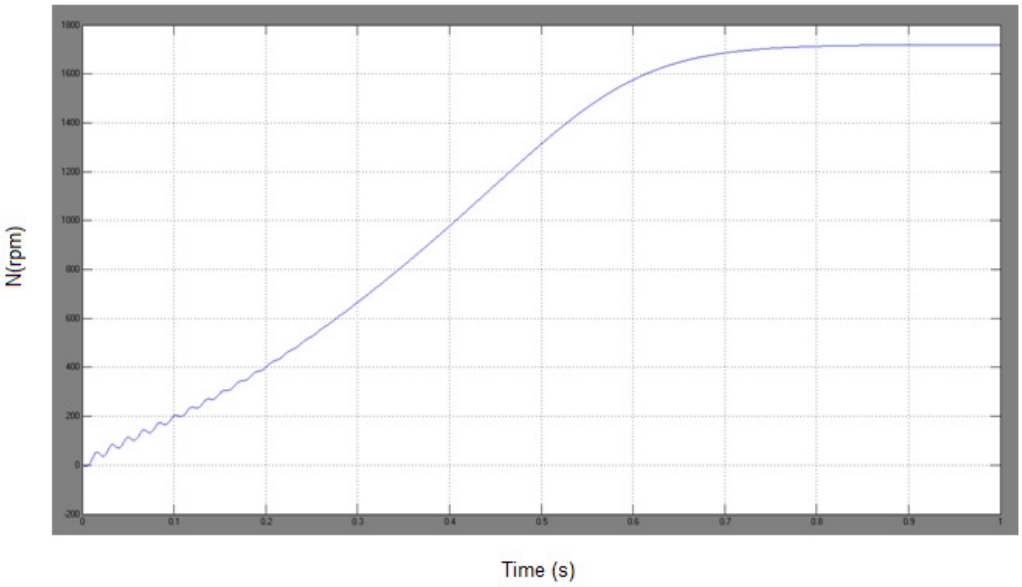

Figure 4.1.2: speed vs. time

The graph shows the speed going from 0 to 1725 rpm (1.0 pu) which is the theoretical value of the machine.

The torque graph (figure 4.1.3) shows the electromagnetic torque developed by the machine that also agrees with the theory, i.e. the oscillation at the beginning is due to the transient of the motor that is not linear (hence the oscillation), then the torque tends to stabilize at its final value while the speed dies the same. The

current (figure 4.1.4) is also oscillatory at the beginning and stabilizes when the motor reaches its final speed.

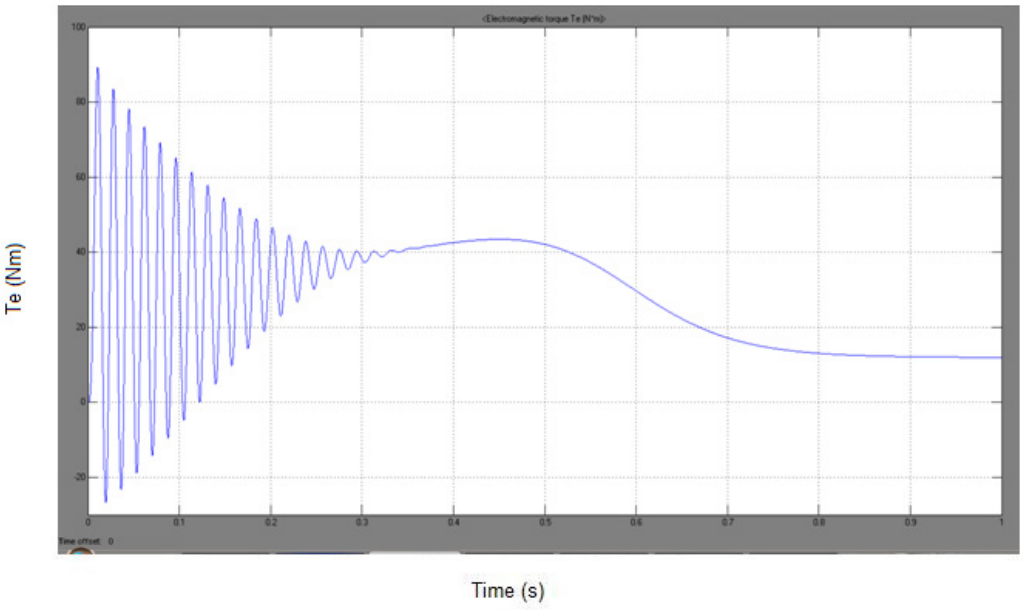

Figure 4.1.3: torque vs. time

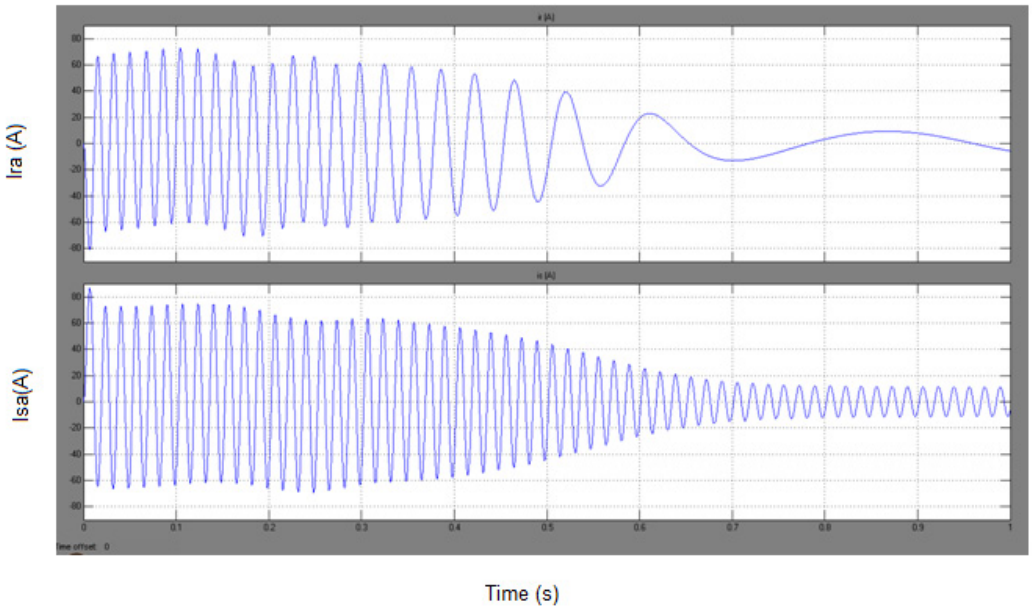

Figure 4.1.4: stator and rotor currents vs. time

4.2. TESTING AC ASYNCHRONOUS MACHINE BLOCK OF MATLAB

The block AC asynchronous machine of Matlab allows changing any parameter of the equivalent circuit of an asynchronous machine (see 3.1.3.). The block was also tested in order to obtain its torque vs. speed graphic; this graphic, shown in figure 4.2.2, was obtained by varying the load mechanical torque of the motor and measuring its resultant speed and electromagnetic torque. It can be seen, in figure 4.2.3, that the response agrees with the theoretical response of an AC asynchronous motor (see reference (Chapman, 2005), chapter 7, page 410). Figure 4.2.1 shows the configuration used to test the block. According to the Matlab help documents when a negative load mechanical torque (Tm) is applied to the AC motor it behaves as a generator; the sine wave applied on Tm varies from 100 to -100 N.m allowing the AC asynchronous machine block to behave as motor and as generator; then the results of both scopes, electromagnetic torque and speed, were exported to Matlab command window as 2 different vectors, since the vectors correspond each other point to point by plotting them the torque vs. speed graph could be obtained (figure 4.2.2).

In figure 4.2.2 it can be seen the response of the AC motor Matlab block when a sinusoidal response is applied as a disturbance in its mechanical load. Two main zones can be identified, the transient zone and the steady-state zone; the transient is due to the non linearity and typical characteristic of the startup on an AC motor, it can be seen an oscillation and certain instability. The steady-state zone corresponds to the stable characteristic of the motor, this is, when the

motor has reached it final speed, the figure described coincides with the theory and is extracted in figure 4.2.3.

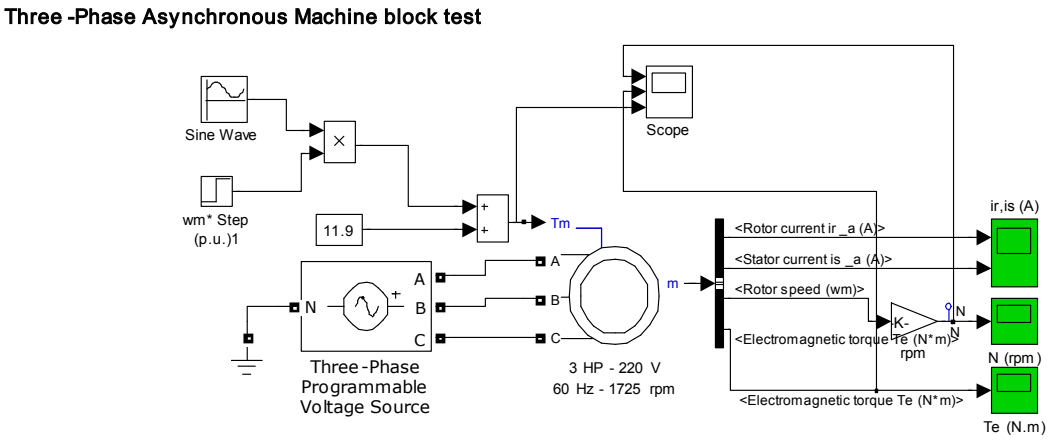

Figure 4.2.1: testing Matlab block setup

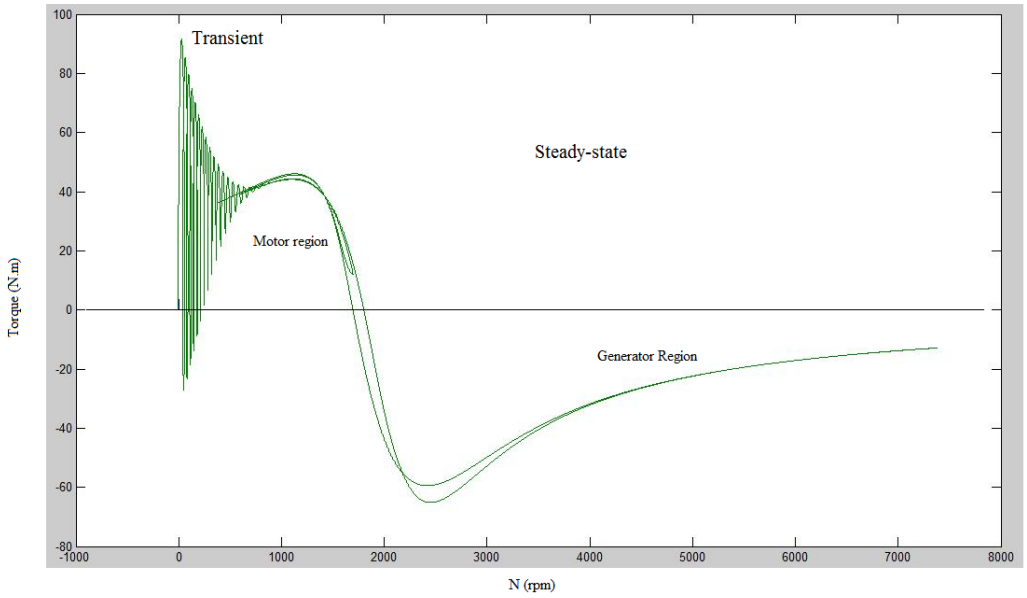

Figure 4.2.2: torque vs. N, unstable region shown

As can be seen in figure 4.2.3, the Matlab block behaves as a real AC motor; this is, for speeds above its limit, the electromagnetic torque falls; if the speed is increased further (or the mechanical load takes negative values) the

electromagnetic torque becomes negative, the motor starts to act as a generator; this is consistent with the real behavior of an AC motor.

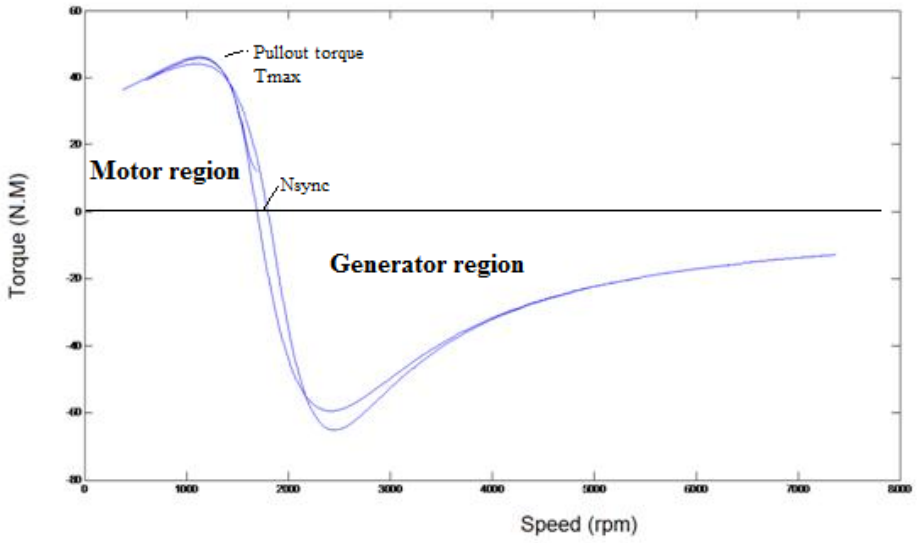

Figure 4.2.3: torque vs. speed for Matlab AC motor block

As can be seen in figure 4.2.3, the Matlab block behaves as a real AC motor; this is, for speeds above its limit, the electromagnetic torque falls; if the speed is increased further (or the mechanical load takes negative values) the electromagnetic torque becomes negative, the motor starts to act as a generator; this is consistent with the real behavior of an AC motor.

The induce torque of the motor is zero at synchronous speed (1800 rpm); also, there is a maximum torque that cannot be exceed (48 N.m aprox.), this is the pullout torque. If the rotor is driven faster than the synchronous speed, then the direction of the induced torque changes and the machine becomes a generator (converts mechanical power to electric power) (Chapman, 2005).

In the configuration of the figure 4.2.1 the AC motor block was set to have an speed limit of 1800 rpm so it is coherent that for higher speeds the electromagnetic torque falls; a perfect graph couldn't be observe due to the effects of the transient and the configuration of the test but it is not important since the objective of the test was to observe the behavior not to obtain accurate values.

4.3. CALIBRATION OF A CONTROLLER

4.3.1. PID Controller

The PID controller is probably the most-used feedback control design. "PID" stands for Proportional-Integral-Derivative, referring to the three terms operating on the error signal to produce a control signal. The general form of a PID controller is:

$$u(t) = K_P e(t) + K_I \int e(t)\mathrm{d}t + K_D \frac{\mathrm{d}}{\mathrm{d}t} e(t). \qquad (4.3.1)$$

where *u(t)* is the control signal sent to the system, *y(t)* is the measured output, *r(t)* is the desired output, and tracking error $e(t) = r(t) - y(t)$. The desired closed loop dynamic is obtained by adjusting the 3 parameters K_P, K_I and K_D, often iteratively by "tuning" and without specific knowledge of a plant model.

The proportional term usually ensures the stability and provides the main gain for the controller. The integral term helps to smooth (or increase depending on its value) any overshoot and allows the controller to reduce its error even to zero.

The derivative term provides damping or shapes the resulting signal, sometimes it also “speeds up” the controller helping it to reach the stable point faster. Theoretically a PI controller only can have zero error; but sometimes in practice the time to reach that value could be very large and a derivative term is required (Wikipedia, PID controller).

The equation of a PID controller in Laplace domains goes:

$$u(s) = K_P e(s) + K_I \frac{1}{s} e(s) + K_D s e(s)$$

$$u(s) = (K_P + K_I \frac{1}{s} + K_D s) e(s) \qquad (4.3.2)$$

The PID controller function is then derived.

$$C(s) = (K_P + K_I \frac{1}{s} + K_D s). \qquad (4.3.3)$$

PID equation in Laplace domain is particularly useful for simulations or to obtain a transfer function of the entire system (Wikipedia, PID controller).

For this project a PI was thought to be enough, the reason is that the inertia of the propeller already damps the motor so there is no need of an additional term that complicates the tuning (see section 4.3.2). The equation 4.3.2 is then used but without the derivative term.

$$\boldsymbol{u(s) = K_p e(s) + K_i \frac{1}{s}(s)} \qquad (4.3.4)$$

Considering other elements explained in further sections a conceptual controller system can be expressed (figure 4.3.1). It's constituted by 3 basic blocks: the input voltage, where the controller applies its effect; the AC motor, the block to control; and the PI controller, responsible to handle the disturbances.

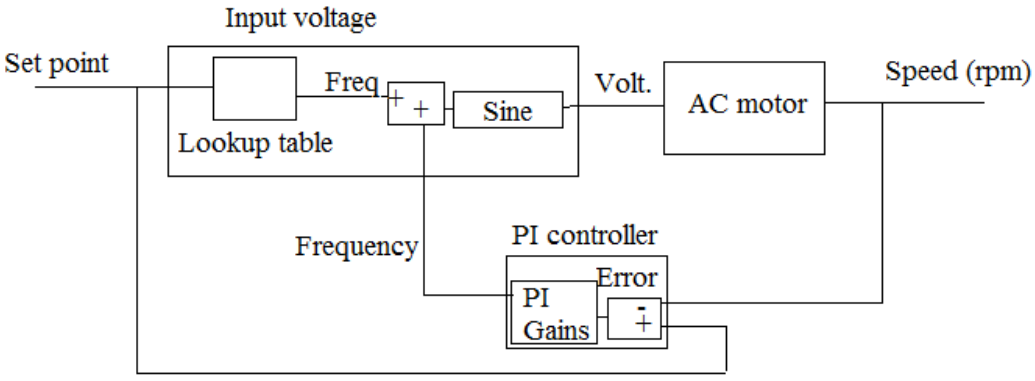

Figure 4.3.1: conceptual controller system without plant model, simple PI

4.3.2. Methods of Calibration

Manual Tuning

If the system must remain online, one tuning method is to first set K_i and K_d values to zero. Increase the K_p until the output of the loop oscillates, then the K_p should be set to approximately half of that value for a "quarter amplitude decay" type response. Then increase K_i until any offset is correct in sufficient time for the process. However, too much K_i will cause instability. Finally, increase K_d, if required, until the loop is acceptably quick to reach its reference after a load disturbance. However, too much K_d will cause excessive response and overshoot. A fast PID loop tuning usually overshoots slightly to reach the set

point more quickly; some systems cannot accept overshoot, in which case an "over-damped" closed-loop system is required, which will require a K_p setting significantly less than half that of the K_p setting causing oscillation (Wikipedia, PID controller).

Ziegler–Nichols Method

Another tuning method is formally known as the Ziegler–Nichols method (table 4.3.1), introduced by John G. Ziegler and Nathaniel B. Nichols. As in the manual method, the Ki and Kd gains are first set to zero. The P gain is increased until it reaches the critical gain, Kc, at which the output of the loop starts to oscillate. Kc and the oscillation period Pc are used to set the gains as shown (Wikipedia, PID controller):

Control Type	Kp	Ki	Kd
P	0.5 Kc	-	-
PI	0.45 Kc	1.2 Kp/Pc	-
PID	0.6 Kc	2Kp/Pc	KpPc/8

Table 4.3.1: Ziegler-Nichols calibration parameters

4.3.3. Calibration values for the simulated PI controller of section 4.6

Using Ziegler-Nichols method the following values were obtained:

Kc=0.45, Pc=0.16 s,

Kp=0.2, Ki=0.15

However, those values didn't show an optimal response (although they were very close to the values obtained by manual tuning) and their simulations were not included.

4.4. RECREATING INPUT SIGNAL FOR CONTROL PURPOSES

From equation 3.1.1 it can be seen that only two parameters can be modified in order to control the speed of an AC motor: the number of poles *p* or the frequency of the voltage input signal *f*e. For the purposes of this project the speed control of the motor is going to be achieved by varying the input frequency since the control changing the poles implies opening the motor and rewiring it internally; which is not part of the scope.

In order to change the frequency of the input voltage signal the "three phase programmable voltage source" block used on the previous simulations was replaced by a set of blocks that recreates the 3-phase sine signals, by using a PWM inverter (Mohan, 1995), and allows varying the frequency of it.

In figure 4.4.1 the mathematically created 3-phased signal is shown, the frequency depends on the input labeled "freq. input"; the amplitude is given by the block labeled "amplitude and id set to be 220 V rms; the phases, set by the block "phases" are separated 120 degrees; and the blocks "Va", "Vb" and "Vc" provide the voltage in Y connection. A clock is present since basically it is a PWM.

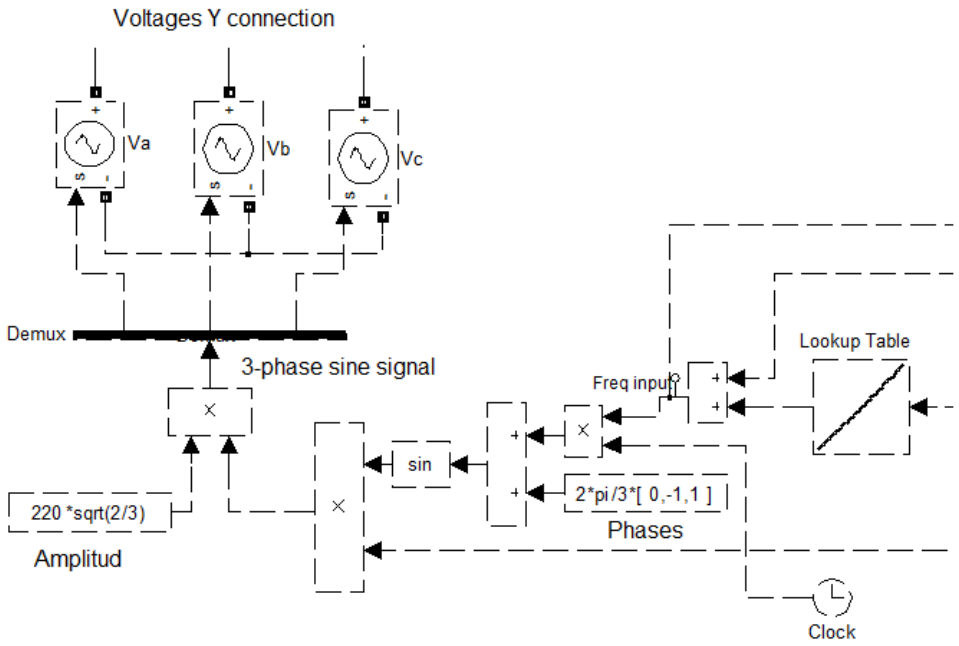

Figure 4.4.1: 3-phase sine signal recreated

The frequency input $\boldsymbol{\theta}$ is added to the phase, and then they go to a sine function; the result is multiplied by the amplitude. This amplitude must be expressed in peak value so for a 3-phase signal the rms value is given by:

$$V_{rms} = \sqrt{\frac{3}{2}} V_p \qquad (4.4.1)$$

therefore the peak value Vp is given by:

$$V_p = \sqrt{\frac{2}{3}} V_{rms} \qquad (4.4.2)$$

hence the value of the amplitude in the figure.

The complete mathematical expression goes:

$$V_{3p} = 220\sqrt{\frac{2}{3}} * sine(\Theta + \phi) \qquad (4.4.3)$$

Of course the ϕ has 3 different values and that is what creates the 3 phases.

The new arrangement acts almost the same than the “three phase programmable voltage source” block except that, having an input for the frequency, this arrangement can vary the frequency of the voltage input and therefore the rotor speed.

4.5. SETTING THE DESIRED SPEED BY USING A LOOKUP TABLE

Now that the input frequency can be modified the rotor speed can be set in any desired value; in practice the range of the rotor speed goes from 100 rpm to 1800 rpm, since the motor behaves very unstable at speeds lower than 100 rpm and cannot go further 1800 rpm due to its power requirements. A lookup table was created (see appendix 8.3) to feed the frequency with the proper value depending on the desired speed (set point SP). These values were obtained empirically by varying the input frequency, this is, setting a specific value and observing the resultant speed in steady state (after 3 seconds). The values can be extrapolated as a lookup table with the proper Matlab block. The lookup table receives the set point (SP) as input, calculates the correspondent frequency and shows it as output, this frequency is the frequency of the 3-phase voltage input signal. The setup of the system is now as seen in figure 4.5.1.

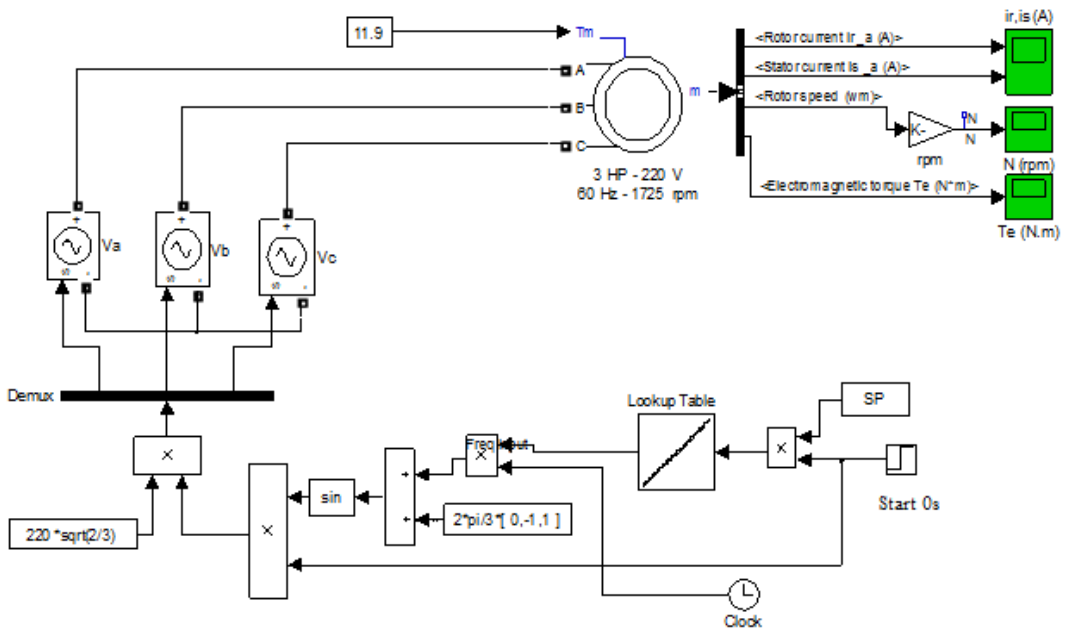

Figure 4.5.1: AC motor with lookup table

Using the lookup table the system reaches the desired speed. In figure 4.5.2 Set Point SP (desired speed) is set on 1000 rpm; for figure 4.5.3 SP=1700.

It can be seen in figures 4.5.1 and 4.5.3 that the rise time is less that 1 second for both cases and, in general, for any other velocity it is never longer than that, so the steady state is assumed after 1 s.

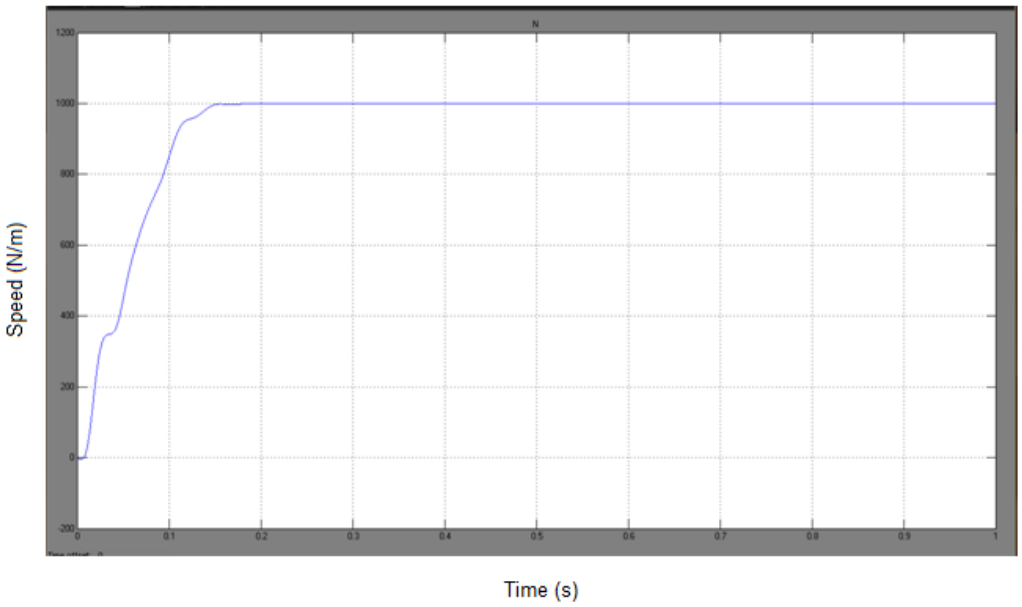

Figure 4.5.2: speed vs. time, SP= 1000 rpm

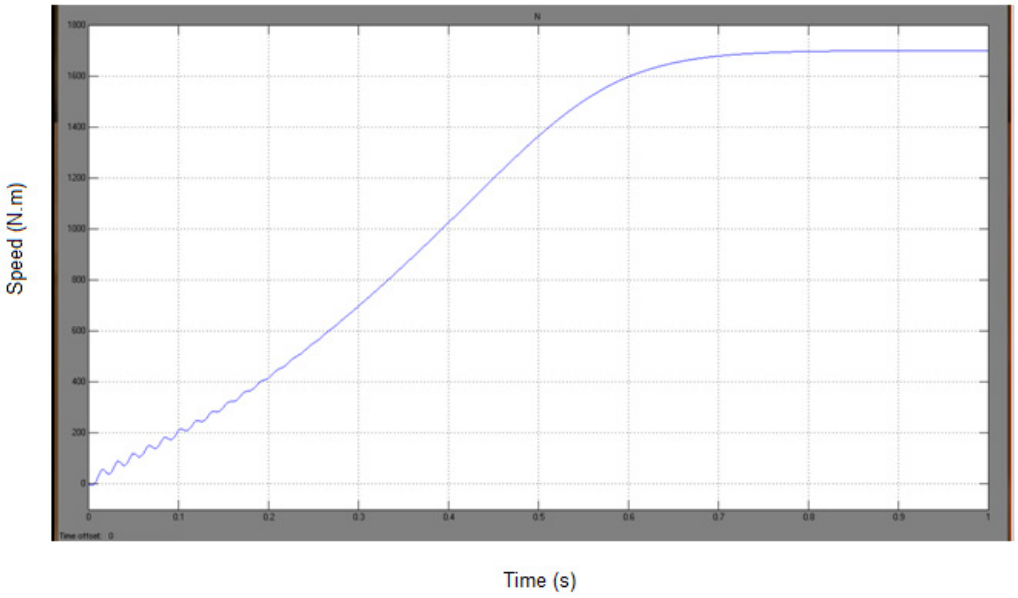

Figure 4.5.3: speed vs. time, SP= 1700 rpm

4.6. PI CONTROLLER

A disturbance of 10% of the nominal load was added to the load mechanical torque (constant disturbance of 1 N.M) and a PI system implemented to deal with it. Since the disturbances are supposed to act in steady state, the disturbance was set to act in t=2 s and the PI controller in t=3 s allowing the graph to show four different states: transient of the AC motor, steady state of the AC motor, disturbance effect and PI controller effect. An attenuation block was added to transform the error; since the error is expressed in rpm, its magnitude is big compared to the magnitude of the frequency; this means that the gains of the PI would be very small. The error was scaled by a factor of 1/100. The setup can be seen in figure 4.6.1.

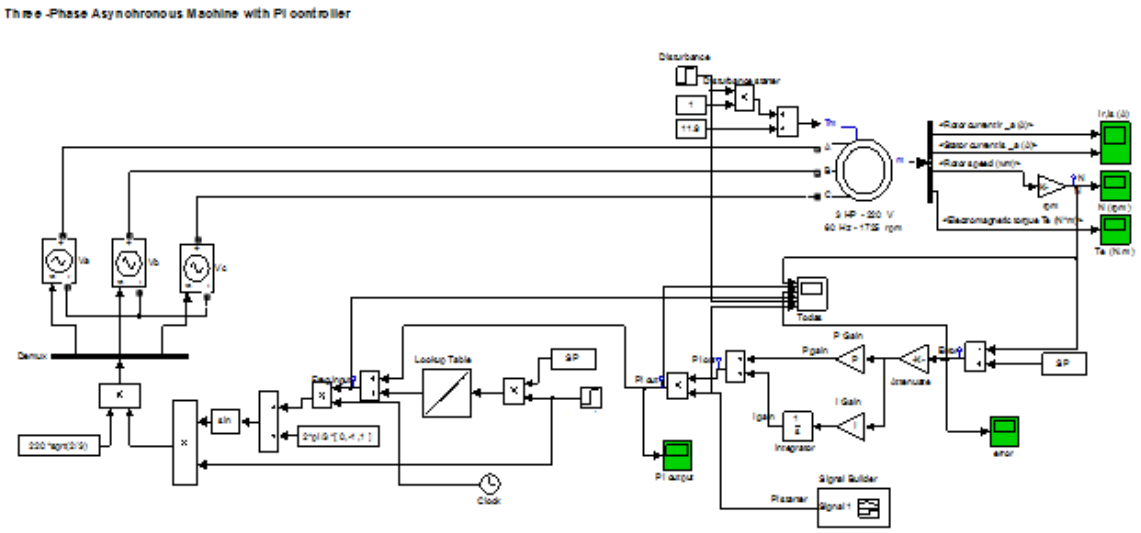

Figure 4.6.1: 3-phase AC motor with PI controller

SP was fixed at 1700 rpm. The gains for the PI controller are P=0.3 and I=0.2 (see section 4.3.3), such gains were obtained by manual calibration. The system

was setup to start the disturbances at 2 seconds and the PI at 3 seconds so the effect of both can be seen. Figure 4.6.2 shows 6 different graphics corresponding to 6 different marks: rotor speed, PI output, frequency input, error (SP-N), disturbance starter (step signal that allows the disturbance to act at 2 s) and, PI starter (step signal that allows the disturbance to act at 3 s).

Figure 4.6.3 shows the behavior of the speed, PI output, freq. input, error, disturbance starter and PI starter. It can be seen how the system behaves depending on the starters activated.

N vs. time is the speed of the rotor for the first 5 second; between 0 and 0.7 seconds is the transient of the motor, the lookup table feeds the proper frequency and the system reaches the desired speed; from 0.7 s to 2 s is the steady state, same than in figure 4.6.2, the system stabilizes at 1700 rpm; at 2s the disturbances are applied and the load is increased in 1 N.m, the speed, therefore, decreases and reaches 1690 rpm; at 3 s the PI starts and an oscillation can be seen, the controller compensates the disturbance effect and the system reaches 1700 rpm again in about 0.4 s. The speed falls 10 rpm when the disturbance actuates and then has an overshoot of 20 rpm and a small oscillation due to the effect of the PI controller.

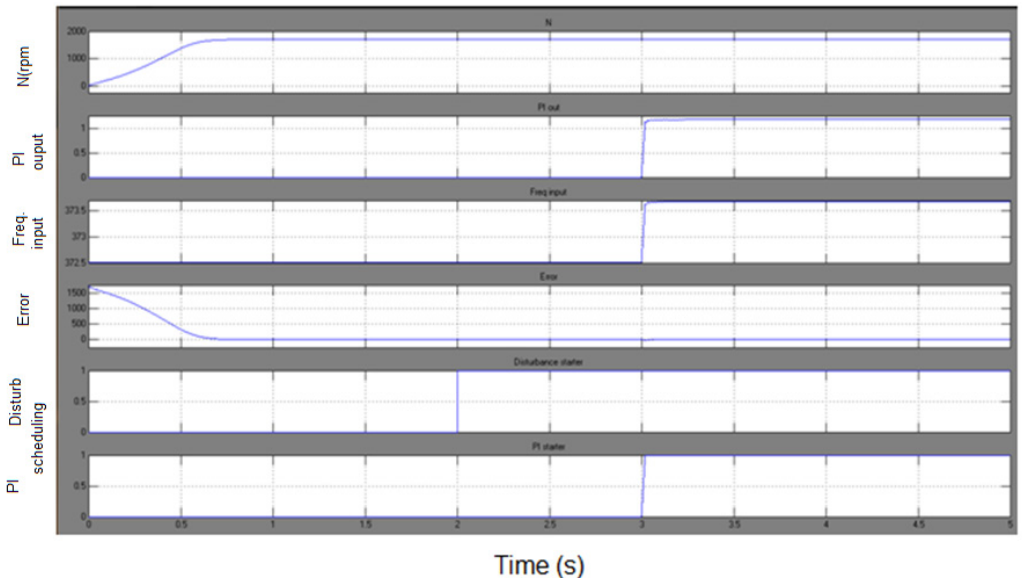

Figure 4.6.2: Speed, PI output, freq input, error, disturbance starter and PI starter vs. time

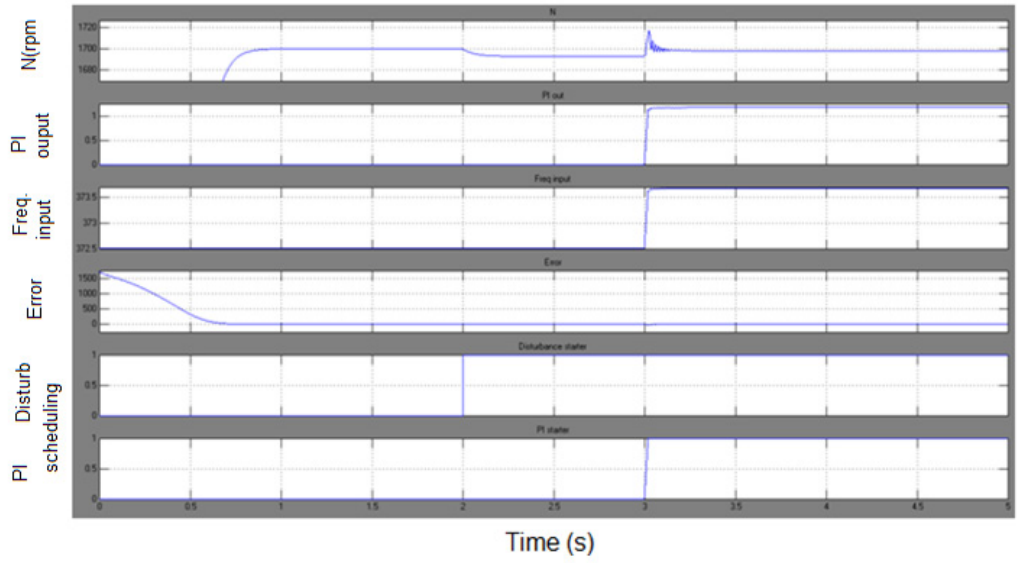

Figure 4.6.3: speed, PI output, freq input, error, disturbance starter and PI starter vs. time (zoomed)

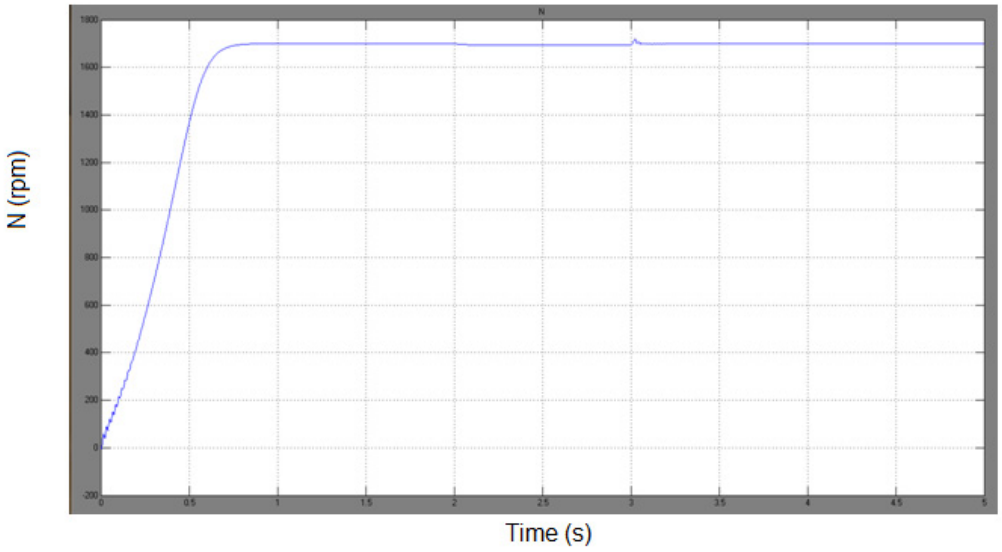

Figure 4.6.4: speed vs. time

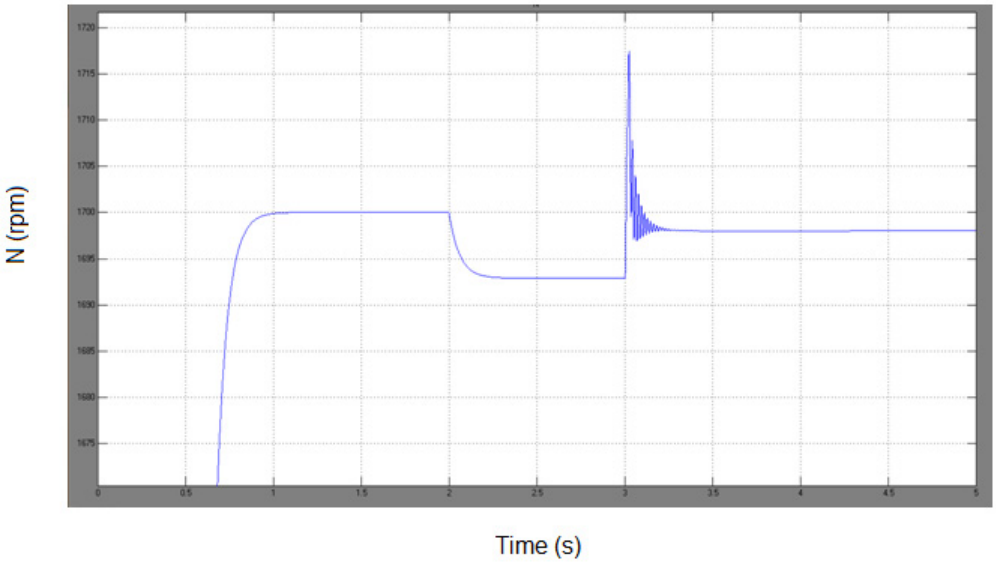

Figure 4.6.5: speed vs. time (zoomed)

The PI was set to start at 3 seconds, the PI starter cannot be a perfect step function since the high frequency components of the step cause instability in the AC motor (Figure 4.6.4 and 4.6.5), and so a very close approximation was built.

When the PI starts an overshoot and an oscillation can be seen, which is consistent with the normal behavior of a PI.

The error (figure 4.6.7 and 4.6.8) reflects the changes of the speed, at the beginning is big since the motor starts from zero but once it reaches the steady state the error goes to zero; when the disturbance actuates the error obviously reflects the difference between the set point and the speed, in this case about 10 rpm; finally when the PI acts the error also oscillates and then falls back to zero.

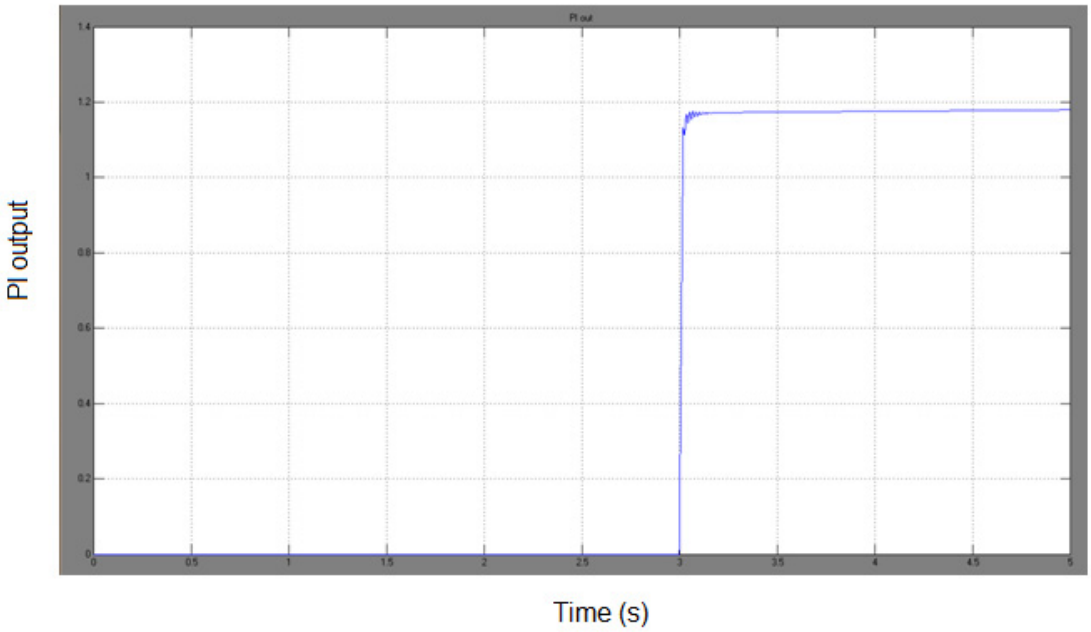

Figure 4.6.6: PI output vs. time

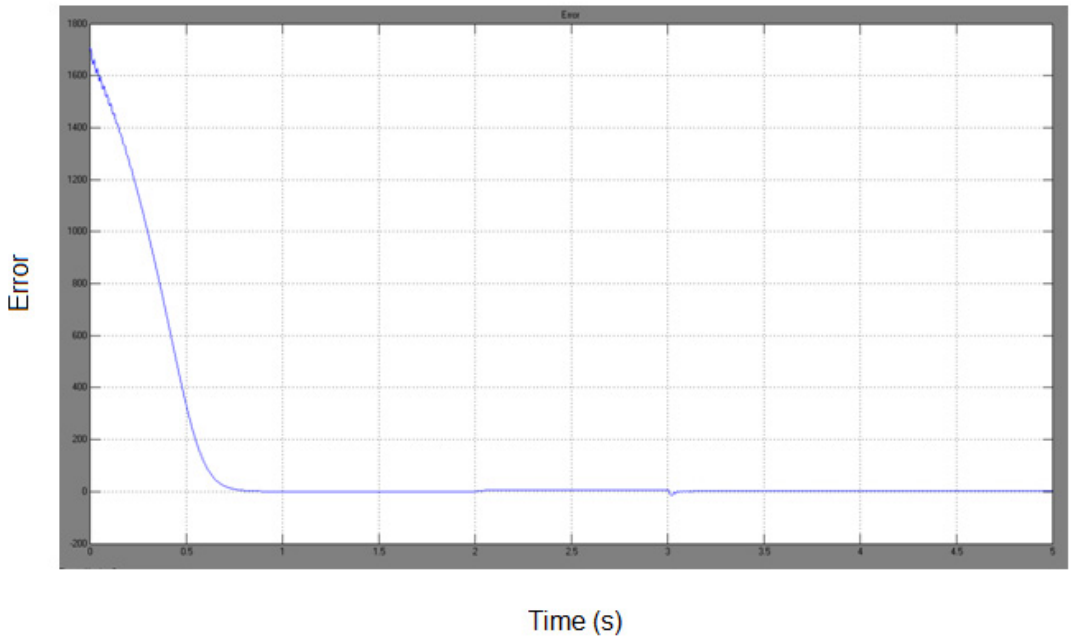

Figure 4.6.7: error vs. time

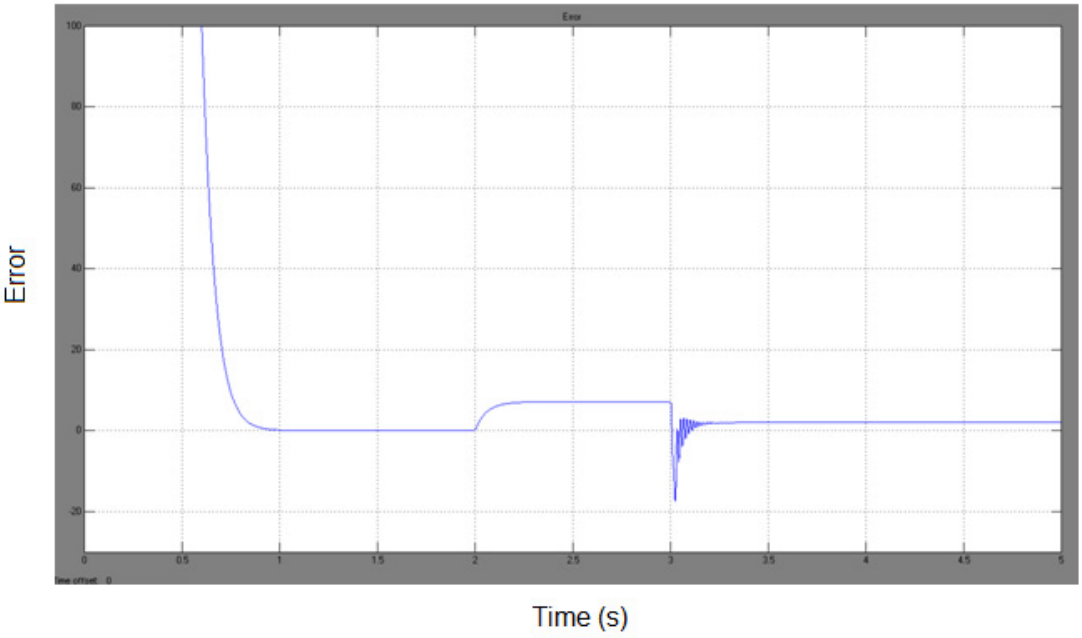

Figure 4.6.8: error vs. time (zoomed)

4.7. INDUCTION MOTOR MODEL USING D-Q PLANE APPROACH

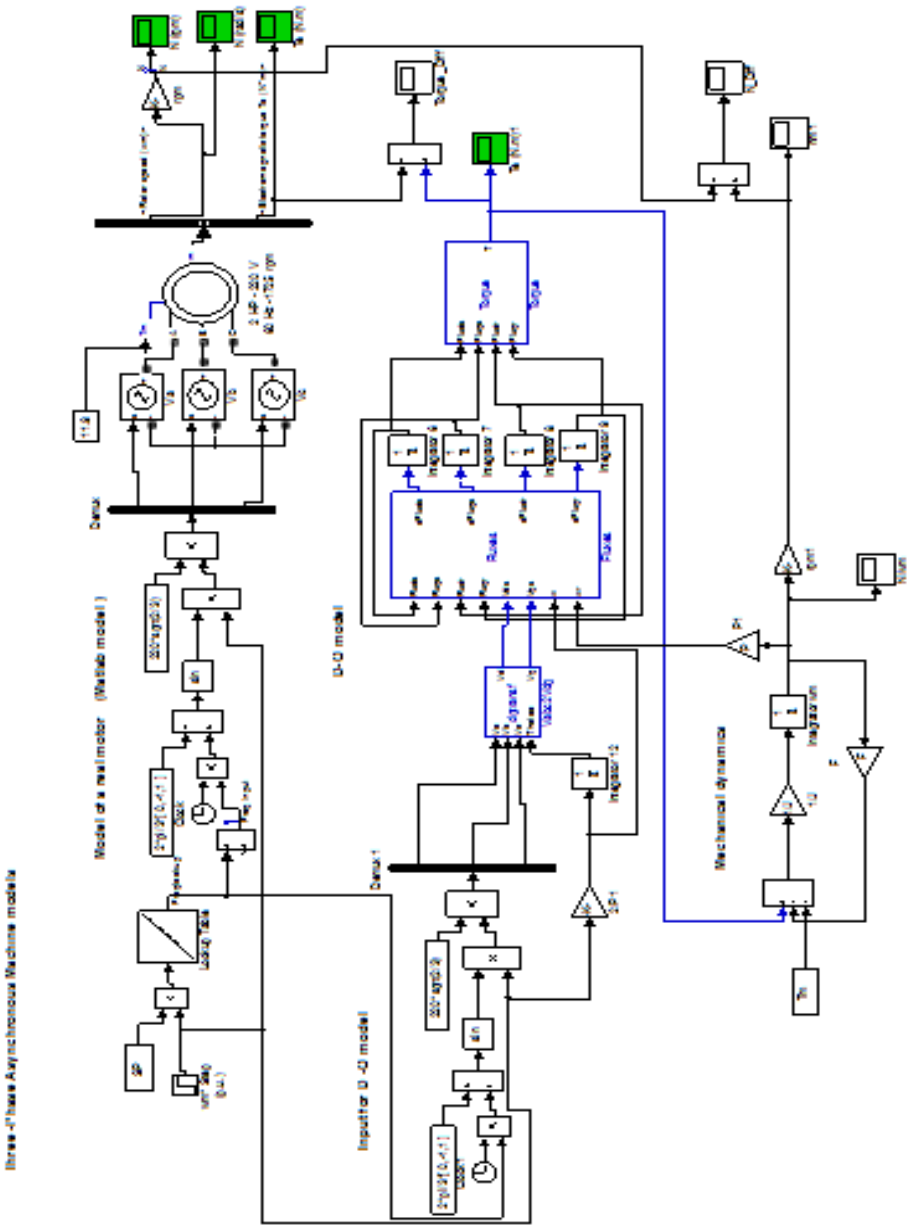

Figure 4.7.1: Simulink schematic, d-q plane model

This system is composed by 3 main parts: the real model, the d-q model and the mechanical dynamics.

The “Real model” is the model used in previous sections that utilizes the asynchronous machine block of Matlab, it simulates the response of a real motor and is added to compare the response of the d-q model with it.

The d-q model has an input similar to the one seen in the real model. The d-q model has 3 embedded blocks: one that transforms the voltage from abc plane to d-q plane (section 3.1.10) called “Vabc2dq”, other that computes the values of each flux (differential equations 3.1.74, 3.1.75, 3.1.77 and 3.1.78; section 3.1.9) called “Fluxes” and, the last one that calculates the torque (equation 3.1.73; section 3.1.9) called “Torque”. A small error is expected since the block that calculates the differential equations uses numerical approximation and it may not follow exactly the real model.

The mechanical dynamics simply used the block diagram derived from equation 3.1.79 and seen in figure 3.1.10.

Parameter	Value
Load	11.9 N.m
Nominal power	2HP
Nominal Voltage	230 Vrms
Rs	0.435 ohms
Lls	4 mH
Lm	69.31 mH
Rr	0.816 ohms
Llr	2 mH
Pole pairs	2
Inertia (J)	0.089 kg m^2
Friction coefficient (F)	0

Table 4.7.1: AC motor parameters

Table 4.7.1 shows the values of the motor used for the following simulations; they are similar to the ones used before except that the power was reduce to 2hp, this doesn't make a significant difference.

4.7.1. Reaching a Defined Set Point of Speed

A first simulation were both models only have to reach a defined set point is shown in figure 4.7.2.

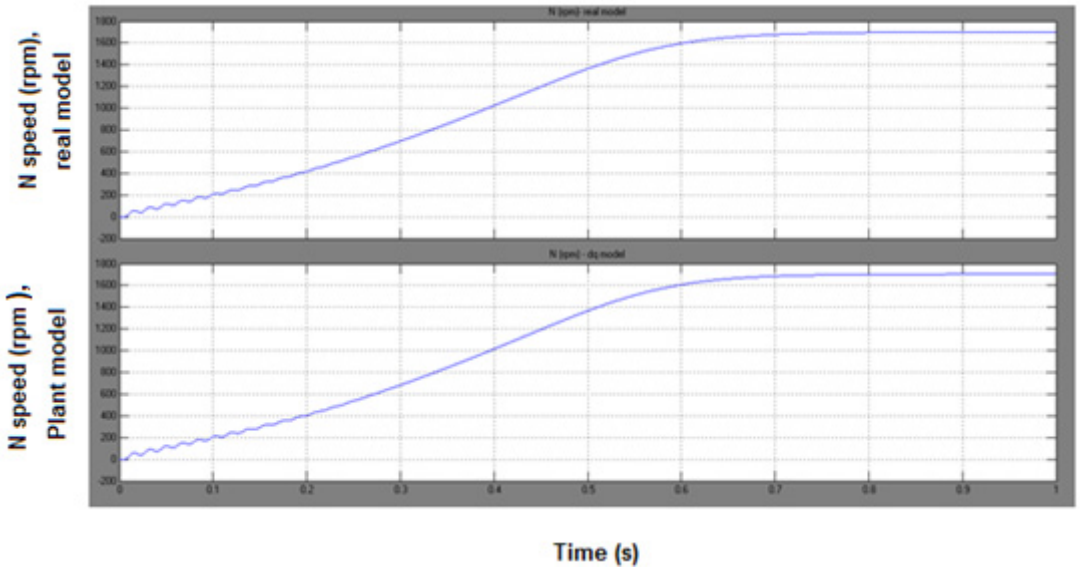

Figure 4.7.2: speeds reaching a set point (rpm), (up) real model, (down) d-q model

It is seen that both models reach 1700 rpm at the same time; both have a transient stage during the first 0.8 second and then stabilize. The d-q model has a slight error of +4 rpm in its steady state.

4.7.2. Transient of Speed

The transient stage for both models is oscillatory (as it was expected from the theory and simulations of previous sections) and the oscillation lasts 0.25 seconds approximately (figure 4.7.3.A); then it raises steadily up to 1700 rpm in 0.8 seconds (figure 4.7.3.b), something also seen in previous simulations. However the difference between both transient stages is relevant and appears as a product of the numerical approximation methods. Even using the same approximation method, the fact that the block Fluxes is a block of itself implied that computations for that block are done in a different time (computational time) than for the other blocks; the real model (Matlab asynchronous block) has the advantage that can do all the computations “at the same time” or in the same space. Although it may seem negligible, this differences accumulating and feeding back, create the small error seen in 4.7.3.

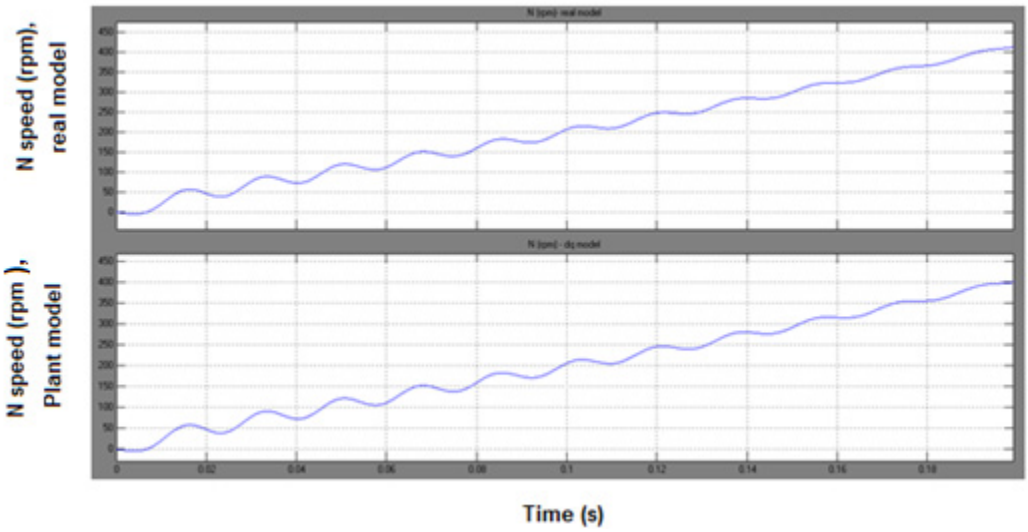

Figure 4.7.3.A: speed transient (rpm) from 0 to 0.2 s, (up) real model, (down) d-q model

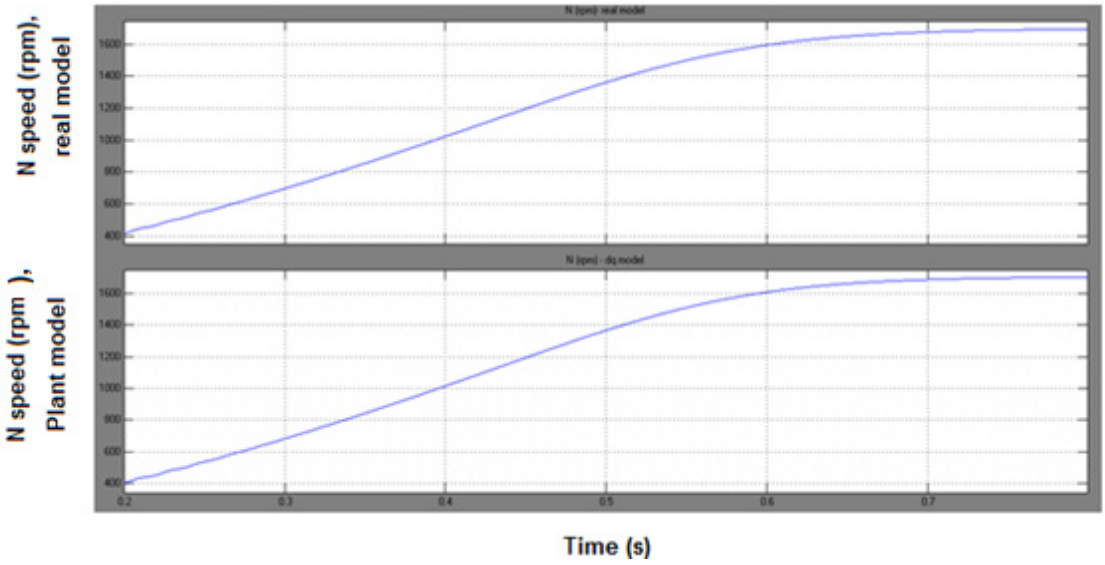

Figure 4.7.3.B: speed transient (rpm) from 0.2 to 0.8 s, (up) real model, (down) d-q model

4.7.3. Error In Speed (rpm)

It was seen (during different trials of the simulation and trying with different entries) that the small error was due the numerical approximation (figure 4.7.4). When the other block was dealt independently extracting their inputs from the real model, the error was negligible, 13 rpm (0.7 % of nominal speed) at its maximum and then stabilizes at 5 rpm (0.29% of nominal speed). The block, therefore, that brings a significant error was the “Fluxes” block; this error cannot be corrected and is innate to the simulating conditions. The error signal was obtained by subtracting the speed of the real model from the speed of the d-q model.

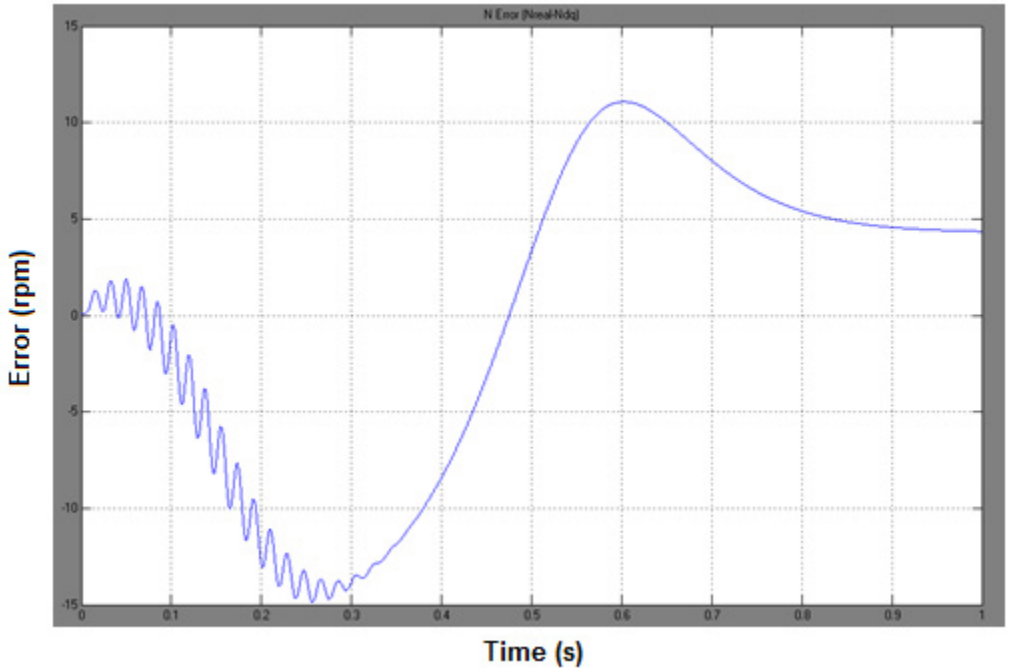

Figure 4.7.4: error in speed (rpm), (real model – dq model)

4.7.4. Torque Signals and Error

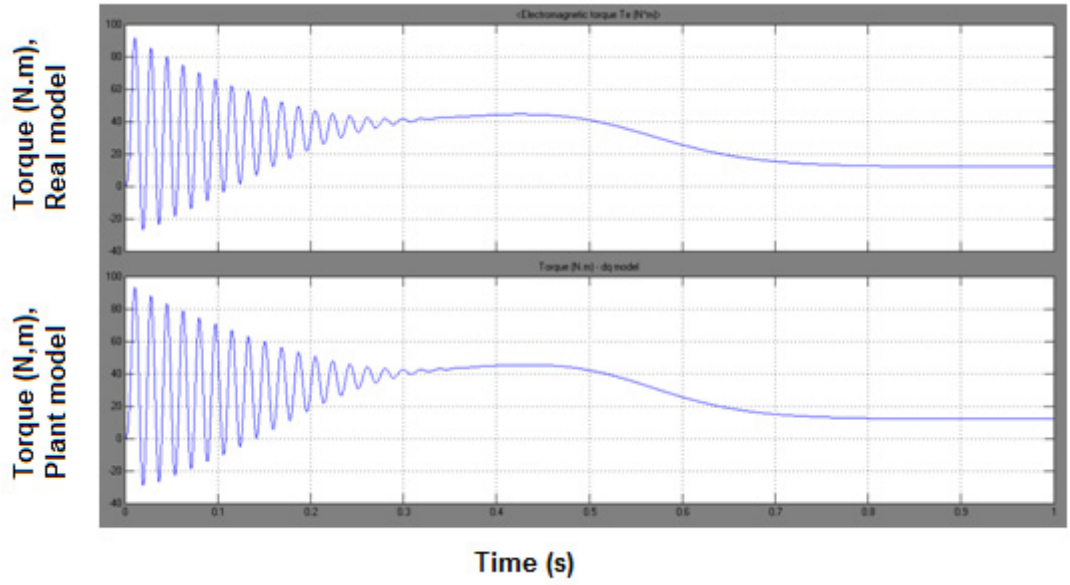

Figure 4.7.5: torque signals (N.m), (up) real model, (down) d-q model

Comparing both torque signals (figure 4.7.5) it can be seen that the shape is similar, both have an oscillatory transient and then stabilize at 11.9 which is the nominal load. The torque signal also has an error due to the same reasons than the speed error (figure 4.7.6).

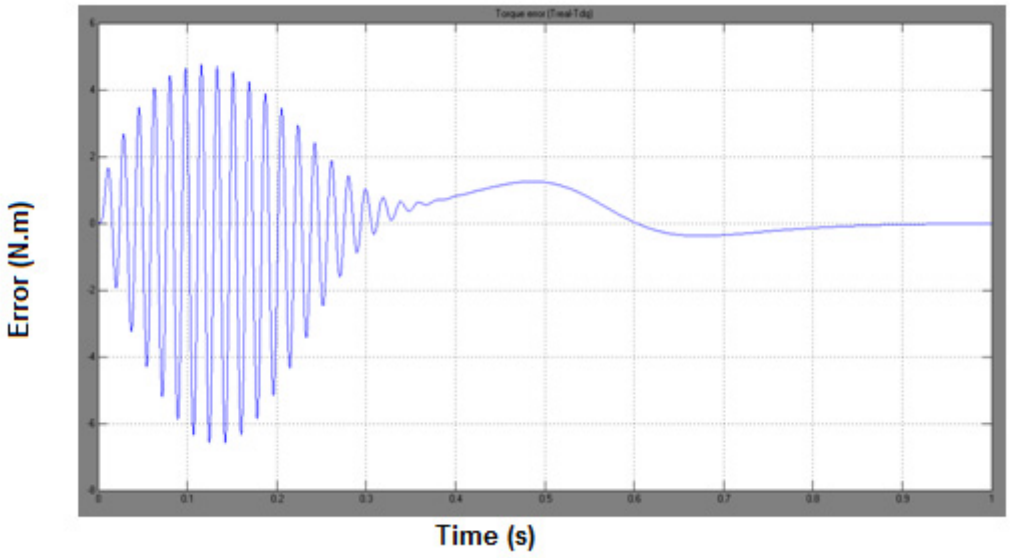

Figure 4.7.6: error in torque (N.m), (real model – dq model)

4.7.5. Changing Between Different Set Points

For this section a change of set points (change in rotor speed reference) occurs once the motor is already started, in other words, it occurs on the run. The point is to see if the d-q model can follow such change effectively. A step signal was set to reduce the set point from 1700 rpm to 1400 rpm after 1 second of started the simulation; simulation time=2s.

It can be seen that the d-q model can follow such changes (figure 4.7.7) and that the error (figure 4.7.8) remains in its original range. The d-q model is reliable as a

mathematical model of an induction motor. Its behavior follows the behavior of a real induction motor so it can be used as a reference for a control system.

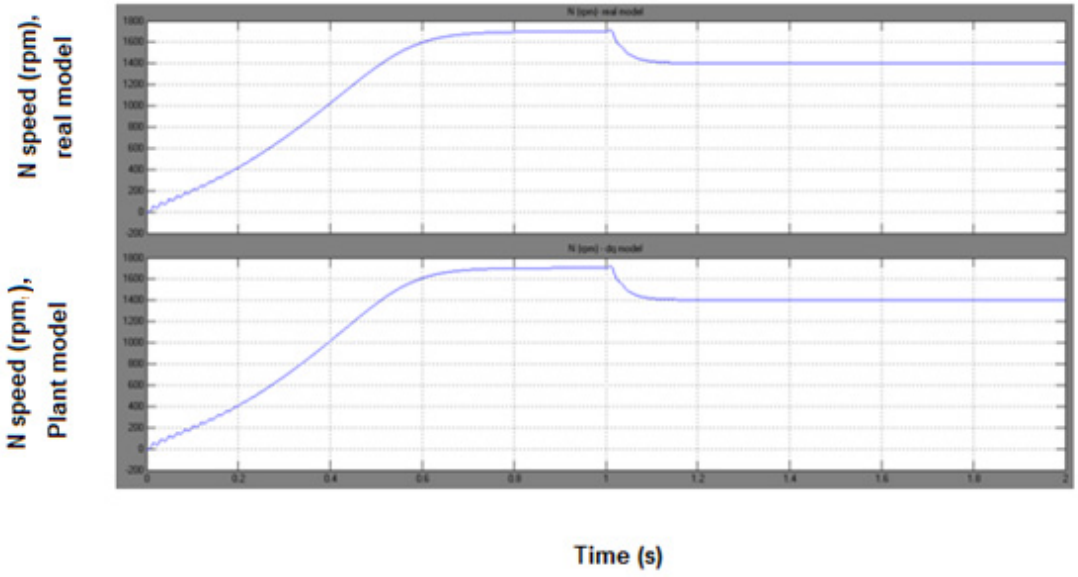

Figure 4.7.7: speeds changing between different set points (rpm), (up) real model, (down) d-q model

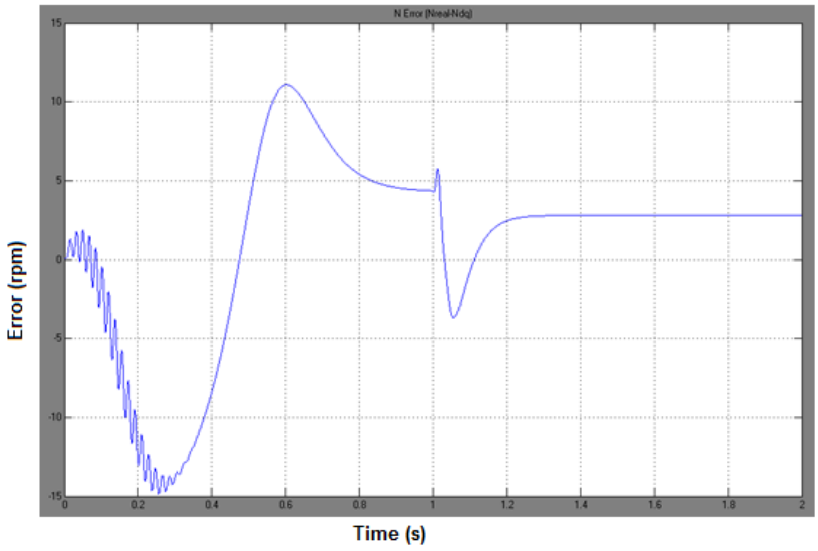

Figure 4.7.8: speed error for change between different set points (rpm), (real model – dq model)

4.8. PI CONTROL FOR INDUCTION MOTOR MODEL IN A D-Q PLANE

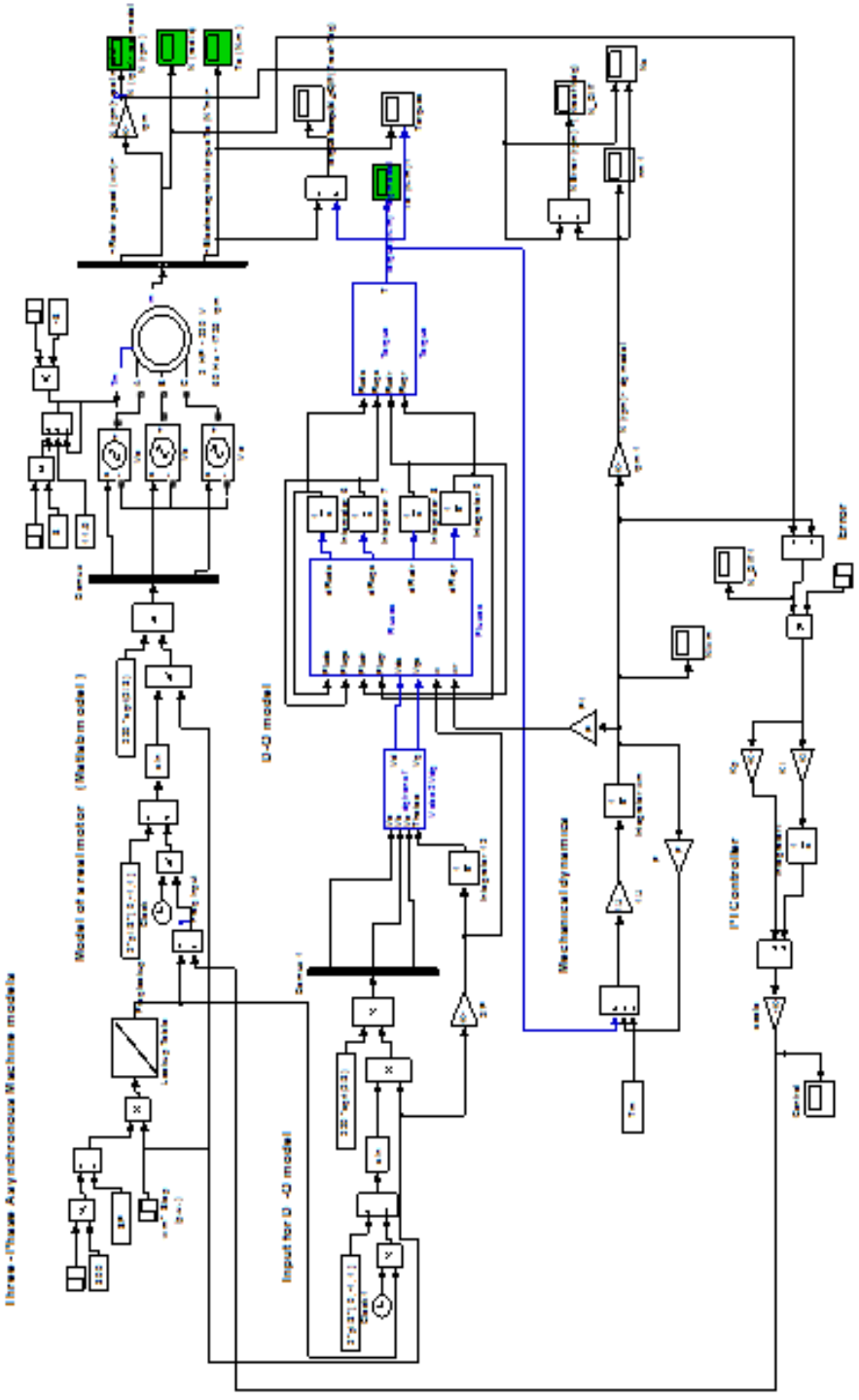

Figure 4.8.1: Simulink schematic, d-q plane model, PI controller

In figure 4.8.1 a PI controller is feed with the error between the mechanical speed of the d-q model (predicted behavior) and the one of the Matlab asynchronous machine (“real” behavior); this error is calculated from the speeds in rads per second and not from the rpm speed in order to reduce the magnitude of the signal (although both signals are identical, except for their magnitude). Since the PI is not supposed to act during the start up of the motor (initial transient) it is activated at t=1s, the same time the disturbances act.

4.8.1. Simulation of a Pi Control for an Induction Motor Model Developed In a D-Q Plane with Ziegler and Nichols Tuning Method

The gains for this simulation were Kp=0.01 and Ki=7. When manually tuning (fine tuning) it became obvious that a higher proportional gain makes the system unstable; the integral gain cannot be larger than this value either since the small error accumulated during changes of speed can make the system unstable. The system has a maximum error of 18 rpm which is 1.06% of the nominal speed (figure s4.8.2).

It can be seen in figure 4.8.3 that there is a small oscillation but that the controller manages to drive the frequency of the motor to the proper value in order to get the desired mechanical speed. The increase of the mechanical load (disturbances) was of 9.5 N.m, 80% of the nominal load, enough to see the effect of the controller. The maximum error is 18 rpm (1.06 % of nominal speed) and 4.7 rpm after 1 second (0.28 % of nominal speed).

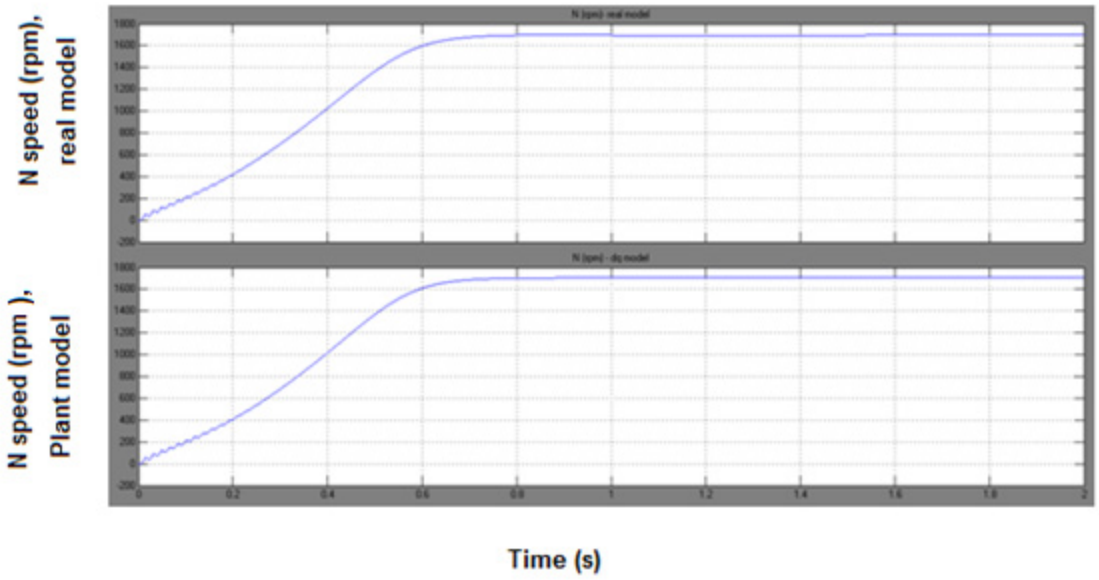

Figure 4.8.2.A: (up) real model speed (rpm) facing disturbances at t=1s; (down) d-q model speed, reference (rpm)

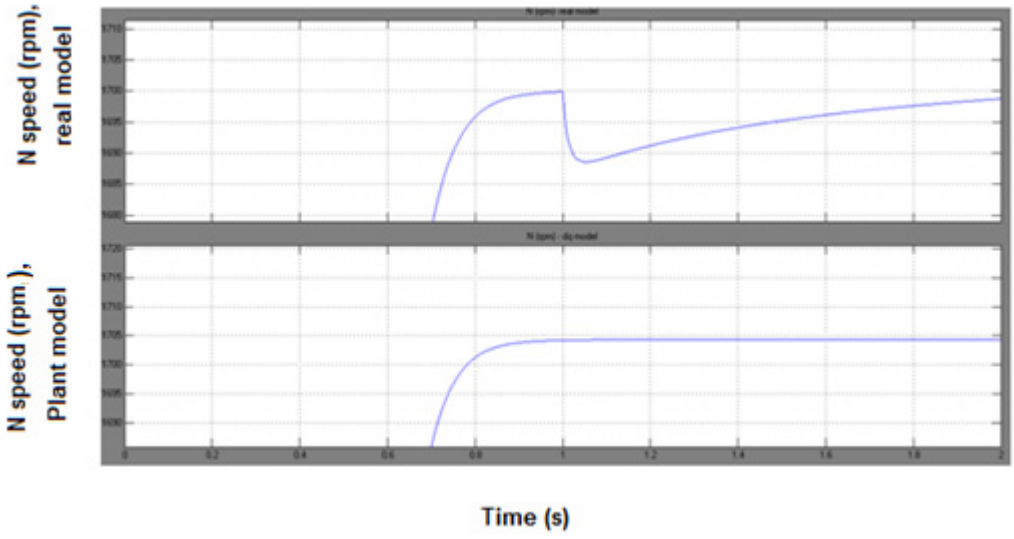

Figure 4.8.2.B: (up) real model speed (rpm) facing disturbances at t=1s; (down) d-q model speed, reference (rpm) (zoomed)

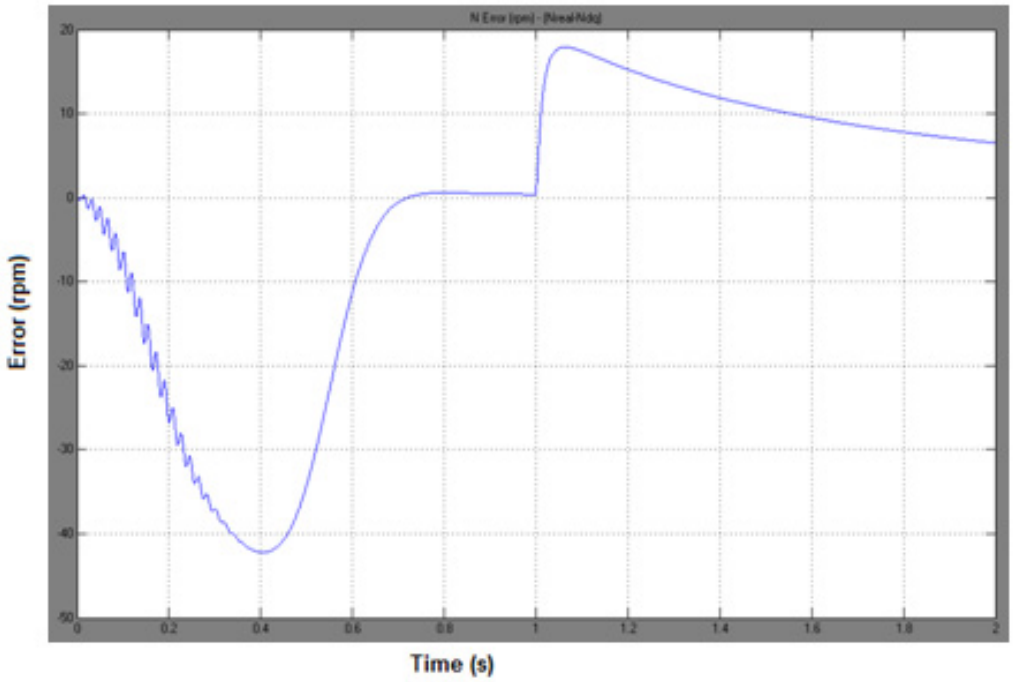

Figure 4.8.3: error (rpm) with disturbances at t=1, wmreal – wmdqmodel

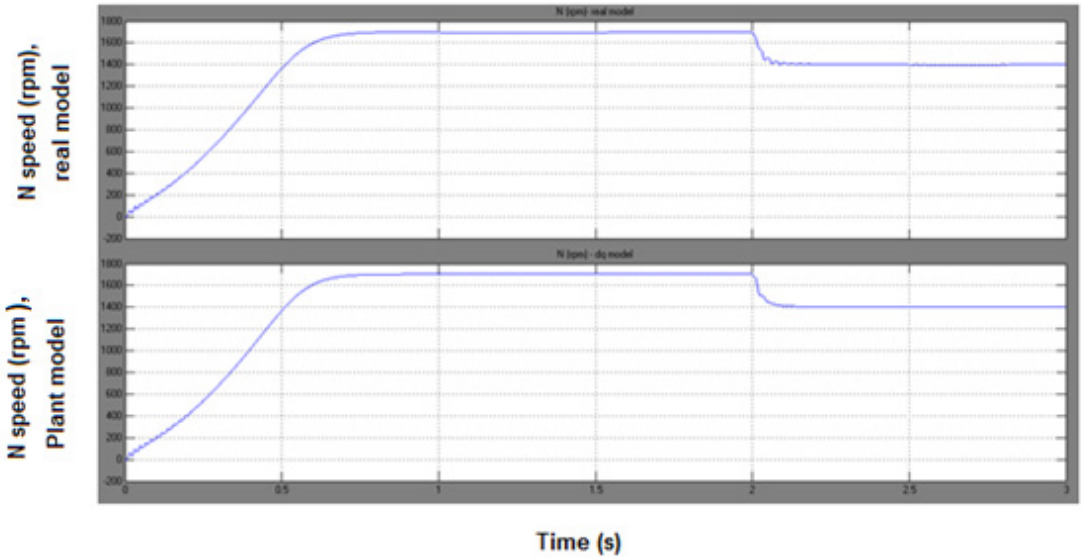

Figure 4.8.4.A: speed facing disturbances and change of set point (rpm), (up) real model, (down) d-q model

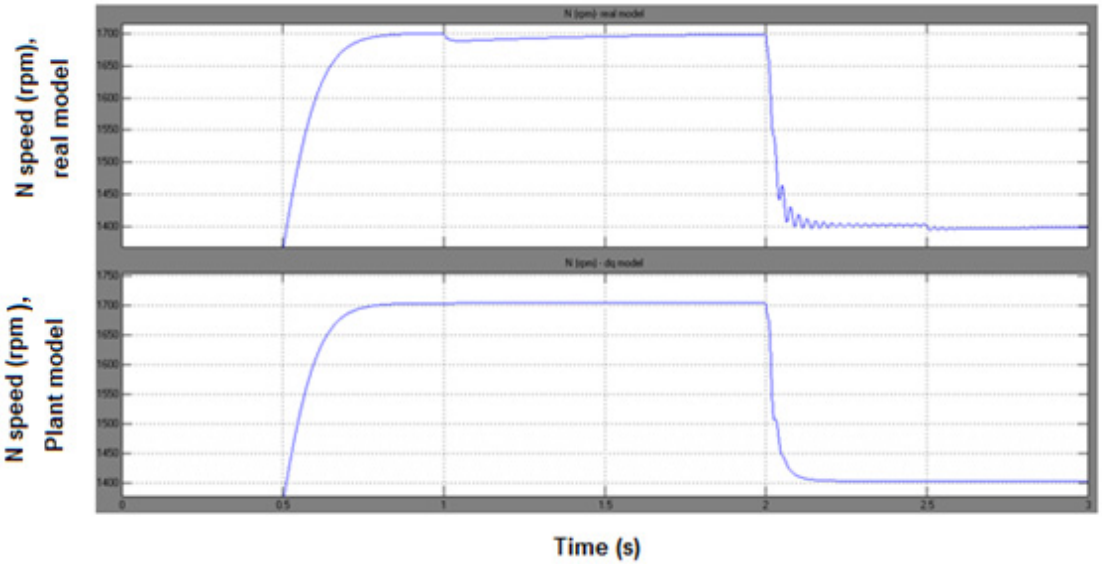

Figure 4.8.4.B: speed facing disturbances and change of set point (rpm), (up) real model, (down) d-q model (zoomed)

In figure 4.8.4 it can be seen a mixture of disturbances with change of speed reference. Between t=0 s and t=1s the motors simply reach the set point of 1700 rpm; at t=1s disturbances and PI controller are activated, at t=2s a change of speed is requested (1400 rpm), at t=2.5s disturbances are turned off.

Feeding back the system through the frequency of the input voltage signal makes it a little unstable, thus the Integral gain had to be big and the proportional gain had to be small. Whit this values of gains a small oscillation is seen but it doesn't drive the system to instability.

4.8.2. Pi Control for Induction Motor Model Developed In D-Q Plane with Hinf Pi

As seen in section 3.5.4, the first set of gains (equations 3.5.41 - 3.5.43) for this PI controller are Kp= 0.0106, Ki=0.1959, and Kd=0.

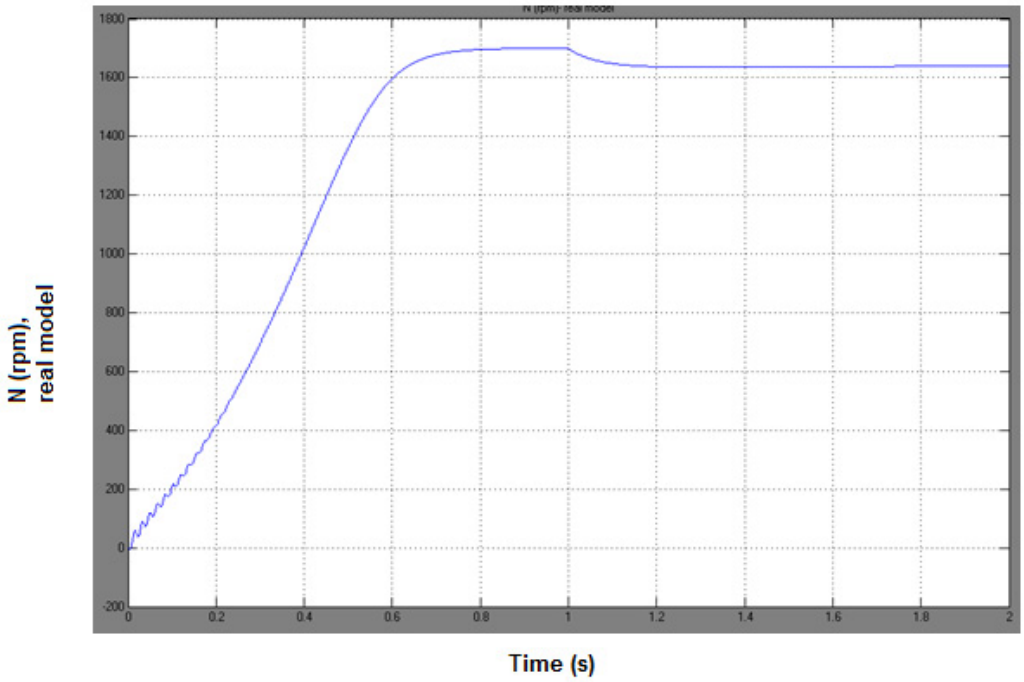

Figure 4.8.5.A: speed with PI controller gains of section 3.5.4

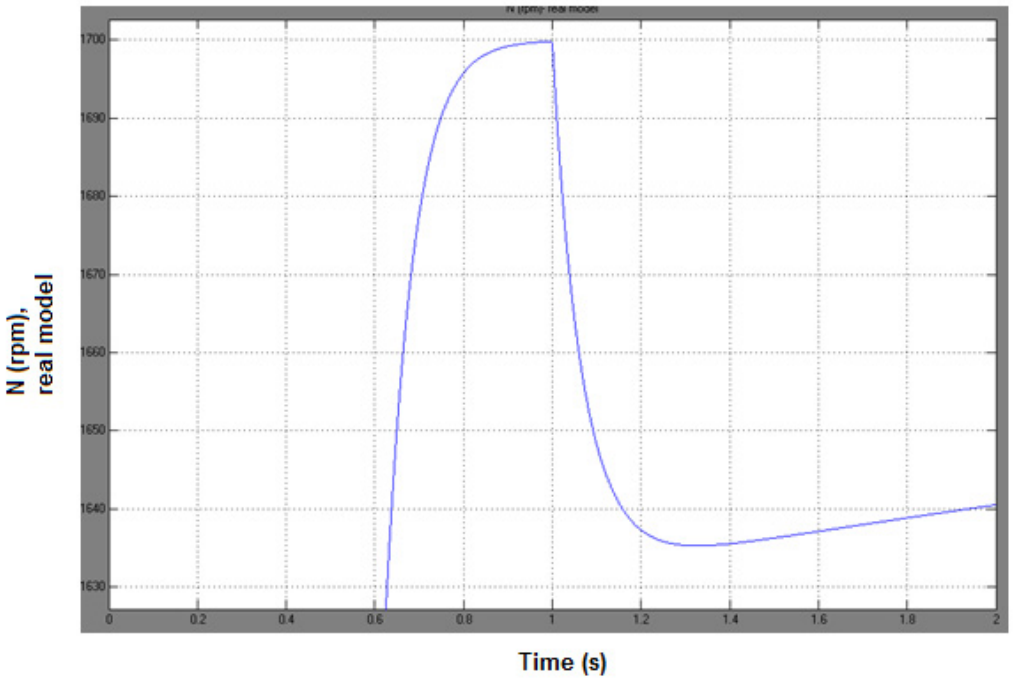

Figure 4.8.5.B: speed with PI controller gains of section 3.5.4 (zoomed)

The system is stable but its maximum error is larger than expected (about 60 rpm, around 3.5%) and after 1 second it doesn't improve much (3.4%). The proportional gain cannot be increased because it can drive the motor to instability; however the integral gain can be increased (manual tuning has it of Ki=7).

Simulation with an increased integral gain (equations 3.5.44 - 3.5.46) can be seen in figures 4.8.5 to 4.8.7.

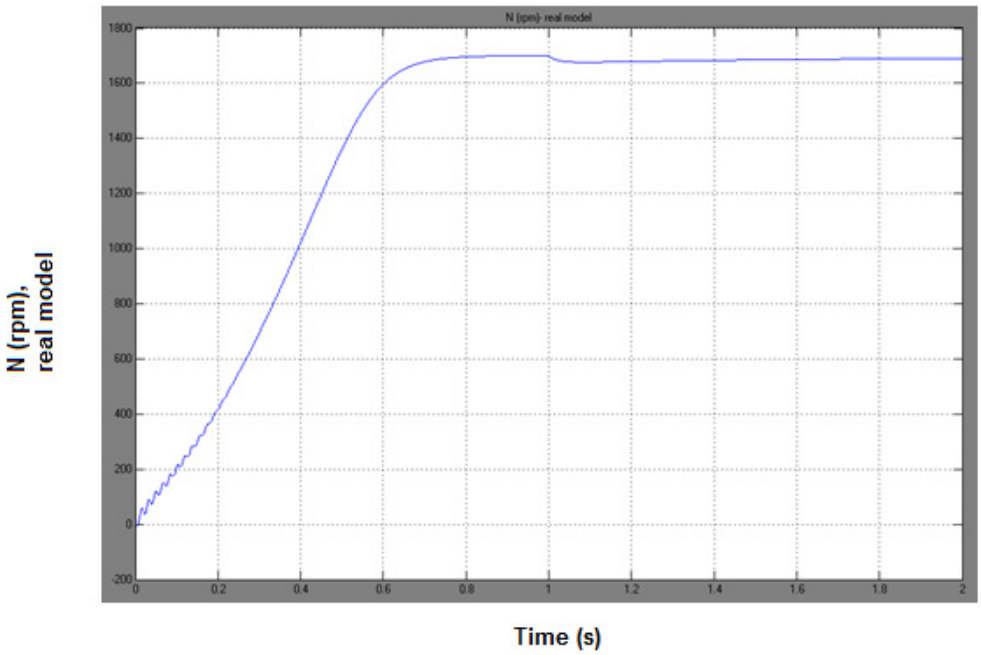

Figure 4.8.6.A: speed with PI controller gains, section 3.5.4, fine tuned

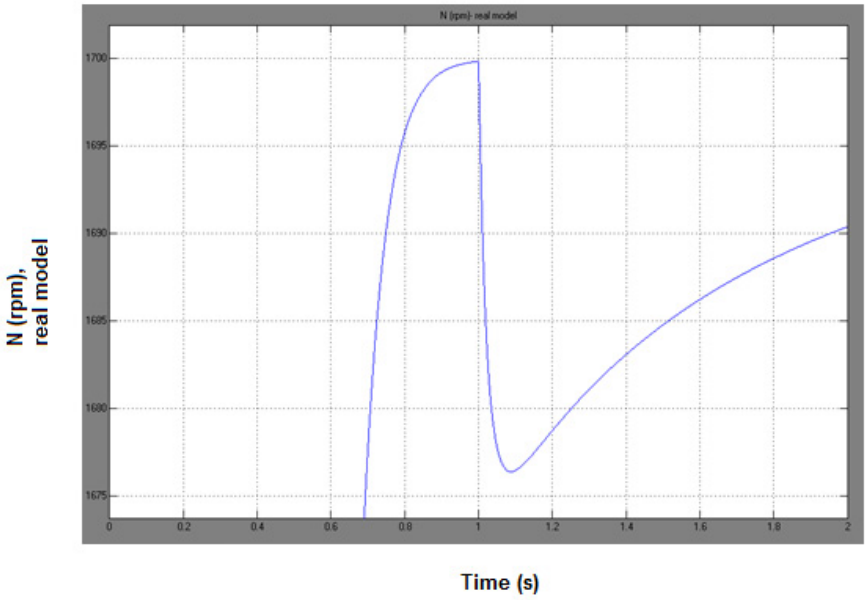

Figure 4.8.6.B: speed with PI controller gains, section 3.5.4, fine tuned (zoomed)

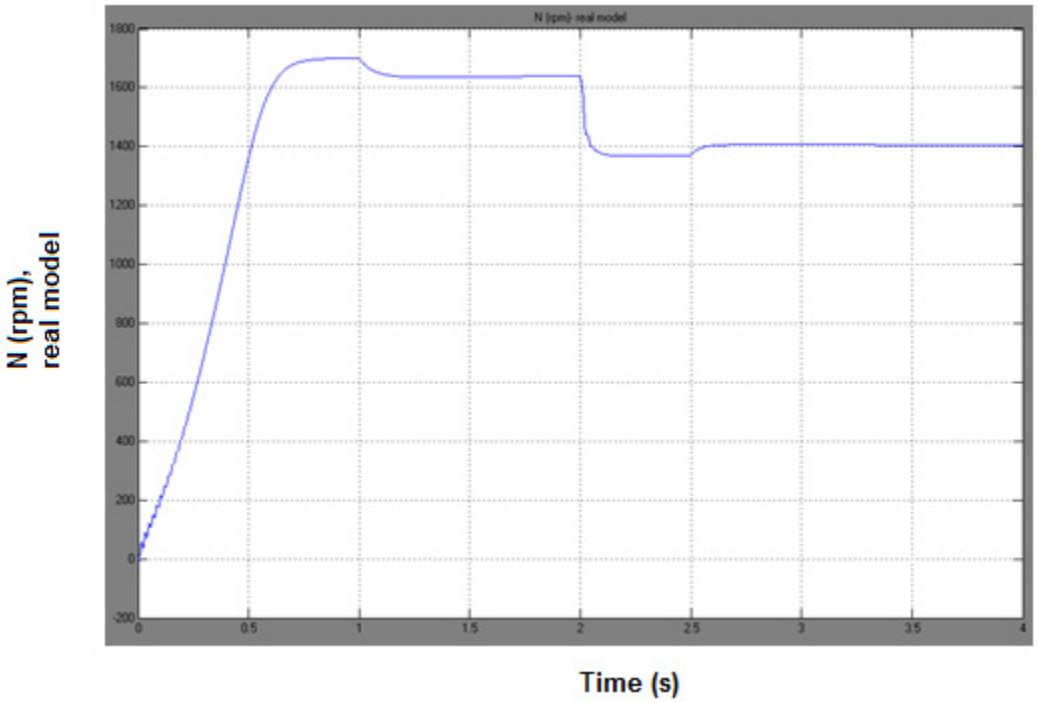

Figure 4.8.7.A: speed with perturbations and change of set point

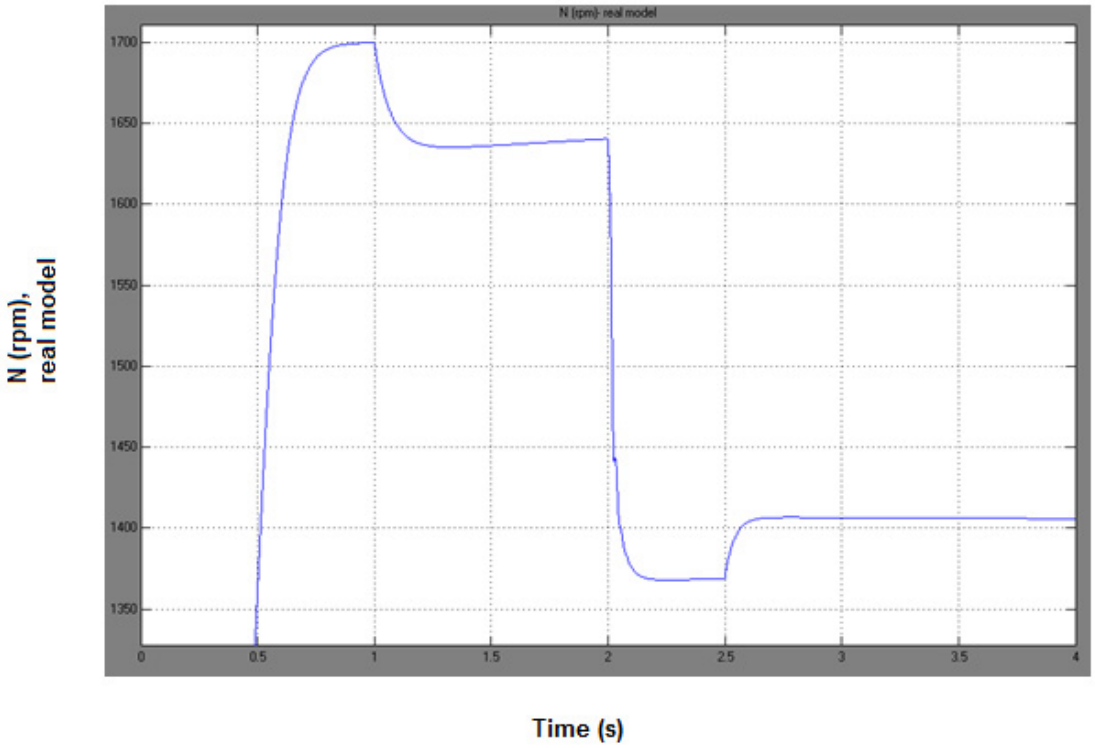

Figure 4.8.7.B: speed with perturbations and change of set point (zoomed)

The figure 4.8.6.b shows a maximum error of -24 rpm which is 1.4% of the nominal speed, and a final error of -10 rpm (0.6% of nominal speed) after 1 second ; both within the limits of the desired error.

The system remains stable for changes of set point, within the same limits as figures 4.8.7 shows.

Parameter	Zieg. & Nich.	Hinf	Hinf tuned
Kp	0.01	0.0106	0.0106
Ki	7	0.1959	4.5
Kd	0	0	0
Max speed error	1 %	3.5 %	1.4 %
Speed error after 1 sec.	0.29%	3.4 %	0.6 %

Table 4.8.1: different gains used in PI controller

From the table 4.8.1 it can be seen that the 3rd set of gains (that shown good results) is closer to the one obtain by Ziegler and Nichols. The system showed similar responses with the 1st and the 3rd sets. They both have small error; however the 3rd set seems a little more stable than the first one (less oscillation in the speed graphs). The best gains for this PI controller are, therefore, Kp= 0.01, Ki=4.5 and Kd=0

4.9. INDUCTION MOTOR MODEL USING VECTOR CONTROL IN A D-Q PLANE

As seen in figure 3.5.1 aligning the rotor flux with the d-axis allows calculating the slip frequency that along with the mechanical frequency, leads to the proper synchronous speed and proper input angle for the abc to dq transformation. That itself constitutes the principles of vector field control.

The figure 4.9.1 represents the system used for that simulation. It's constituted of 3 parts: the real motor, simulated by Matlab block; the d-q model, represented by 3 embedded blocks; and the mechanical system, dealing with the load and torque. As in section 4.7 the real motor is feed by a 3-phase sinusoidal voltage, as well as the d-q model; and the mechanical system has the same values that in previous sections. The main difference between this mode and the one of section 4.7 is that the "Fluxes" block contains the reduced equations seen in section 3.6, i.e. equations 3.6.4, 3.6.5, 3.6.7 and 3.6.9; and the "Torque" block contains the reduced equation 3.6.3.

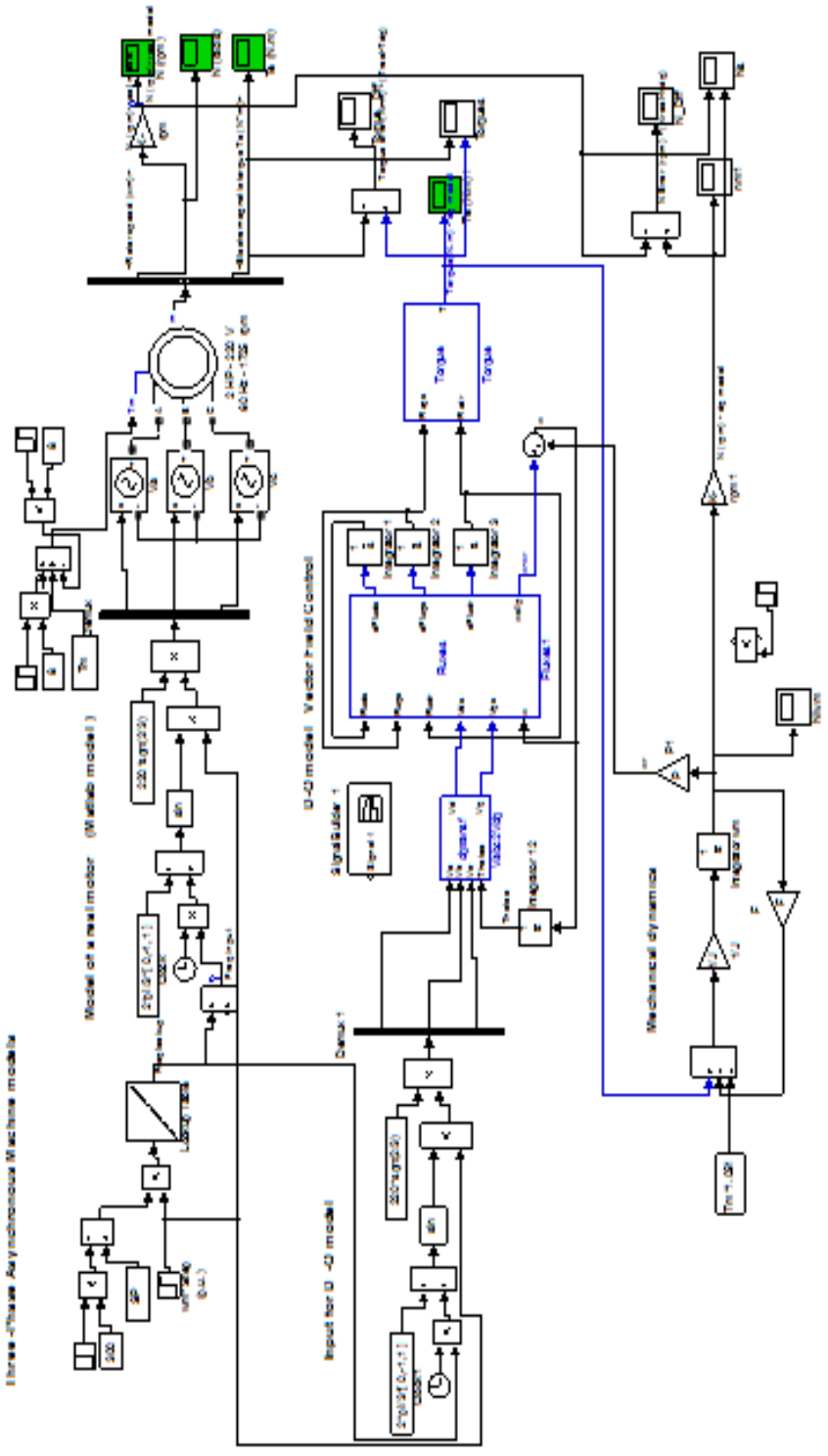

Figure 4.9.1: Simulink schematic, vector field control

4.9.1. Reaching a Defined Set Point of Speed

This test requires that the model reaches a defined set point (rpm); it is compared with a real motor (Matlab model) in order to see if there is any error. The set point for this test is a mechanical speed of 1700 rpm (figure 4.9.2)

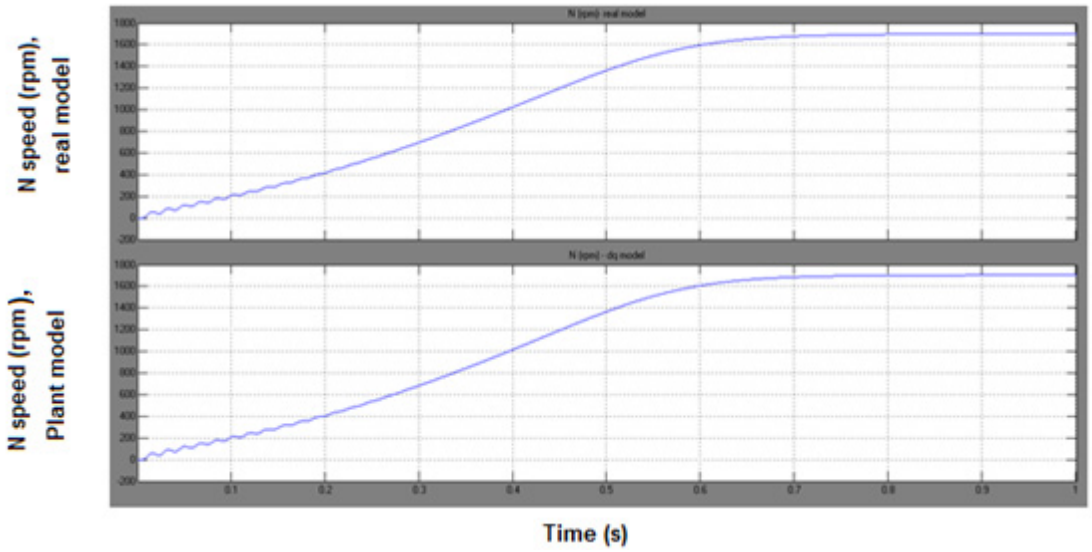

Figure 4.9.2: speeds reaching a set point (rpm), (up) real model, (down) d-q model

As in the model of section 4.7 the d-q model obtains the desired speed in the same time, with a small oscillation at the beginning and with a final error of +4 rpm.

4.9.2. Transient of Speed

The transient is similar to the ones already seen; it has also an oscillatory phase and then it stabilizes at the set point. Both models follow the same path (figures 4.9.3)

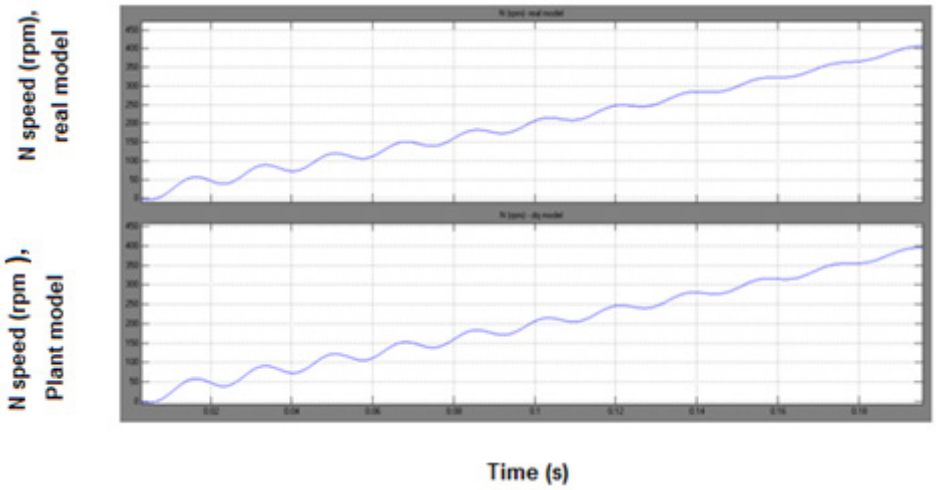

Figure 4.9.3.A: speed transient (rpm) from 0 to 0.2 s, (up) real model, (down) d-q model

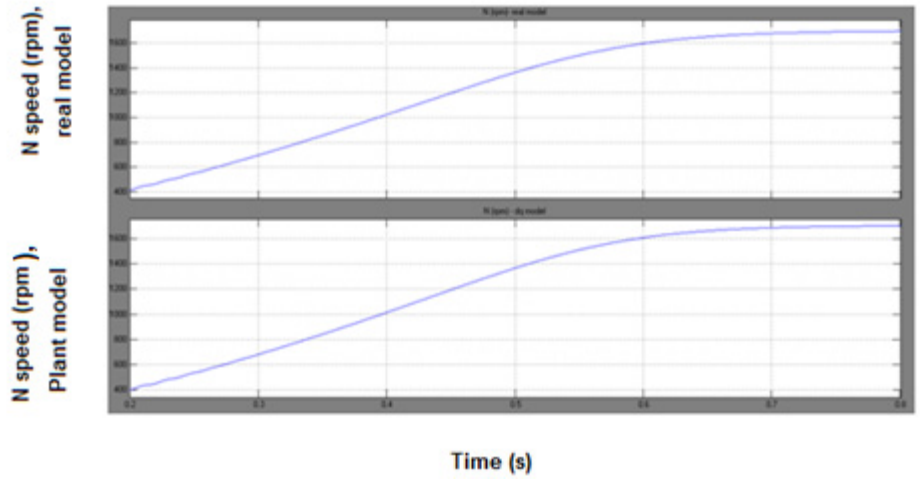

Figure 4.9.3.B: speed transient (rpm) from 0.2 to 0.8 s, (up) real model, (down) d-q model

4.9.3. Error In Speed

As seen in figure 4.9.4 the error is very similar to the one of section 4.7, it reaches a peak around 13 rpm (0.7 % of nominal speed) and then stabilizes at 5 rpm (0.29% of nominal speed).

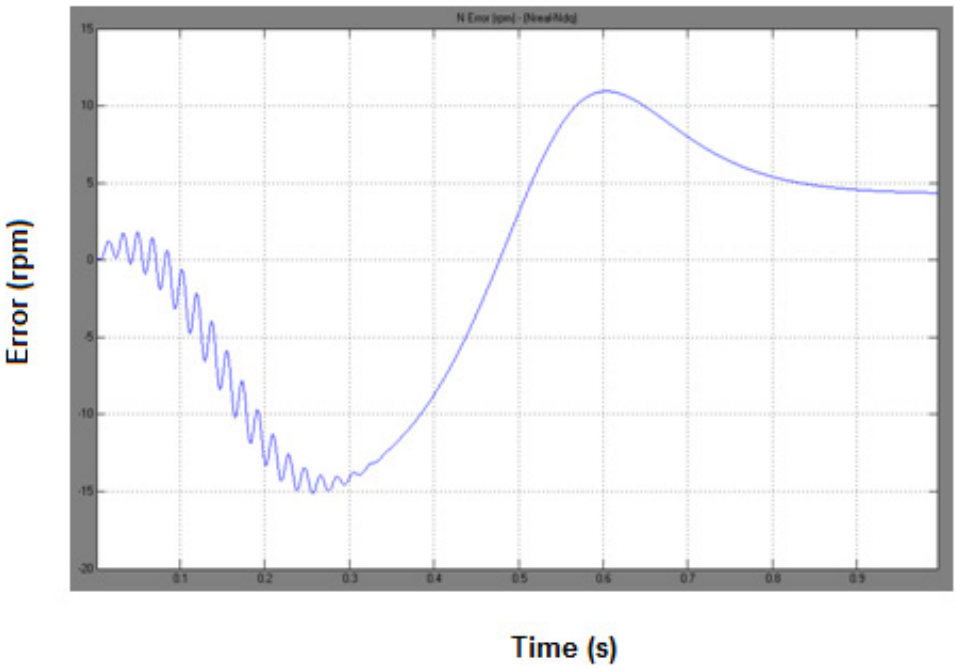

Figure 4.9.4: error in speed (rpm), (real model – dq model)

4.9.4. Torque Signals and Error

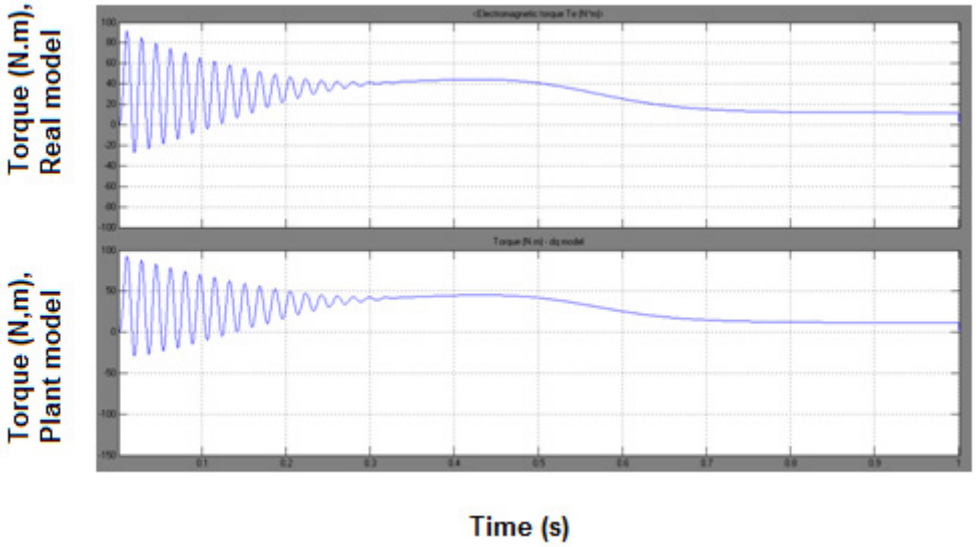

Figure 4.9.5: torque signals (N.m), (up) real model, (down) d-q model

Both torque signal and error (figure 4.9.5 and 4.9.6) resemble their equivalents of section 4.7. Real and Plant model follow the same path and error has some oscillations during the transient phase but then stabilizes at zero.

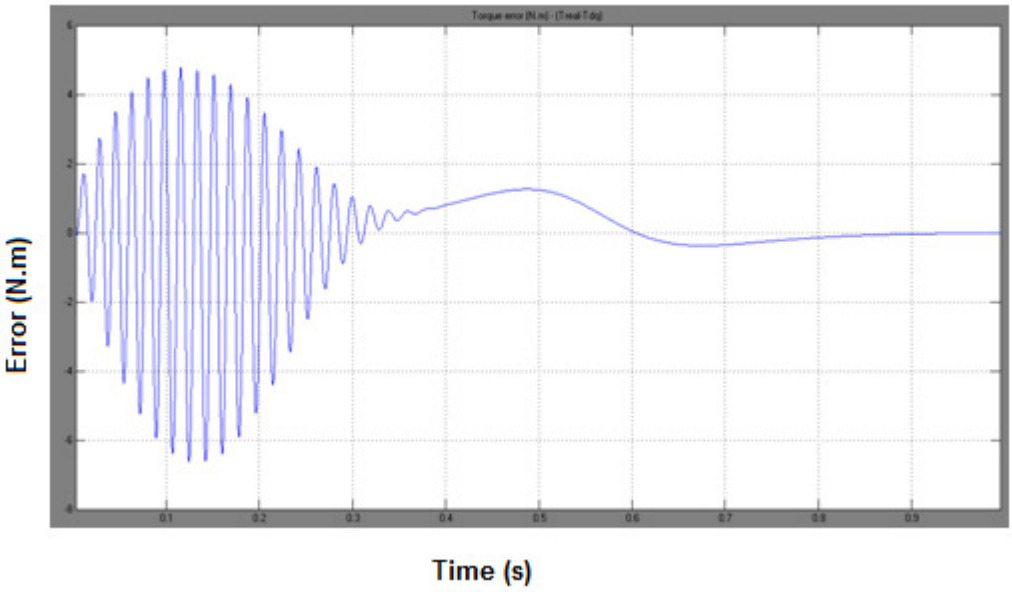

Figure 4.9.6: error in torque (N.m), (real model – dq model)

4.9.4. Changing Between Different Set Points

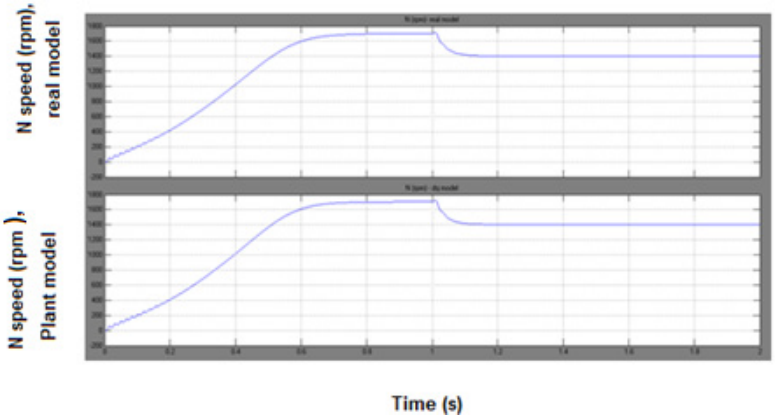

Figure 4.9.7: speeds changing between different set points (rpm), (up) real model, (down) d-q model

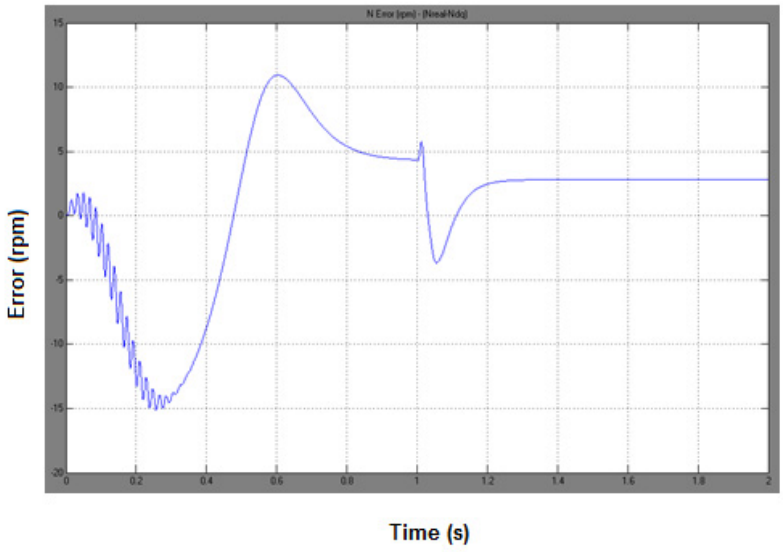

Figure 4.9.8: speed error for change between different set points (rpm), (real model – dq model)

All the graphics turn out to be exactly as in section 4.7; this is obviously due to the fact that both models use almost the same equations. Notice that for Vector Control one flux was eliminated to find the slip frequency; however by feeding back the angle Thetae to the transformation block, the result is the same as if it had the original equations; the advantage is that now the model allows finding ws directly. The system allows controlling 2 variables torque and speed, the cost is a more complex control system.

4.10. BALDOR MOTORS PLATFORM

The Baldor motors platform (test-bench) is a system of 2 coupled motors (AC inductor motor, 60 Hz, 3-phases) attached by a torque cell (see figure 4.10.1). One of the motors (once started) can act as a generator for the other one, which will be the test motor. The torque cell allows the user to measure torque and

speed between the two motors; the system also has place for instrumentations and control systems.

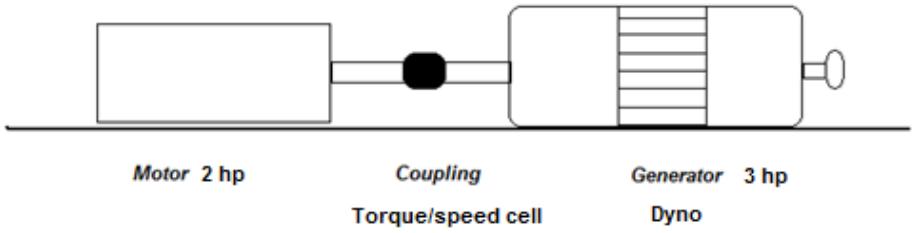

Figure 4.10.1: conceptual layout of Baldor motors system

The generator controls the test motor by a frequency drive incorporated in the system; a speed control potentiometer is also included that can act as the nominal speed. Instruments, torque cell and the control system provide the data that will be use to feed this project control mechanism. As seen in figure 4.10.1, the user can add a second control system block that fits his purpose.

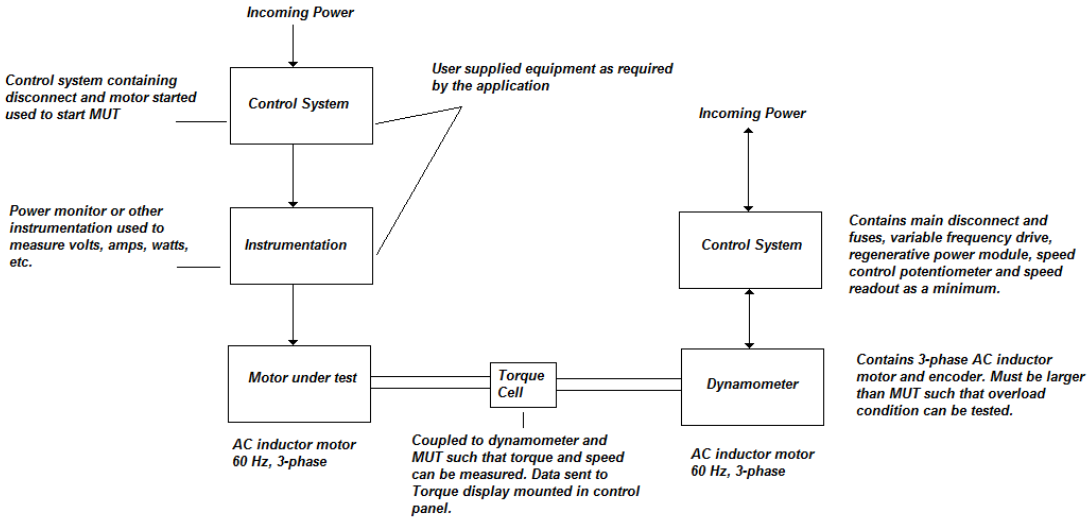

Figure 4.10.2: Baldor motors platform blocks diagram

At 460 V, 60 Hz, 2.0 HP, the typical performance characteristics of the platform can be seen in table 4.10.1 (Horlick, 2009).

Performance data at 460 V, 60 Hz, 2.0 HP (typical performance)							
General Characteristics							
Full load torque	5.94 LB-FT			**Start configuration**	DOL		
No-load current	2.16 Amp			**Break-down torque**	23.5 LB-FT		
Line-line res. @ 25°C	8.41 Ω A ph/ 0 Ω B ph			**Pull-up torque**	15.3 LB-FT		
Temp. rise @ rated load	48°C			**Locked-rotor torque**	18.0 LB-FT		
Temp. rise @ S.F. load	58°C			**Starting current**	23.4 Amp		
Load characteristics							
% of rated load	**25**	**50**	**75**	**100**	**125**	**150**	**S.F.**
Power factor	30.0	48.0	61.0	71.0	77.0	81.0	75.0
Efficiency	68.6	79.6	83.3	84.3	84.0	83.3	84.1
Speed	1792.0	1781.0	1771.0	1759.0	1744.0	1731.0	1750.0
Line Amperes	2.22	2.42	2.72	3.13	3.6	4.14	3.41

Table 4.10.1: Baldor motors characteristics

4.10.1. Modeling Baldor Motor

The Baldor motor is going to be modeled following the dynamic model seen in section 3.1.3. Having this model, coherent with Matlab model, the simulations already done in previous sections can be adjusted to the Baldor parameters. In order to do that a series of test must be performed (see next section). Although a simulated version of the Baldor motor is useful, the control system must be implemented in the real motor. The simulations will help as a guide for the design and configuration of such system.

4.10.2. Tests to Model an AC Motor, Determining Circuit Model Parameters

In order to determine the parameters of an induction motor model (see section 3.1.3) a series of tests must be done to determine Rs, Rr, Xs, Xr and Xm from the equivalent circuit shown below.

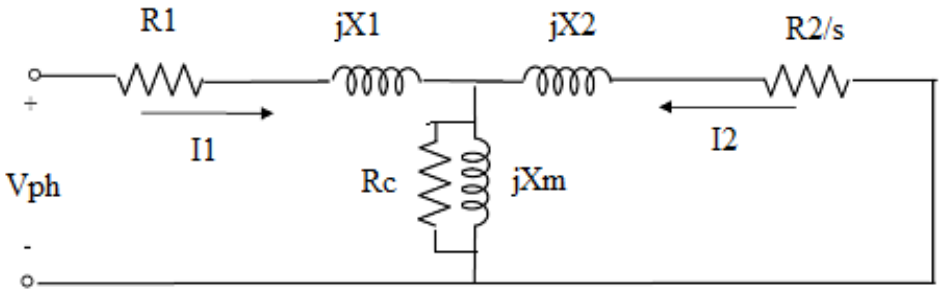

Figure 4.10.3: dynamic model of an induction motor

The tests must be performed under precisely controlled conditions, since the resistances vary with temperature and the rotor resistance also varies with rotor frequency (Chapman, 2005).

The No-Load Test.

It measures the rotational losses of the motor and provides information about its magnetization current. The test circuit for this test is shown in figure 4.7.4. Two wattmeter, a voltmeter and 3 ammeters are connected to the induction motor, which is allowed to spin freely.

The only load in the motor is the friction and winding losses, so all the converted power is consumed by mechanical losses and the slip of the motor is very small. The initial equivalent circuit is shown below in figure 4.10.5. The resistance

equivalent to its power converted must to be added, since it was initially simplified.

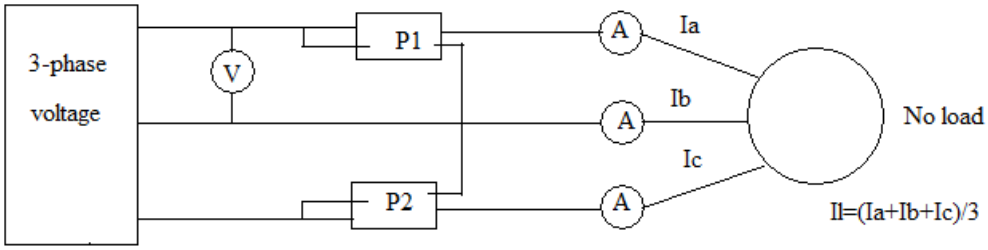

Figure 4.10.4: test circuit for no-load test

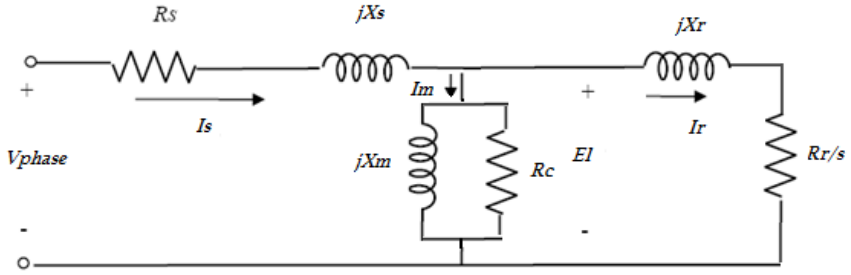

Figure 4.10.5: initial equivalent circuit

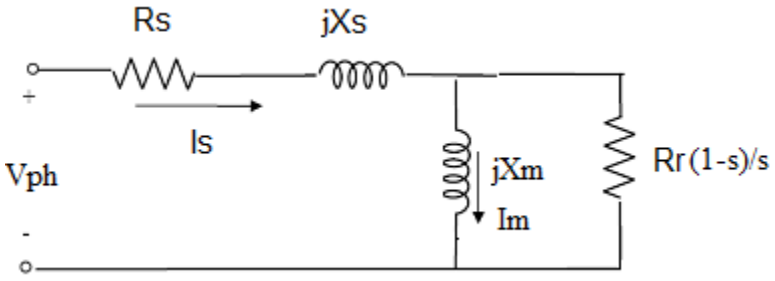

Figure 4.10.6: reduced equivalent circuit for no-load test

Since it has a very small slip, the resistance corresponding to its power converted $R_r\left(\frac{1-s}{s}\right)$ is much larger that its resistance conrresponding to the rotor

copper losses Rr and much larger than the rotor reactance Xr, so both are neglected; this also means that Ir is very small. Then the circuit can be reduced to the one in figure 4.10.6.

In this motor at no-load conditions the input power measured must be equal to the losses in the motor. The stator copper losses are given by:

$$P_{SCL} = 3I_l^2 R_s \qquad (4.10.1)$$

The input power must be:

$$P_{in} = P_{SCL} + P_{core} + P_{F\&W} + P_{misc} \qquad (4.10.2)$$

where Prot is the rotational losses of the motor. Also

$$P_{in} = P_{core} + P_{F\&W} + P_{misc} \qquad (4.10.4)$$

where Pf&W is the power consumed by the equivalent impedance $R_r\left(\frac{1-s}{s}\right)$. Given the input power of the motor the rotational losses can be determined. The parallel between $R_r\left(\frac{1-s}{s}\right)$and Rc is larger than Xm impedance so must of the current will flow through the inductive component jXm. The equivalent input impedance is approximately:

$$\left|Z_{eq}\right| = \frac{V_\phi}{I_l} \approx X_s + X_M \qquad (4.10.5)$$

Finding Xs by other means Xm can be determined (Chapman, 2005).

The Dc Test for Stator Resistance

This test is used to determine Rs. A dc voltage is applied to the stator windings; because the current is DC, there is no induced voltage in the rotor circuit and no resulting rotor current flow. Also, the reactance of the motor is zero at DC current. Therefore the only element limiting the current is the resistance Rs. The test circuit can be seen in figure 4.10.7.

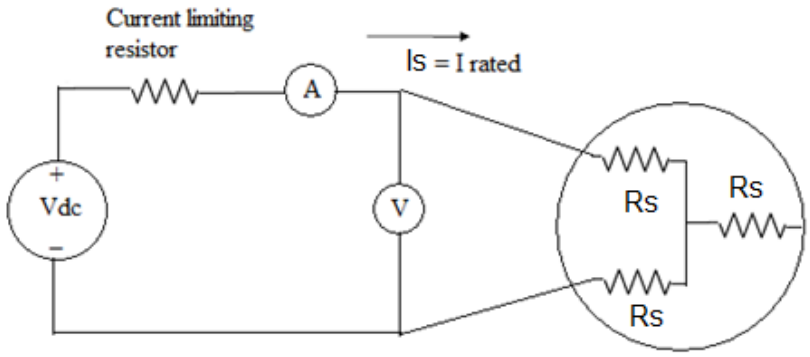

Figure 4.10.7: test circuit for DC test for stator resistance

The circuit shows a DC power supply connected to two of the three terminals of a Y-connected induction motor. The current in the stator windings is adjusted to the rated value, and the voltage between terminals is measured. The current in the stator windings is adjusted to the rated value in an attempt to heat the windings to the same temperature they would have during normal operation.

The total resistance is then:

$$2R_s = \frac{V_{DC}}{I_{DC}} \qquad (4.10.6)$$

$$R_s = \frac{V_{DC}}{2I_{DC}} \qquad (4.10.7)$$

With Rs, the stator copper losses P_{SCL} can be determined (see equation 4.10.1 PSCL) as well as the rotational losses Prot (see equation 4.10.3, P_{rot}) (Chapman, 2005).

The Locked-Rotor Test

This test is intended to determine the total rotor reactance referred to the stator, and R2. In this test, the rotor is locked so it cannot move, a voltage is applied to the motor and the resulting voltage, current and power are measured.

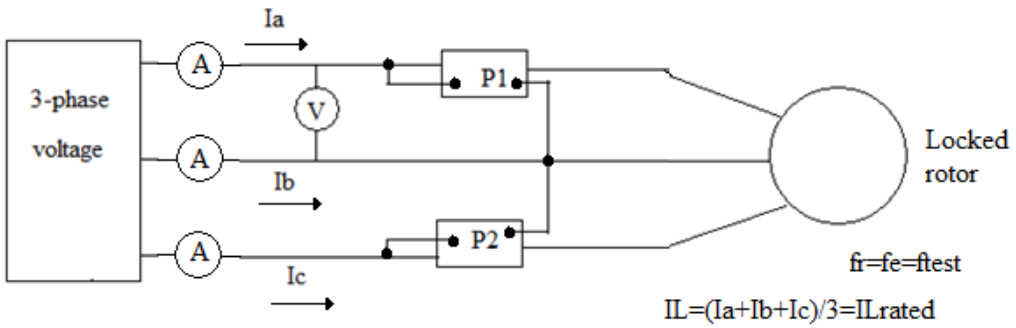

Figure 4.10.8: test circuit for locked-rotor test

The test circuit is shown in figure 4.10.8. An AC voltage is applied to the stator and the current flow is adjusted to be approximately full-load value. When the current is full-load value, the voltage, current and power flowing into the motor are measured.

Figure 4.10.9 shows the equivalent circuit. Since the rotor is not moving the slip s=1, and the resistance Rr/s is just equal to Rr, a small value. Since Rr and Xr are so small most of the current will flow through them, just a small portion will flow through Xm, so it can be neglected. This test, however, is not entirely accurate for design B and C motors and some adjustments must be done (see reference (Chapman, 2005), page 456).

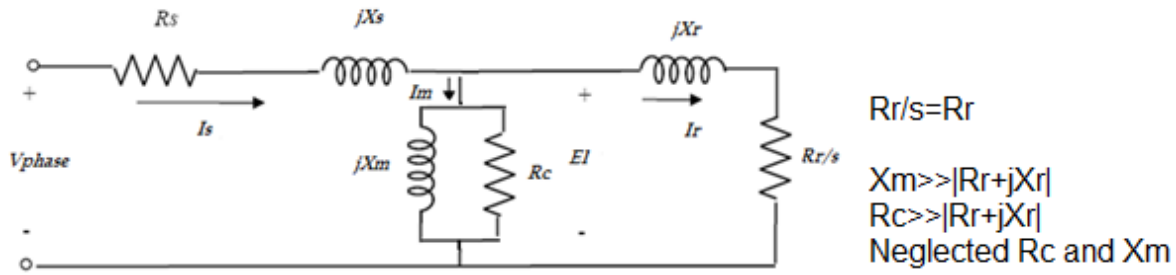

Figure 4.10.9: motor equivalent circuit for locked-rotor test

The input power is given by

$$P = \sqrt{3}V_T I_L cos\boldsymbol{\theta} \qquad (4.10.8).$$

So the locked-rotor power factor can be found as

$$PF = cos\boldsymbol{\theta} = \frac{\boldsymbol{P_{in}}}{\sqrt{3}V_T I_L} \qquad (4.10.9)$$

The magnitude of the total impedance in the motor circuit is

$$|Z_{LR}| = \frac{V_\emptyset}{I_1} = \frac{V_T}{\sqrt{3}I_L} \qquad (4.10.10)$$

and the angle of the total impedance is $\boldsymbol{\theta}$. Therefore

$$Z_{LR} = R_{LR} + jX'_{LR}$$

$$Z_{LR} = |Z_{LR}|cos\theta + j|Z_{LR}|sin\theta \quad (4.10.11)$$

The locked-rotor resistance is equal to

$$R_{LR} = R_s + R_r \quad (4.10.12)$$

while the locked-rotor reactance X'_{LR} is equal to

$$X'_{LR} = X'_s + X'_r \quad (4.10.13)$$

where X'_s and X'_r are the stator and rotor reactances at the test frequency respectively.

The rotor resistance Rr can be found as

$$R_r = R_{LR} - R_s \quad (4.10.14)$$

where Rs was determined in the DC test. Total rotor reactance referred to the stator can also be found. Since the reactance is directly proportional to the frequency, the total equivalent reactance at the normal operating frequency can is given by

$$X_{LR} = \frac{f_{rated}}{f_{test}} = X_s + X_r \quad (4.10.15)$$

However, there is no simple way to separate the contributions of each reactance. An empirical table (table 4.10.2) can be consult to get their proportions (Chapman, 2005).

	Xs and Xr as functions of XLR	
Rotor design	**Xs**	**Xr**
Wound rotor	0.5 XLR	0.5 XLR
Design A	0.5 XLR	0.5 XLR
Design B	0.4 XLR	0.6 XLR
Design C	0.3 XLR	0.7 XLR
Design D	0.5 XLR	0.5 XLR

Table 4.10.2: rules of thumb for Xs and Xr

5. METHODOLOGY AND PROCEDURES

The project is limited to control an AC electrical motor; AC motor was selected over DC motors due to the advantages that the former has over the later. In spite of its simpler design and control, DC motors must be maintained frequently due to their brushes and rectifiers degradation; sometimes they are installed in places with difficult accessibility, for marine vehicles this is usually the case. AC motors have smaller size, lighter weight, lower rotary inertia and lower price; however, since they are basically non-linear and time-varying with a dynamic coupling system, the controller design is complex (Zhaoming, 2007).

The platform to test the performance of an electrical motor was recently acquired by FAU. This controlled motor system allows to change and control different parameters for the motor. The motor itself can be characterized as well. The system is called 3HP Dyno from Baldor motor products, Horlick (is referred as test bench in most of this document).

It is capable of measure several characteristic and performance values of the electrical motor, such as voltage, current, torque, speed, etc. It can be controlled by the Variable Frequency Drive and certain values (load, current, speed) can be changed as needed. Since the motor has its own frequency driver; this work can

be focused on the control system only. Using this system as test model the control system can be implemented in a controlled environment.

5.1. IDENTIFYING MOTOR PARAMETERS PROCEDURE.

5.1.1. Calculate Motor Values Procedure

Given the model of the motor explained in previous sections, this procedure is meant to identify all the parameters that the model is based on (Rs, Xs, Xm, Rr, Xr; Rc is neglected). A torque cell/Encoder is needed (Test-bench has one: frequency and direction, SM-universal encoder) and a serial connection with it. The test bench has a procedure to identify these parameters (autotune, see (Control Techniques, 2007)) but it is considered as an alternative option. This is the procedure to characterize the motors:

- Program the generator with 4 sets of 5 different loads. Section 5.1.
- Connect a PC to the torque cell using the serial connection, protocol R232. Check (Horlick, 2009) for reference on the torque cell and the encoder. Check (Control Techniques, 2007) for reference on communications or section 8.6.2. of this document.
- Start the motor and the generator. Se startup procedure 8.6.1.
- Run the motor for different loads, 4 sets of 5 different loads. See section 5.1.4.

- Gather the data and calculate the parameters using Matlab program. See section 5.1.3 and 5.1.5.

5.1.2. Serial Communications with Encoder

The Encoder gives values of torque and rpm for different loads at different instants, those values are transmitted via serial port. The encoder can also be connected directly to the Unidrive of the motor and this will handle the information. For, in this project, we are using 2 Unidrives (one for the generator and one for the motor) the Encoder can be assume connected to the Unidrive of the generator. However if it is directly connected to a PC, a program to transform the information into values of rpm and torque is needed (see (Horlick, 2009) for details on connection).

5.1.3. Theoretical Calculations for the Equivalent Circuit Parameters

As seen in section 3.4

$$T_e = \frac{3V_{TH}{}^2 R_r/_S}{\omega_s\left[\left(R_{TH}+{}^{R_r}/_S\right)^2+(X_{TH}+X_r)^2\right]} \qquad (3.4.13)$$

From the equation above, Vth, Rr, Rth, Xth and Xr are unknown. The synchronous speed is a constant for a fixed input frequency (assumed 60 Hz). The induce torque (Te) can be measure by the torque cell and set to the proper value. Slip can be calculated from the motor speed (given by the torque cell too). A system of 5 equations is needed; this means that at least 5 measures of torque and slip must be done in steady-state. Each pair torque-slip represents one

equation. Using equation 3.4.13 with 5 different values of torque and slip the system can be solved for Vth, Rr, Rth, Xth and Xr.

The values for Rs, Xs and Xm can be solved from equations 3.4.7, 3.4.9 and, 3.4.10

$$X_m = \frac{V_{TH} X_s}{V_\phi - V_{TH}} \quad (5.1.1)$$

$$R_s = \frac{R_{TH} (X_s + X_m)^2}{X_m^2} \quad (5.1.2)$$

$$X_s = X_{TH} \quad (5.1.3)$$

5.1.4. Theoretical Calculations for Inertia and Friction

$$\dot{\omega}_m = \frac{1}{J}(T_e - F\omega_m - T_l) \quad (3.1.79)$$

Consider that $\dot{\omega}_m$ is mechanical acceleration so the following must be true too:

$$\dot{\omega}_m = \frac{\omega_{m1} - \omega_{m0}}{t_1 - t_0} \quad (5.1.4)$$

Assuming the system is going from the nominal load to ever increasing values of load, it follows that $T_{e0} = T_{l0}$ for $t = t_0$; i.e. assume that it starts from steady-state.

$$\frac{\omega_{m1} - \omega_{m0}}{t_1 - t_0} = \frac{1}{J}(T_{l0} - F\omega_{m1} - T_{l1}) \quad (5.1.5)$$

The values of $t_0, t_1, \omega_{m0}, \omega_{m1}, T_{l0}\ and\ T_{l1}$ are measured; the unknowns are J and F. The system needs two sets of $t_0, t_1, \omega_{m0}, \omega_{m1}, T_{l0}\ and\ T_{l1}$ to be solved, this is, 2 changes between steady-states.

5.1.5. Sets of experiments

4 sets of 5 measures (5 different arbitrary loads) are designed to gather enough information and average the results. The load of the motor must change from the smallest load to the largest while the motor is running and the time between changes must be recorded. According to Horlick (Horlick, 2009) tests the maximum torque of the motor is 15.5 N.m so no load can be grater that that.

1st set of loads: 2, 4, 6, 8, 10 N.m

2nd set of loads: 3,5,7,12,14 N.m

3rd set of loads: 2,9,11,13,15 N.m

4th set of loads: 2,3,4,5,6 N.m

5th set of loads: 7,8,9,10,11 N.m

After the system reaches steady state for each defined value of load, the equations of 5.1.3 can be applied (each set must solve 1 system). During the transitions the time must be recorded so the change between state 1 and state 2 provides information to solve 5.1.5 (each set solves the system twice)

5.1.6. Matlab Program

Having the data (torque-n rpm) this information is computed in a Matlab function (motorvalues.m) that solves the system of 5 equations, 5 unknowns. The function can compute for any number of sets given that each set has 5 measures of torque and speed and 4 of time between states; the fact that 4 sets were chosen

is to achieve a good average without too many measurements. It receives Vf, a matrix nX5 of torque values (Tq), a matrix nX5 of speed rpm (n), and a matrix of nX4 (s), where n is the number of sets, 5 is the number of measurements per set for section 5.1.3, 4 is the number of measurements of time for equation 5.1.5; and Vf is phase voltage.

5.2. GENERAL SPECIFICATIONS

Machine	HP	Volts	Amps	RPM	Hz	PH	Poles	Des	Frame
Motor	2	208-230/460	6.8-6.3/3.15	1755	60	3	4	B	145T
Generator base rpm max rpm	3	230/460	 7.6/3.8 7.2/3.6	 1750 2625	 60 90	3	4	B	182TC

Table 5.2.1: specifications of the motors

Unidrive	Input volt	Input freq	PH	Peak current normal duty	Output Volt	Max. output curr. Heavy duty
SP1204	208-230 V	50-60 Hz	3	12.1 A	0-230 V	10.6 A
SP1203	208-230 V	50-60 Hz	2	10.5 A	0-230 V	7.5 A

Table 5.2.2: specifications of the drives (VFD)

5.3. CONVERSION OF TEST-BENCH FROM 480V TO 230V

The original configuration of the test-bench when it came from factory had only one derive for the dyno and left open the possibility to connect any start up mechanism for the motor. It also came configure for 480V.

Seatech doesn’t have that voltage in any of the labs (although it has the voltage intake in the electric room, there is no wiring from the electric room to any of the

labs for 480V) and the labs in Boca Raton with that voltage are occupied. It was decided to change the test-bench configuration to 230V, the motors and some of the components have that capacity. Some other components had to be purchased. A drive similar to the one of the dyno was added for the motor. The test-bench ended up as shown in appendix 8.7 and 8.8.

The drives have PI controlling and programming capabilities which allow the results of this project to be applied in to them.

6. TEST-BENCH EXPERIMENT RESULTS

The test-bench has installed 2 VFD drives (Control Techniques, Unidrive SP1204 and SP1203) in its main box; SP1204 controls the dyno and SP1203 controls the motor. Both are power up by the same power supply and have similar characteristics, except for the specifications of each motor that were configured in their specific drive (see table 6.1).

The original configuration sent by factory was stored in SP1404 since the test-bench had to be changed to from 480V to 230V (see section 5.3); that configuration of SP1404 was used as a reference and the most important parameters are presented in the table 6.1. The parameters of SP1204 were configured based on SP1404. Notice that the differences between SP1404 and SP1204 are only in the voltage and current ratings, all the other parameters remain the same; this is obvious since they were configured for the same dynamotor. A full documentation of SP1204 parameters was also done but only the parameters that differ from the default parameters are shown. The extraction of the parameters of SP1404 represented a challenge, the drive could only be energized by the test-bench itself. As a future reference if SP1404 needs to be power up again, one of the already installed drives (SP1203 preferably) has to be dismounted and SP1404 must be mounted instead. This is a tedious and time consuming process. Having a copy of the original parameters saves time.

Parameter	SP1404	SP1204	SP1203
0.07 (P)	0.03	0.03	0.01
0.08 (I)	0.1	0.1	4.5
0.09 (D)	0	0	0
0.14 (Torq select)	0	0	0
0.32 (max heavy duty curr rating, A)	5.8	10.6	9.3
0.37 (serial comm. Address)	1	1	2
0.38 (P curr gain)	32	32	-
0.39 (I curr gain)	615	615	-
0.40 (autotune)	0	0	-
0.42 (poles)	Auto	Auto	Auto
0.43 (power fact)	0.923	0.923	0.71
0.44 (rat voltage, V)	480	230	230
0.45 (rat speed, rpm)	1750	1750	1755
0.46 (rat current, A)	4.8	7.6	6.3
0.47 (rat freq, Hz)	60	60	60
0.49 (security status)	L2	L2	L2
0.48 (operat mode)	Cl vect	CL vect	CL vect
5.17 (stator resistance, Ω)	2.152	2.152	-
5.24 (transient inductance mH)	4.781	4.781	-
5.25 (stator inductance, mH)	394.86	394.86	-
5.29 (saturation breakpoint)	56	56	-
5.30 (saturation breakpoint)	81	81	-
4.08 (torque reference, N.m)	0	0	0

Table 6.1.1: configuration of Unidrives on test-bench

The values of the stator resistance and inductance of the dyno, extracted from the original configuration (SP1404), turn out to be very similar to the ones used in the simulations; since the difference between dyno and motor is not significant, it can be assume that the values of the motor are close and, therefore, the simulations are a good guide of the behavior of the motor. Notice also that the values of Kp and Ki that SP1404 had (those values were calculated in the factory by the drive itself since it has that capability) are very close to the ones calculated theoretically in this project; the model is therefore valid. The changing

of the speed reference worked well as it was shown by one parameter of the drive that showed the reference speed.

Several errors on the configuration of the drive SP1204 were fixed, the drive shows “ready” state in its display meaning that it’s ready to run the motor.

During the first experiments the dyno motor only reached 46 rpm and after a minute the thermal protection turned it off (very low speeds generate a fast increase in temperature, the drive had to shut down the motor to protect it). The total motor current was 12.44 A which falls in the range. The configuration of SP1404 and SP1204 was stored in two separated smartcards in case is needed in the future.

Programming on the motor VFD was done and the motor was energized; however for security reasons it was not started till the dyno could reach 1800 rpm (a very specific warning in Horlick manual (Horlick, 2009)). Table 6.1 shows some of the values that were configured for SP1203; the ones with a dash (-) weren’t tested.

Future work on this project has to speed up the dyno up to 1800 rpm and checks its velocity. Check also the proper change of the speed when the speed reference changes. The gains configured in the dyno are not likely to be changed; the ones that already have work fine.

With respect to the motor it is recommended to decouple the shaft (ask provider for guidance) and execute autotune function (see appendix 8.6.3) on SP1203; this way the drive itself can calculate the motor parameters.

7. CONCLUSIONS

Induction motors show their usefulness if properly controlled; a wide range of cruse velocities for the ship, an effective control of disturbances, and the possibility of run the motor at any torque with any speed can be derived from the approach explained in this project.

To control an induction motor, which in essence is non linear, with a PID controller model, which is mostly used for linear systems, requires reducing the error to the maximum; this can be achieved by creating a plant model that “predicts” the behavior of the motor reducing the error on the input signal of the controller.

Depending on the plant model chosen, the system will have different number of differential equations. The d-q model developed in this problem has 4 differential equations for the 4 fluxes, one for the torque and one for the mechanical speed; since the mechanical speed is the variable to control and the input is the electrical frequency a system of one input, one output is obtained. This plant model is useful and accurately describes the behavior of a real motor given the proper parameters; but, due to its non linearity, its control is difficult.

A system where a plant model predicts the behavior of the motor, an open loop system manages the changes of speed reference (rpm) and a PI control is in charge only of the perturbations is possible and it was proved to work.

The main difficulty, aside from the mathematical manipulation of the differential equations and the linearization of the model, is the tuning of the PI. The system is too sensitive to changes so special care must be taken in calibrating the PI gains. Different sets of gains for the PI controller can be found depending on the tuning method. A good compromise is a set of gains obtained mathematical but finely tuned manually.

It is also possible to use a different plant model, one involving Vector Field Control, it was shown that the behavior of the plant is the same as the one proposed initially with the great advantage that now the system has 2 outputs for one input; this is, it is theoretically possible to control both output variables just by controlling the internal currents of the motor, a more accurate, more flexible control system, although more complex to develop, is propose as future work.

Implementation of this project is possible in the test-bench platform; dyno and motor drives have the capability to control a motor using PID control and Vector Field Control, as well as the option to program in a PLC any other different configuration. The test-bench needs more tuning and a further usage of its capabilities is strongly suggested as a future project.

8. APPENDIXES

8.1. ELECTRIC MOTOR

An electric motor is a device using electrical energy to produce mechanical energy, nearly always by the interaction of magnetic fields and current-carrying conductors. The reverse process, using mechanical energy to produce electrical energy, is accomplished by a generator or dynamo. Traction motors used on vehicles often perform both tasks. (Wikipedia).

8.1.1. Categorization of Electric Motors

The classic division of electric motors has been that of Alternating Current (AC) types vs. Direct Current (DC) types. This is more a de facto convention, rather than a rigid distinction. For example, many classic DC motors run on AC power, these motors being referred to as universal motors.

Rated output power is also used to categorize motors, those of less than 746 Watts, for example, are often referred to as fractional horsepower motors (FHP) in reference to the old imperial measurement.

The ongoing trend toward electronic control further muddles the distinction, as modern drivers have moved the commutator out of the motor shell. For this new breed of motor, driver circuits are relied upon to generate sinusoidal AC drive

currents, or some approximation of. The two best examples are: the brushless DC motor and the stepping motor, both being poly-phase AC motors requiring external electronic control, although historically, stepping motors (such as for maritime and naval gyrocompass repeaters) were driven from DC switched by contacts.

Considering all rotating (or linear) electrical motors require synchronism between a moving magnetic field and a moving current sheet for average torque production, there is a clearer distinction between an asynchronous motor and synchronous types. An asynchronous motor requires slip between the moving magnetic field and a winding set to induce current in the winding set by mutual inductance; the most ubiquitous example being the common AC induction motor which must slip in order to generate torque. In the synchronous types, induction (or slip) is not a requisite for magnetic field or current production (e.g. permanent magnet motors, synchronous brush-less wound-rotor doubly-fed electric machine).

8.1.2. Torque Capability of Motor Types

When optimally designed for a given active current (i.e., torque current), voltage, pole-pair number, excitation frequency (i.e., synchronous speed), and core flux density, all categories of electrical motors or generators will exhibit virtually the same maximum continuous shaft torque (i.e., operating torque) within a given physical size of electromagnetic core. Some applications require bursts of torque beyond the maximum operating torque, such as short bursts of torque to

accelerate an electric vehicle from standstill. Always limited by magnetic core saturation or safe operating temperature rise and voltage, the capacity for torque bursts beyond the maximum operating torque differs significantly between categories of electrical motors or generators (Wikipedia).

8.2. INDUCTION MOTORS

8.2.1. Induction Machine

An induction motor (IM) is a type of alternating current motor where power is supplied to the rotating device by means of electromagnetic induction. It is also called asynchronous motor.

An induction motor is sometimes called a *rotating transformer* because the stator (stationary part) is essentially the primary side of the transformer and the rotor (rotating part) is the secondary side.

Induction motors are now the preferred choice for industrial motors due to their rugged construction, absence of brushes (which are required in most DC motors) and — thanks to modern power electronics — the ability to control the speed of the motor (Wikipedia, Induction motor).

8.2.2. Principle of operation and comparison to synchronous motors

The basic difference between an induction motor and a synchronous AC motor is that in the latter a current is supplied onto the rotor. This then creates a magnetic field which, through magnetic interaction, links to the rotating magnetic field in the

stator which in turn causes the rotor to turn. It is called synchronous because at steady state the speed of the rotor is the same as the speed of the rotating magnetic field in the stator.

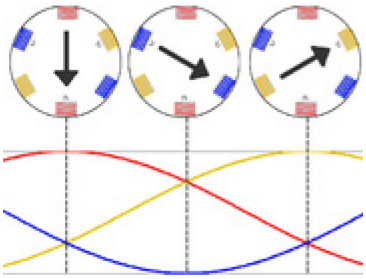

Figure 8.2.1: 3-phase power supply provides a rotating magnetic field in an induction motor

By way of contrast, the induction motor does not have any direct supply onto the rotor; instead, a secondary current is induced in the rotor. To achieve this, stator windings are arranged around the rotor so that when energized with a polyphase supply they create a rotating magnetic field pattern which sweeps past the rotor. This changing magnetic field pattern induces current in the rotor conductors. These currents interact with the rotating magnetic field created by the stator and in effect cause a rotational motion on the rotor.

However, for these currents to be induced, the speed of the physical rotor and the speed of the rotating magnetic field in the stator must be different, or else the magnetic field will not be moving relative to the rotor conductors and no currents will be induced. If by some chance this happens, the rotor typically slows slightly until a current is re-induced and then the rotor continues as before. This difference between the speed of the rotor and speed of the rotating magnetic field in the stator is called *slip*. It is dimensionless and is the ratio between the

relative speeds of the magnetic field as seen by the rotor (the *slip speed*) to the speed of the rotating stator field. Due to this an induction motor is sometimes referred to as an asynchronous machine (Wikipedia, Induction motor).

8.3. LOOKUP TABLE

Input	Output	
Desired RPM	Frequency	
0		
15	pi*1	3.1416
30	pi*2	6.2832
45	pi*3	9.4248
60	pi*4	12.566368
75	pi*5	15.707968
90	pi*6	18.849568
105	pi*7	21.991168
120	pi*8	25.132768
135	pi*9	28.274368
149	pi*10	31.415968
164	pi*11	34.557568
179	pi*12	37.699168
194	pi*13	40.840768
209	pi*14	43.982368
224	pi*15	47.123968
239	pi*16	50.265568
253	pi*17	53.407168
268	pi*18	56.548768
283	pi*19	59.690368
298	pi*20	62.831968
313	pi*21	65.973568
327	pi*22	69.115168
342	pi*23	72.256768
357	pi*24	75.398368
372	pi*25	78.539968
386	pi*26	81.681568
401	pi*27	84.823168
416	pi*28	87.964768
431	pi*29	91.106368
445	pi*30	94.247968
460	pi*31	97.389568
475	pi*32	100.531168
489	pi*33	103.672768
504	pi*34	106.814368
519	pi*35	109.955968
536	pi*36	113.097568
548	pi*37	116.239168
562	pi*38	119.380768
577	pi*39	122.522368
591	pi*40	125.663968
606	pi*41	128.805568
621	pi*42	131.947168
635	pi*43	135.088768
650	pi*44	138.230368
664	pi*45	141.371968
679	pi*46	144.513568
693	pi*47	147.655168
708	pi*48	150.796768
722	pi*49	153.938368
737	pi*50	157.079968
751	pi*51	160.221568
765	pi*52	163.363168
780	pi*53	166.504768
794	pi*54	169.646368
809	pi*55	172.787968
823	pi*56	175.929568
837	pi*57	179.071168
852	pi*58	182.212768
866	pi*59	185.354368
881	pi*60	188.495968
895	pi*61	191.637568
909	pi*62	194.779168
924	pi*63	197.920768
938	pi*64	201.062368
952	pi*65	204.203968
966	pi*66	207.345568
981	pi*67	210.487168
995	pi*68	213.628768
1009	pi*69	216.770368
1023	pi*70	219.911968
1037	pi*71	223.053568

1051	pi*72	226.195168
1066	pi*73	229.336768
1080	pi*74	232.478368
1094	pi*75	235.619968
1109	pi*76	238.761568
1123	pi*77	241.903168
1134	pi*78	245.044768
1151	pi*79	248.186368
1165	pi*80	251.327968
1179	pi*81	254.469568
1193	pi*82	257.611168
1207	pi*83	260.752768
1221	pi*84	263.894368
1235	pi*85	267.035968
1250	pi*86	270.177568
1264	pi*87	273.319168
1278	pi*88	276.460768
1292	pi*89	279.602368
1306	pi*90	282.743968
1320	pi*91	285.885568
1334	pi*92	289.027168
1348	pi*93	292.168768
1362	pi*94	295.310368
1375	pi*95	298.451968
1389	pi*96	301.593568
1402	pi*97	304.735168
1417	pi*98	307.876768
1431	pi*99	311.018368
1445	pi*100	314.159968
1459	pi*101	317.301568
1473	pi*102	320.443168
1486	pi*103	323.584768
1500	pi*104	326.726368
1514	pi*105	329.867968
1528	pi*106	333.009568
1542	pi*107	336.151168
1555	pi*108	339.292768
1569	pi*109	342.434368
1583	pi*110	345.575968
1597	pi*111	348.717568
1610	pi*112	351.859168
1624	pi*123	355.000768
1638	pi*114	358.142368
1651	pi*115	361.283968
1665	pi*116	364.425568
1679	pi*117	367.567168
1692	pi*118	370.708768
1706	pi*119	373.850368
1719	2*pi*60	376.991968
1733	pi*121	380.133568
1746	pi*122	383.275168
1759	pi*123	386.416768
1772	pi*124	389.558368
1784	pi*125	392.699968
1796	pi*126	395.841568
1806	pi*127	398.983168
1817	pi*128	402.124768

Table 8.3.1: lookup table.

8.4. CURVES OF TORQUE VS. N VARYING FE

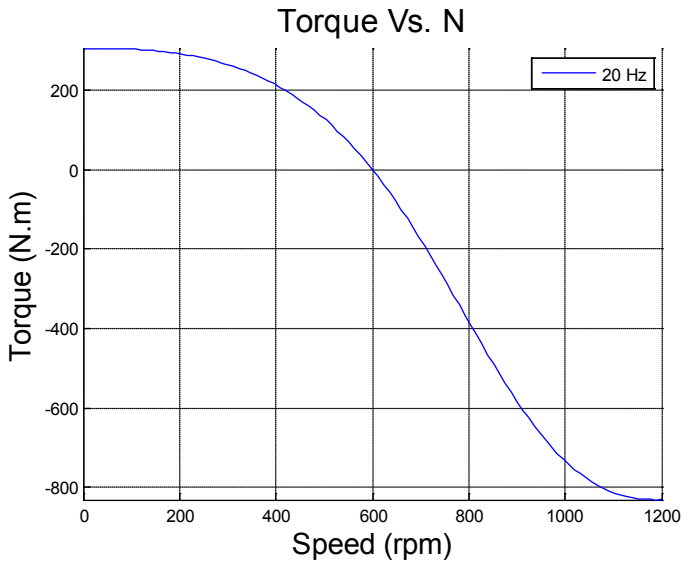

Figure 8.4.1: torque (N.m) vs. speed (rpm) for fe=20 Hz

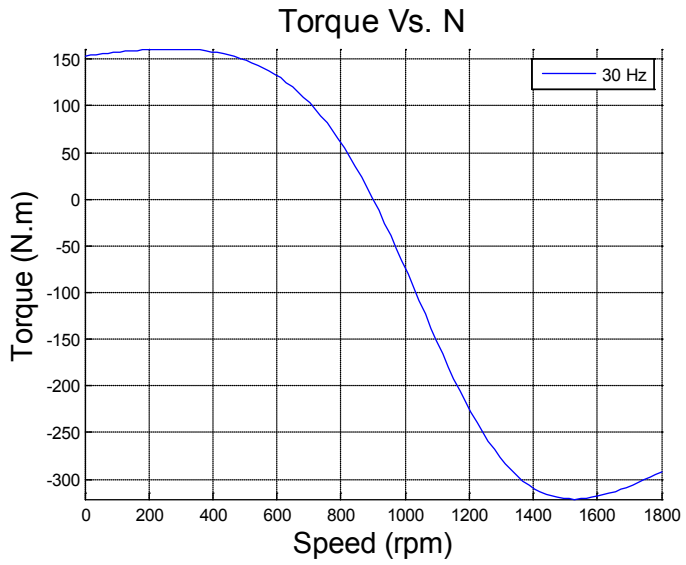

Figure 8.4.2: torque (N.m) vs. speed (rpm) for fe=30 Hz

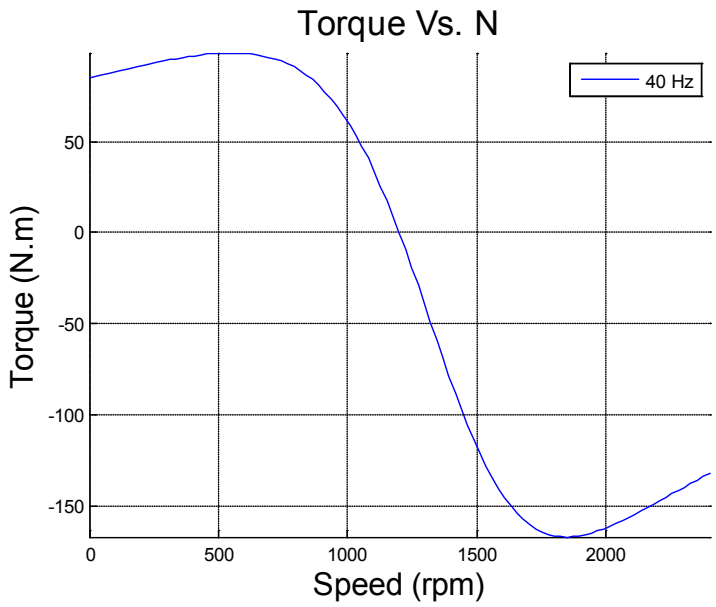

Figure 8.4.3: torque (N.m) vs. speed (rpm) for fe=40 Hz

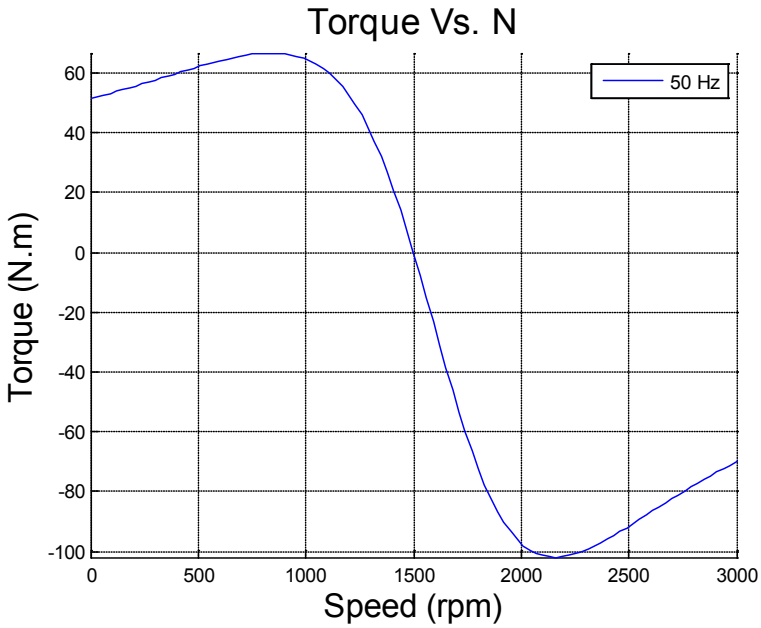

Figure 8.4.4: torque (N.m) vs. speed (rpm) for fe=50 Hz

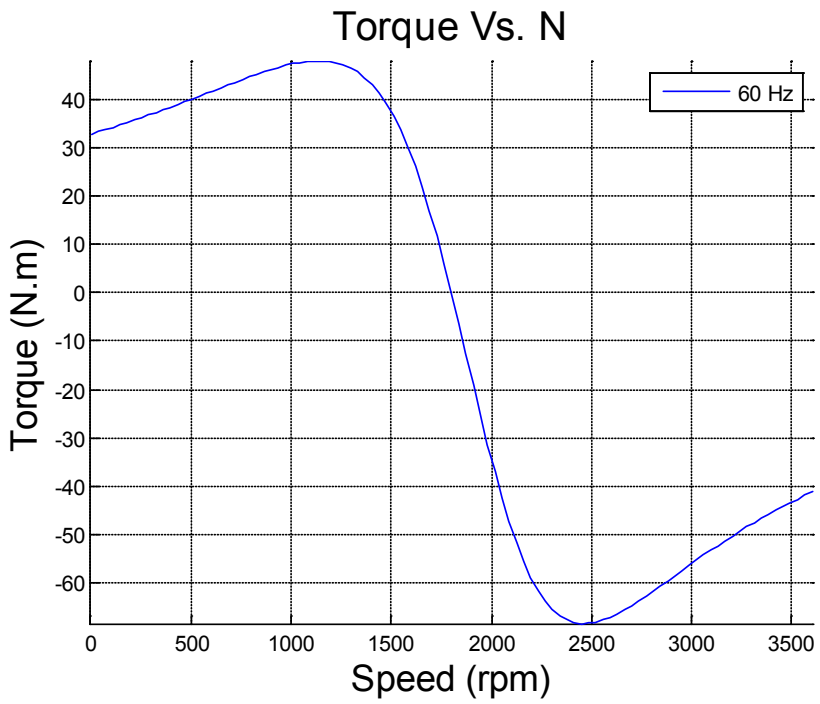

Figure 8.4.5: torque (N.m) vs. speed (rpm) for fe=60 Hz

8.5. PARTIAL DERIVATIVES OF TORQUE EQUATION

$\frac{\partial T_m}{\partial \omega_{el}}$ and $\frac{\partial T_m}{\partial \omega_m}$ can be found from equation 3.4.13.

Define the constant

$$A = 3{V_\phi}^2 \left(\frac{L_m}{L_s+L_m}\right)^2 R_{2r} \tag{8.5.1}$$

The torque equation can be re-written as:

$$T(\omega_{el}, \omega_m) = \frac{A}{s\omega_s\left[\left(R_{th}+{R_r}/{s}\right)^2+\omega_e^2(L_s+L_r)^2\right]} \tag{8.5.2}$$

Redefine R_{th} as:

$$R_{th} = R_s \left(\frac{L_m}{L_s + L_m}\right)^2 \tag{8.5.3}$$

notice that $R_{th} = R_{TH}$ from equation 3.4.9, since the frequency of the reactance does not affect the value of the equivalent resistance. In other words, the equivalent resistance does not depend on f_e and can be considered as a constant.

For simplicity in the calculations the denominator is defined as:

$$D = s\omega_s \left[\left(R_{th} + {R_r}/{s}\right)^2 + \omega_{el}^2 (L_s + L_r)^2\right] \tag{8.5.4}$$

or, after some replacements.

$$D = (\omega_s - \omega_m) \left[\left(R_{th} + {R_r}/{s}\right)^2 + \left(\frac{P\omega_s}{2}\right)^2 (L_s + L_r)^2\right] \tag{8.5.5}$$

So the partial derivative of the torque with respect to the electrical frequency is:

$$\frac{\partial T_m}{\partial \omega_{el}} = A \left[\frac{-\frac{\partial D}{\partial \omega_{el}}}{D^2}\right] \tag{8.5.6}$$

Applying chain rule

$$\frac{\partial D}{\partial \omega_{el}} = \frac{\partial D}{\partial \omega_s} \frac{\partial \omega_s}{\partial \omega_{el}} \tag{8.5.7}$$

As seen in equation 3. 5.1

$$\omega_s = {}^{2\omega_{el}}/_{P} \tag{8.5.8}$$

So $\frac{\partial \omega_s}{\partial \omega_{el}} = {}^{2}/_{P}$

Differentiating D with respect to ws leads to:

$$\frac{\partial D}{\partial \omega_s} = \frac{\partial\left[(\omega_s-\omega_m)\left(R_{th}+{}^{R_r}/_{s}\right)^2\right]}{\partial \omega_s} + \frac{\partial\left[(\omega_s-\omega_m)\left(\frac{P\omega_s}{2}\right)^2(L_s+L_r)^2\right]}{\partial \omega_s} = \frac{\partial D_1}{\partial \omega_s} + \frac{\partial D_2}{\partial \omega_s} \tag{8.5.9}$$

Starting with D1

$$\frac{\partial D_1}{\partial \omega_s}$$

$$= \frac{(\omega_s{}^2 - 2\omega_s\omega_m)R_r{}^2 + (2\omega_s{}^2 - 4\omega_s\omega_m + 2\omega_m{}^2)R_rR_{th} + (\omega_s{}^2 - 2\omega_s\omega_m + \omega_m{}^2)R_{th}{}^2}{\omega_s{}^2 - 2\omega_s\omega_m + \omega_m{}^2}$$

(8.5.10)

Forming the perfect squared binomials leads to:

$$\frac{\partial D_1}{\partial \omega_s} = \frac{(\omega_s{}^2-\omega_m)R_r{}^2+(\omega_s-\omega_m)^2R_rR_{th}+(\omega_s-\omega_m)^2R_{th}{}^2}{(\omega_s-\omega_m)^2}$$

Adding some terms to form the binomial of $R_r{}^2$

$$\frac{\partial D_1}{\partial \omega_s} = \frac{(\omega_s{}^2-\omega_m+\omega_m{}^2)R_r{}^2+(\omega_s-\omega_m)^2R_rR_{th}+(\omega_s-\omega_m)^2R_{th}{}^2-\omega_m{}^2R_r{}^2}{(\omega_s-\omega_m)^2}$$

$$\frac{\partial D_1}{\partial \omega_s} = \frac{(\omega_s-\omega_m)^2R_r{}^2+(\omega_s-\omega_m)^2R_{2r}R_{th}+(\omega_s-\omega_m)^2R_{th}{}^2-\omega_m{}^2R_r{}^2}{(\omega_s-\omega_m)^2}$$

$$\frac{\partial D_1}{\partial \omega_s} = \frac{(\omega_s-\omega_m)^2(R_r+R_{th})^2-{\omega_m}^2{R_r}^2}{(\omega_s-\omega_m)^2}$$

$$\frac{\partial D_1}{\partial \omega_s} = (R_r + R_{th})^2 - \left(\frac{\omega_m R_r}{\omega_s-\omega_m}\right)^2 \quad (8.5.11)$$

Differentiating D2:

$$\frac{\partial D_2}{\partial \omega_s} = \left(3\left(\frac{P\omega_s}{2}\right)^2 - 2\left(\frac{P}{2}\right)^2 \omega_s\omega_m\right)(L_s + L_r)^2 \quad (8.5.12)$$

$$\frac{\partial D_2}{\partial \omega_s} = \omega_s\left(\frac{P}{2}\right)^2 (3\omega_s - 2\omega_m)(L_s + L_r)^2 \quad (8.5.13)$$

The partial derivative of the torque with respect to electrical frequency is then:

$$\frac{\partial T_m}{\partial \omega_{el}} = A\left\{\frac{-\left(\frac{2}{P}\right)\left[(R_r+R_{th})^2-\left(\frac{\omega_m R_r}{\omega_s-\omega_m}\right)^2+\omega_s\left(\frac{P}{2}\right)^2(3\omega_s-2\omega_m)(L_s+L_r)^2\right]}{\left[(\omega_s-\omega_m)\left[\left(R_{th}+{R_r}/{s}\right)^2+\left(\frac{P\omega_s}{2}\right)^2(L_s+L_r)^2\right]\right]^2}\right\} \quad (8.5.14)$$

The same procedure can be applied to the partial derivative of the torque with respect to the mechanical speed.

$$\frac{\partial T_m}{\partial \omega_m} = A\left[\frac{-\frac{\partial D}{\partial \omega_m}}{D^2}\right] \quad (8.5.15)$$

$$\frac{\partial D}{\partial \omega_m} = \frac{\partial D_1}{\partial \omega_m} + \frac{\partial D_2}{\partial \omega_m}$$

$$\frac{\partial D_1}{\partial \omega_m} = \frac{{\omega_s}^2{R_r}^2-\left({\omega_s}^2-2\omega_s\omega_m+{\omega_m}^2\right){R_{th}}^2}{{\omega_s}^2-2\omega_s\omega_m+{\omega_m}^2}$$

$$\frac{\partial D_1}{\partial \omega_m} = \frac{{\omega_s}^2{R_r}^2-(\omega_s-\omega_m)^2{R_{th}}^2}{(\omega_s-\omega_m)^2}$$

$$\frac{\partial D_1}{\partial \omega_m} = \left(\frac{\omega_s R_r}{\omega_s - \omega_m}\right)^2 - {R_{th}}^2$$

$$\frac{\partial D_1}{\partial \omega_m} = \left(\frac{R_r}{s}\right)^2 - {R_{th}}^2 \qquad (8.5.16)$$

$$\frac{\partial D_2}{\partial \omega_m} = -\left(\frac{P\omega_s}{2}\right)^2 (L_s + L_r)^2 \qquad (8.5.17)$$

The partial derivative of the torque with respect to the mechanical speed is:

$$\frac{\partial T_m}{\partial \omega_m} = A\left\{\frac{{R_{th}}^2 - \left(\frac{R_r}{s}\right)^2 + \left(\frac{P\omega_s}{2}\right)^2 (L_s + L_r)^2}{\left[(\omega_s - \omega_m)\left[\left(R_{th} + {R_r}/{s}\right)^2 + \left(\frac{P\omega_s}{2}\right)^2 (L_s + L_r)^2\right]\right]^2}\right\} \qquad (8.5.18)$$

In both partial derivatives are synchronous speed and slip terms; however both depend only on the electrical frequency and the mechanical speed; other terms are used to reduce the length of the expressions or to show them in a simpler form.

8.6. TEST BENCH PROCEDURES

8.6.1 Test-Bench Startup Procedure

Taken from Horlik manual (Horlick, 2009).

8.6.1.1. Special Directions for Operating Dynamometer Test Stand Site Selection

The dynamometer test stand is designed for installation inside a building. The site selected for the installation of this equipment must be free from excessive moisture. An ideal installation would be a concrete pad with a level surface on which the dynamometer would be placed. If the surface is not level, then it will be

necessary to insert shims under the base of the dynamometer to avoid distortion of its frame.

8.6.1.2. On-Site Balancing and Alignment

The dynamometer has been aligned at the factory. Proper alignment should be verified at the site prior to start-up.

8.6.1.3. Equipment Ground

The frame of the dynamometer stand and the control panel must be solidly connected to a low resistance path to ground before energization.

8.6.1.4. Wiring the Dynamometer

The dynamometer test stand must be wired in exact accordance with the wiring diagram (Horlick Manual). Prior to initial startup, verify that the electrical installation is correct. A source of power should be brought to the dynamometer motor and a separate source of power should be brought to the motor under test. The motor under test should also be equipped with a motor start as illustrated on the electric schematic. There are no safety features to protect against faulty installation wiring.

8.6.1.5. Starting/Operating the Dynamometer

- After the wiring of the test stand has been checked for wiring accuracy, follow this procedure to start the system.
- Position the Main Disconnect Switch to the “On” position

- Depress the “Dyno Motor Start” push button
- Once the system is started, the variable frequency drive connected to the dynamometer will automatically ramp the system speed to 1800 RPM. The system speed is displayed on the torque meter.
- Once the system has reached 1800 RPM, the motor under test can be started. This motor can be started either locally at the main control panel or locally at the remote motor starter.

  Note: the dynamometer must be running and up to speed prior to Starting the motor under test. Under no conditions, should the Motor under test be started without the dynamometer running At 1800 rpm.
- Once the motor under test is started, its torque can be tested by varying the speed of the dynamometer motor. There is a potentiometer on the front of the control enclosure that is used for speed control. As you lower the speed of the system, the torque will increase on the motor under test. The system speed should not be lowered below the slip speed of the motor under test.
- Once testing is complete, the motor under test should be stopped.
- Once it is verified that the motor under test is no longer running, the dynamometer motor can be stopped by depressing the “Dyno Motor Stop” push button.
- When work is complete, position the Main Disconnect Switch to the “Off” position.
- *Danger:*

- Do not operate test stand unless enclosure of control cubicle and frame of motor-generator are solidly connected to ground.
- Do not operate test stand with protective covers removed.
- To prevent shock hazard, interrupt the line voltage supplying the test stand prior to performing trouble-shooting or maintenance procedures.

8.6.2. Communications Connection Procedure

Taken from Control's Techniques manual (Control Techniques, 2007)

- Acquire USB to EIA485 interface described in section 8.6.2.1.
- Install CTSoft in your PC, section 8.6.2.4.
- Check serial communication parameters using the keypad, leave the default values, section 8.6.2.2. and 8.6.2.3.
- Upload or download parameters, section 8.6.2.5.
- Monitor the Unidrive if requires, section 8.6.2.6. or 8.6.2.7.

8.6.2.1. Serial Communications Introduction

The Unidrive SP has a standard 2-wire EIA485 interface (serial communications interface) which enables all drive set-up, operation and monitoring to be carried out with a PC or controller if required. Therefore, it is possible to control the drive entirely by serial communications without the need for a SM-keypad or other control cabling. The drive supports two protocols selected by parameter configuration:

- Modbus RTU

- CT ANSI

Modbus RTU has been set as the default protocol, as it is used with the PC-tools commissioning/start-up software as provided on the CD ROM. The serial communications port of the drive is a RJ45 socket, which is isolated from the power stage and the other control terminals (see section 4.12 *Serial communications connections* on page 84 for connection and isolation details, User guide).

The communications port applies a 2 unit load to the communications network.

USB/EIA232 to EIA485 Communications

An external USB/EIA232 hardware interface such as a PC cannot be used directly with the 2-wire EIA485 interface of the drive. Therefore a suitable converter is required.

Suitable USB to EIA485 and EIA232 to EIA485 isolated converters are available from Control Techniques as follows:

- CT USB Comms cable (CT Part No. 4500-0096)
- CT EIA232 Comms cable (CT Part No. 4500-0087)

When using one of the above converters or any other suitable converter with the Unidrive SP, it is recommended that no terminating resistors be connected on the network. It may be necessary to 'link out' the terminating resistor within the converter depending on which type is used. The information on how to link out

the terminating resistor will normally be contained in the user information supplied with the converter.

8.6.2.2. Configuration of Serial Communication Parameters Using the Keypad

It is recommended to leave the default parameters (section 5.2.3). However, in order to change them:

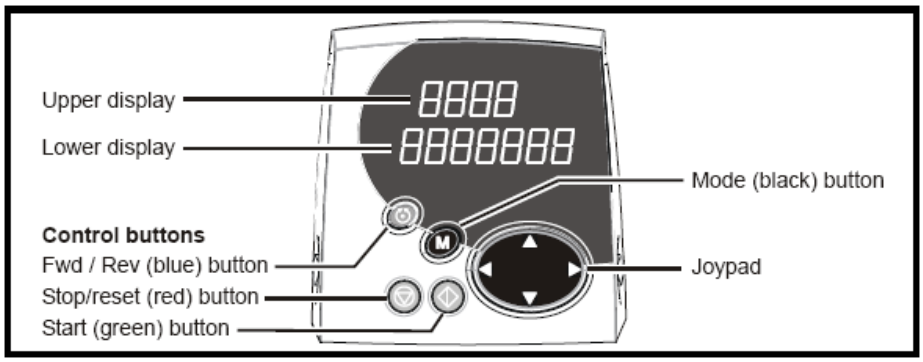

Figure 8.6.1: Unidrive display

Control buttons

The keypad consists of:

1. Joypad - used to navigate the parameter structure and change parameter values.

2. Mode button - used to change between the display modes – parameter view, parameter edit, status.

3. Three control buttons - used to control the drive if keypad mode is selected.

4. Help button (SM-Keypad Plus only) - displays text briefly describing the selected parameter. The Help button toggles between other display modes and parameter help mode. The up and down functions on the joypad scroll the help text to allow the whole string to be viewed. The right and left functions on the joypad have no function when help text is being viewed.

The display examples in this section show the SM-Keypad 7 segment LED display.

Saving parameters

When changing a parameter in Menu 0, the new value is saved when pressing the Mode button to return to parameter view mode from parameter edit mode.

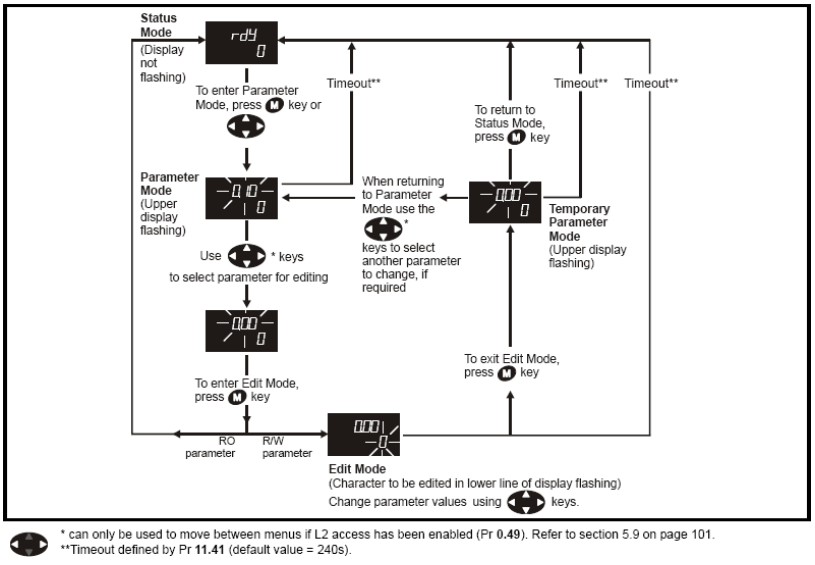

Figure 8.6.2: flow chart of Unidrive menus

If parameters have been changed in the advanced menus, then the change will not be saved automatically. A save function must be carried out.

Procedure:

Enter 1000* in Pr. xx.00

Either:

• Press the red reset button

• Toggle the reset digital input

• Carry out a drive reset through serial communications by setting Pr 10.38 to 100 (ensure that Pr. xx.00 returns to 0).

*If the drive is in the under voltage trip state or is being supplied from a low voltage DC supply, a value of 1001 must be entered into Pr xx.00 to perform a save function.

8.6.2.3. Serial Communications Set-Up Parameters

The following parameters need to be set according to the system requirements. In most of the cases the default parameters will work for the purpose of this project.

0.35 {11.24}	Serial mode								
RW	Txt							US	
⇳	AnSI (0) rtU (1)				⇨	rtU (1)			

Table 8.6.1: serial mode

This parameter defines the communications protocol used by the 485 comms port on the drive. This parameter can be changed via the drive keypad, via a Solutions Module or via the comms interface itself. If it is changed via the comms

interface, the response to the command uses the original protocol. The master should wait at least 20ms before send a new message using the new protocol. (Note: ANSI uses 7 data bits, 1 stop bit and even parity; Modbus RTU uses 8 data bits, 2 stops bits and no parity.)

Comms value	String	Communications mode
0	AnSI	ANSI
1	rtU	Modbus RTU protocol
2	Lcd	Modbus RTU protocol, but with an SM-Keypad Plus only

Table 8.6.2: serial mode parameters

ANSIx3.28 protocol

Modbus RTU protocol, but with an SM-Keypad Plus only

This setting is used for disabling communications access when the SMSM-Keypad Plus is used as a hardware key.

0.36 {11.25}	Serial communications baud rate								
RW	Txt							US	
⇳	300 (0), 600 (1), 1200 (2), 2400 (3), 4800 (4), 9600 (5), 19200 (6), 38400 (7), 57600 (8)*, 115200 (9)*				⇨	19200 (6)			

Table 8.6.3: serial communications baud rate

* only applicable to Modbus RTU mode

This parameter can be changed via the drive keypad, via a Solutions Module or via the comms interface itself. If it is changed via the comms interface, the

response to the command uses the original baud rate. The master should wait at least 20ms before sending a new message using the new baud rate.

Note:

When using the CT EIA232 Comms cable the available baud rate is limited to 19.2k baud.

0.37 {11.23}	Serial communications address								
RW	Txt							US	
⇳	0 to 247				⇨	1			

Table 8.6.4: serial communications address

Used to define the unique address for the drive for the serial interface. The drive is always a slave.

Modbus RTU

When the Modbus RTU protocol is used addresses between 0 and 247 are permitted. Address 0 is used to globally address all slaves, and so this address should not be set in this parameter.

ANSI

When the ANSI protocol is used the first digit is the group and the second digit is the address within a group. The maximum permitted group number is 9 and the maximum permitted address within a group is 9. Therefore, Pr 0.37 is limited to 99 in this mode. The value 00 is used to globally address all slaves on the

system, and x0 is used to address all slaves of group x, therefore these addresses should not be set in this parameter.

8.6.2.4. Ctsoft

CTSoft is a Windows™ based software commissioning/start-up tool for Unidrive SP and other Control Techniques products. CTSoft can be used for commissioning/start-up and monitoring, drive parameters can be uploaded, downloaded and compared, and simple or custom menu listings can be created. Drive menus can be displayed in standard list format or as live block diagrams. CTSoft is able to communicate with a single drive or a network. CTSoft can be found on the CD which is supplied with the drive and is also available for download from www.controltechniques.com (file size approximately 25MB).

CTSoft system requirements:

- Windows 2000/XP/Vista. Windows 95/98/98SE/ME/NT4 and
- Windows 2003 server are NOT supported
- Internet Explorer V5.0 or later must be installed
- Minimum of 800x600 screen resolution with 256 colors. 1024x768 is recommended.
- 128MB RAM
- Pentium III 500MHz or better recommended.

- Adobe Acrobat Reader 5.1 or later (for parameter help).
- Microsoft.Net Frameworks 2.0
- Note that you must have administrator rights to install CTSoft.

To install CTSoft from the CD, insert the CD and the auto-run facility should start up the front-end screen from which CTSoft can be selected. Any previous copy of CTSoft should be uninstalled before proceeding with the installation (existing projects will not be lost). Included with CTSoft are the user guides for the supported drive models. When help on a particular parameter is request by the user, CTSoft links to the parameter in the relevant advanced user guide.

8.6.2.5. Uploading and Downloading with CTSoft

CTSoft allows you to change, download or upload different parameters to the Unidrive.

In order to download parameters from Unidrive simply click on the “download parameters” bottom. When downloading parameters from Unidrive it gathers all the values for each parameter on each menu (figure 8.6.3).

To change or modify parameters first edit them (depending on the menu or the parameter, check page 106 of user guide for a list of basic parameters) in the subfolder “parameters” and then upload them into the Unidrive.

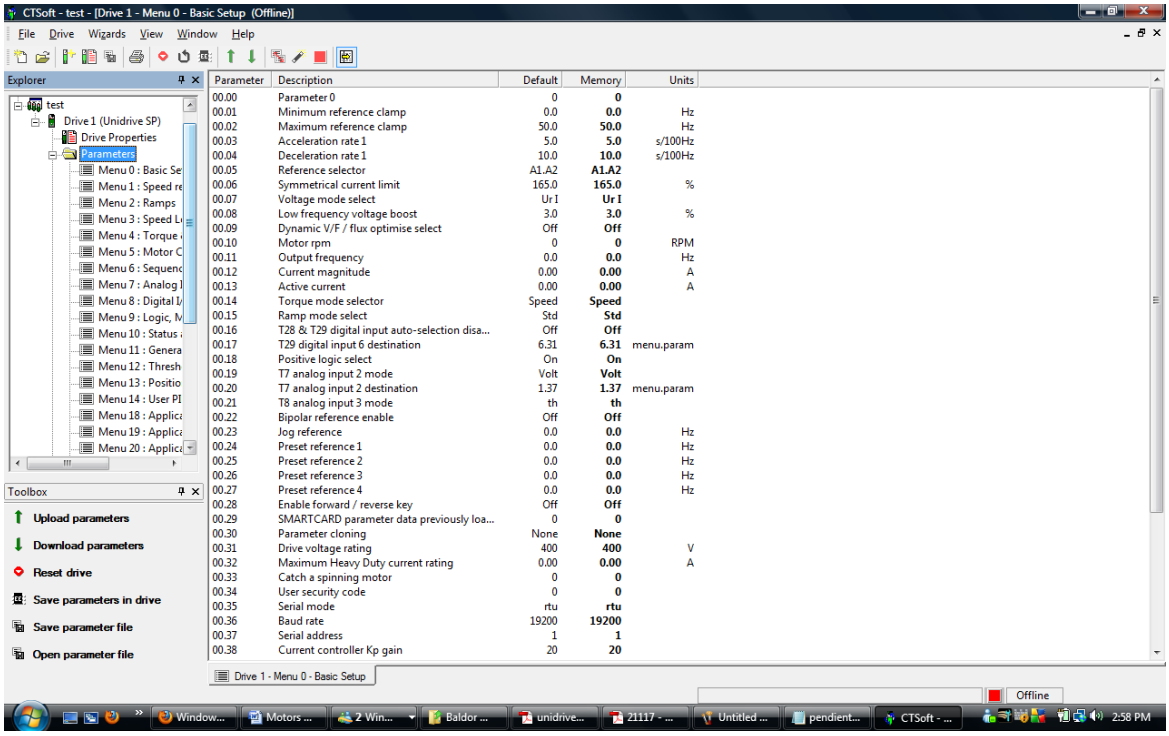

Figure 8.6.3: CTSoft

8.6.2.6. Monitoring Unidrive

0.10 {5.04}	Estimated motor speed								
RO	Bit	FI				NC	PT		
OL	⇕	±180,000 rpm			⇨				

Table 8.6.5: estimated motor speed

Open-loop

Pr 0.10 (5.04) indicates the value of motor speed that is estimated from the following:

0.12 Post-ramp frequency reference

0.42 Motor - no. of poles.

0.10 {3.02}	Motor speed								
RO	Bi	FI				NC	PT		
VT	⇳	±Speed_max rpm			⇨				

Table 8.6.6: motor speed

Closed-loop

Pr 0.10 (3.02) indicates the value of motor speed that is obtained from the speed feedback.

0.11 {5.01}	Drive output frequency								
RO	Bi	FI				NC	PT		
OL	⇳	±SPEED_FREQ_MAX Hz			⇨				
VT	⇳	±1250.0 Hz			⇨				

Table 8.6.7: drive output frequency

Open-loop & closed loop vector

Pr 0.11 displays the frequency at the drive output.

0.11 {3.29}	Drive encoder position								
RO	Uni	FI				NC	PT		
SV	⇳	0 to 65,535 $1/2^{16}$ths of a revolution			⇨				

Table 8.6.8: drive encoder position

8.6.2.7. CTScope

CTScope is a full featured software oscilloscope for viewing and analyzing changing values within the drive. The time base can be set to give high speed capture for tuning or intermittent capture for longer term trends. The interface is based on a traditional oscilloscope, making it familiar to engineers across the globe.

CTScope is free of charge and can be obtained from:

http://www.emersonct.com/products/ac_drives/unidrive_sp_high_performance/software.aspx.

8.6.3. Running the motors

Taken from Control Techniques user manual (Control Techniques, 2007)

- Check motor parameters using the keypad, section 8.6.3.1.
- Modify or adjust parameters accordingly, same steps than section 8.6.2.2.
- If motor parameters are not available, see auto-tuning, section 8.6.3.4 or perform the identifying motor parameters procedure of section 5.1.
- Check or select operational mode, section 8.6.3.2.
- Check basic requirements, section 8.6.3.3.
- Set the proper speed reference, section 8.6.3.5.
- Start the motors with the start bottom of the main panel.

8.6.3.1. Motor parameters

0.42 {5.11}	No. of motor poles								
RW	Txt							US	
OL	⇳	0 to 60 (Auto to 120 Pole)			⇨	Auto (0)			
CL	⇳				⇨	VT	Auto (0)		
						SV	6 POLE (3)		

Table 8.6.9: number of motor poles

Open-loop

This parameter is used in the calculation of motor speed, and in applying the correct slip compensation. When auto is selected, the number of motor poles is automatically calculated from the rated frequency (Pr 0.47) and the rated full load rpm (Pr 0.45). The number of poles = 120 * rated frequency / rpm rounded to the nearest even number.

Closed-loop vector

This parameter must be set correctly for the vector control algorithms to operate correctly. When auto is selected, the number of motor poles is automatically calculated from the rated frequency (Pr 0.47) and the rated full load rpm (Pr 0.45). The number of poles = 120 * rated frequency /rpm rounded to the nearest even number.

0.43 {5.10}	Motor rated power factor								
RW	Uni							US	
OL ⇳	0.000 to 1.000				⇨	0.850			
VT ⇳					⇨				

Table 8.6.10: motor rated power factor

The power factor is the true power factor of the motor, i.e. the angle between the motor voltage and current.

Open-loop

The power factor is used in conjunction with the motor rated current (Pr 0.46) to calculate the rated active current and magnetizing current of the motor. The rated active current is used extensively to control the drive, and the magnetizing current is used in vector mode Rs compensation. It is important that this parameter is set up correctly.

This parameter is obtained by the drive during a rotational autotune. If a stationary autotune is carried out, then the nameplate value should be entered in Pr 0.43.

Closed-loop vector

If the stator inductance (Pr 5.25) contains a non-zero value, the power factor used by the drive is continuously calculated and used in the vector control algorithms (this will not update Pr 0.43).

If the stator inductance is set to zero (Pr 5.25) then the power factor written in Pr 0.43 is used in conjunction with the motor rated current and other motor parameters to calculate the rated active and magnetizing currents which are used in the vector control algorithm.

This parameter is obtained by the drive during a rotational autotune. If a stationary autotune is carried out, then the nameplate value should be entered in Pr 0.43.

0.43 {3.25}		Encoder phase angle							
RW	Uni							US	
SV	⇳	0.0 to 359.9°			⇨	0.0			

Table 8.6.11: encoder phase angle

The phase angle between the rotor flux in a servo motor and the encoder position is required for the motor to operate correctly. If the phase angle is known it can be set in this parameter by the user.

Alternatively the drive can automatically measure the phase angle by performing a phasing test (see autotune in servo mode Pr 0.40). When the test is complete the new value is written to this parameter. The encoder phase angle can be modified at any time and becomes effective immediately. This parameter has a factory default value of 0.0, but is not affected when defaults are loaded by the user.

0.44 {5.09}	Motor rated voltage								
RW	Uni				RA			US	
⇳	0 to AC_VOLTAGE_SET_MAX V				⇨	200V drive: 230 400V drive: EUR> 400 USA> 460 575V drive: 575 690V drive: 690			

Table 8.6.12: motor rated voltage

Open-loop & Closed-loop vector

Enter the value from the rating plate of the motor.

0.45 {5.08}	Motor rated full load speed (rpm)								
RW	Uni							US	
OL	⇳	0 to 180,000 rpm			⇨	EUR> 1,500 USA> 1,800			
VT	⇳	0.00 to 40,000.00 rpm			⇨	EUR> 1,450.00 USA> 1,770.00			

Table 8.6.13: motor rated full load speed (rpm)

Open-loop

This is the speed at which the motor would rotate when supplied with its base frequency at rated voltage, under rated load conditions (=synchronous speed - slip speed). Entering the correct value into this parameter allows the drive to increase the output frequency as a function of load in order to compensate for this speed drop.

Slip compensation is disabled if Pr 0.45 is set to 0 or to synchronous speed, or if Pr 5.27 is set to 0. If slip compensation is required this parameter should be set

to the value from the rating plate of the motor, which should give the correct rpm for a hot machine. Sometimes it will be necessary to adjust this when the drive is commissioned because the nameplate value may be inaccurate.

Slip compensation will operate correctly both below base speed and within the field weakening region. Slip compensation is normally used to correct for the motor speed to prevent speed variation with load. The rated load rpm can be set higher than synchronous speed to deliberately introduce speed droop. This can be useful to aid load sharing with mechanically coupled motors.

Closed loop vector

Rated load rpm is used with motor rated frequency to determine the full load slip of the motor which is used by the vector control algorithm.

Incorrect setting of this parameter can result in the following:

- Reduced efficiency of motor operation
- Reduction of maximum torque available from the motor
- Failure to reach maximum speed
- Over-current trips
- Reduced transient performance
- Inaccurate control of absolute torque in torque control modes

The nameplate value is normally the value for a hot machine, however, some adjustment may be required when the drive is commissioned if the nameplate value is inaccurate.

The rated full load rpm can be optimized by the drive (For further information, refer to section 8.1.3 Closed loop vector motor control on page 135, user guide).

0.46 {5.07}	Motor rated current								
RW	Uni				RA			US	
⇳	0 to Rated_current_max A				⇨	Drive rated current [**11.32**]			

Table 8.6.14: motor rated current

Enter the name-plate value for the motor rated current.

0.47 {5.06}	Rated frequency								
RW	Uni							US	
OL	⇳	0 to 3,000.0Hz			⇨	EUR> 50.0, USA> 60.0			
VT	⇳	0 to 1,250.0Hz			⇨	EUR> 50.0, USA> 60.0			

Table 8.6.15: rated frequency

<u>Open-loop & Closed-loop vector</u>

Enter the value from the rating plate of the motor.

8.6.3.2. Operating-Mode Selection

0.48 {11.31}	Operating mode selector								
RW	Txt	NC					PT		
⇳	1 to 4				⇨	OL	1		
						VT	2		
						SV	3		

Table 8.6.16: operating mode selector

The settings for Pr 0.48 are as follows:

Setting		Operating mode
OPEn LP	1	Open-loop
CL VECt	2	Closed-loop vector
SerVO	3	Servo
rEgEn	4	Regen

Table 8.6.17: operating mode selector settings

This parameter defines the drive operating mode. Pr xx.00 must be set to 1253 (European defaults) or 1254 (USA defaults) before this parameter can be changed. When the drive is reset to implement any change in this parameter, the default settings of all parameters will be set according to the drive operating mode selected and saved in memory.

8.6.3.3. Quick Start Connections

8.6.3.3.1 Basic Requirements

This section shows the basic connections which must be made for the drive to run in the required mode. For minimal parameter settings to run in each mode

please see the relevant part of section 7.3 Quick Start commissioning/start-up, page 122 (user guide (Control Techniques, 2007)).

Drive control method	Requirements
Terminal mode	Drive Enable Speed reference Run forward or run reverse command
Keypad mode	Drive Enable
Serial communications	Drive Enable Serial communications link

Table 8.6.18: min. connection requirements for each control mode

Operating mode	Requirements
Open loop mode	Induction motor
Closed loop vector - RFC mode	Induction motor
Closed loop vector mode	Induction motor with speed feedback
Closed loop servo mode	Permanent magnet motor with speed and position feedback

Table 8.6.19: min. connection requirements for each mode of operation

Speed feedback

Suitable devices are:

- Incremental encoder (A, B or F, D with or without Z)
- Incremental encoder with forward and reverse outputs (F, R with or without Z)

- SINCOS encoder (with, or without Stegmann Hiperface, EnDat or SSI communications protocols)
- EnDat absolute encoder

8.6.3.4 Autotune

0.40 {5.12}		Autotune							
RW	Uni								
OL	⇳	0 to 2			⇨	0			
VT	⇳	0 to 4			⇨	0			
SV	⇳	0 to 6			⇨	0			

Table 8.6.20: autotune

Open-Loop

There are two autotune tests available in open loop mode, a stationary and a rotating test. A rotating autotune should be used whenever possible, so the measured value of power factor of the motor is used by the drive.

- The stationary autotune can be used when the motor is loaded and it is not possible to remove the load from the motor shaft.
- A rotating autotune first performs a stationary autotune, before rotating the motor at 2/3 base speed in the forward direction for several seconds. The motor must be free from load for the rotating autotune.

To perform an autotune, set Pr **0.40** to 1 for a stationary test or 2 for a rotating test, and provide the drive with both an enable signal (on terminal 31) and a run signal (on terminal 26 or 27).

Following the completion of an autotune test the drive will go into the inhibit state. The drive must be placed into a controlled disable condition before the drive can be made to run at the required reference. The drive can be put in to a controlled disable condition by removing the SAFE TORQUE OFF (SECURE DISABLE) signal from terminal 31, setting the drive enable parameter Pr **6.15** to OFF (0) or disabling the drive via the control word (Pr **6.42** & Pr **6.43**). For further information refer to section *Pr 0.40 {5.12} Autotune* on page 130 user guide (Control Techniques, 2007).

Closed-loop

There are three autotune tests available in closed loop vector mode, a stationary test, a rotating test and an inertia measurement test. A stationary autotune will give moderate performance whereas a rotating autotune will give improved performance as it measures the actual values of the motor parameters required by the drive. An inertia measurement test should be performed separately to a stationary or rotating autotune.

- The stationary autotune can be used when the motor is loaded and it is not possible to remove the load from the motor shaft.

- A rotating autotune first performs a stationary autotune, before rotating the motor at 2/3 base speed in the forward direction for approximately 30 seconds. The motor must be free from load for the rotating autotune.
- The inertia measurement test can measure the total inertia of the load and the motor. This is used to set the speed loop gains (see *Speed loop gains*, user guide) and to provide torque feed forwards when required during acceleration. During the inertia measurement test the motor speed changes from 1/3 to 2/3 rated speed in the forward direction several times. The motor can be loaded with a constant torque load and still give an accurate result, however, non-linear loads and loads that change with speed will cause measurement errors.

To perform an autotune, set Pr **0.40** to 1 for a stationary test, 2 for a rotating test, or 3 for an inertia measurement test and provide the drive with both an enable signal (on terminal 31) and a run signal (on terminal 26 or 27).

Following the completion of an autotune test the drive will go into the inhibit state. The drive must be placed into a controlled disable condition before the drive can be made to run at the required reference. The drive can be put in to a controlled disable condition by removing the SAFE TORQUE OFF (SECURE DISABLE) signal from terminal 31, setting the drive enable parameter Pr **6.15** to OFF (0) or disabling the drive via the control word (Pr **6.42** & Pr **6.43**).

Setting Pr **0.40** to 4 will cause the drive to calculate the current loop gains based on the previously measured values of motor resistance and inductance. The

drive does apply any voltage to the motor during this test. The drive will change Pr **0.40** back to 0 as soon as the calculations are complete (approximately 500ms). For further information refer to section *Pr 0.40 {5.12} Autotune* on page 133, user guide (Control Techniques, 2007).

8.6.3.5 Speed Reference

8.6.3.5.1 Speed Limits

Open-loop

Set Pr 0.01 at the required minimum output frequency of the drive for both directions of rotation. The drive speed reference is scaled between Pr 0.01 and Pr 0.02. [0.01] is a nominal value; slip compensation may cause the actual frequency to be higher.

0.01 {1.07}		**Minimum reference clamp**								
RW		Bi						PT	US	
OL	⇳	±3,000.0Hz			⇨	0.0				
CL	⇳	±SPEED_LIMIT_MAX Hz/rpm			⇨	0.0				

Table 8.6.21: minimum reference clamp.

When the drive is jogging, [0.01] has no effect.

Closed-loop

Set Pr 0.01 at the required minimum motor speed for both directions of rotation. The drive speed reference is scaled between Pr 0.01 and Pr 0.02.

0.02 {1.06}		Maximum reference clamp							
RW		Uni						US	
OL	⇳	0 to 3,000.0Hz			⇨	EUR> 50.0 USA> 60.0			
CL	⇳	SPEED_LIMIT_MAX Hz/rpm			⇨	VT	EUR> 1,500.0 USA> 1,800.0		
						SV	3,000.0		

Table 8.6.22: maximum reference clamp.

The drive has additional over-speed protection

Open-loop

Set Pr 0.02 at the required maximum output frequency for both directions of rotation. The drive speed reference is scaled between Pr 0.01 and Pr 0.02. [0.02] is a nominal value; slip compensation may cause the actual frequency to be higher.

Closed-loop

Set Pr 0.02 at the required maximum motor speed for both directions of rotation. The drive speed reference is scaled between Pr 0.01 and Pr 0.02.

For operating at high speeds see section 8.6 High speed operation on page 141, user guide.

8.6.3.5.2. Reaching the Speed.

Open-loop

When the drive is enabled with Pr 0.33 = 0, the output frequency starts at zero and ramps to the required reference. When the drive is enabled when Pr 0.33

has a non-zero value, the drive performs a start-up test to determine the motor speed and then sets the initial output frequency to the synchronous frequency of the motor. Restrictions may be placed on the frequencies detected by the drive as follows:

0.33 {6.09}	Catch a spinning motor								
RW	Uni							US	
OL	⇳	0 to 3			⇨	0			

Table 8.6.23: catch a spinning motor

Pr 0.33	Function
0	Disabled
1	Detect all frequencies
2	Detect positive frequencies only
3	Detect negative frequencies only

Table 8.6.24: catch a spinning motor settings

0.33 {5.16}	Rated rpm autotune								
RW	Uni							US	
VT	⇳	0 to 2			⇨	0			

Table 8.6.25: rated rpm autotune

Closed-loop vector

The motor rated full load rpm parameter (Pr 0.45) in conjunction with the motor rated frequency parameter (Pr 0.46) defines the full load slip of the motor. The slip is used in the motor model for closed-loop vector control.

The full load slip of the motor varies with rotor resistance which can vary significantly with motor temperature. When Pr 0.33 is set to 1 or 2, the drive can automatically sense if the value of slip defined by Pr 0.45 and Pr 0.46 has been set incorrectly or has varied with motor temperature. If the value is incorrect parameter Pr 0.45 is automatically adjusted. The adjusted value in Pr 0.45 is not saved at power-down. If the new value is required at the next power-up it must be saved by the user.

Automatic optimization is only enabled when the speed is above 12.5% of rated speed, and when the load on the motor load rises above 62.5% rated load. Optimization is disabled again if the load falls below 50% of rated load.

For best optimization results the correct values of stator resistance (Pr 5.17), transient inductance (Pr 5.24), stator inductance (Pr 5.25) and saturation breakpoints (Pr 5.29, Pr 5.30) should be stored in the relevant parameters. These values can be obtained by the drive during an autotune (see Pr 0.40 for further details).

Rated rpm auto-tune is not available if the drive is not using external position/speed feedback.

The gain of the optimizer, and hence the speed with which it converges, can be set at a normal low level when Pr 0.33 is set to 1. If this parameter is set to 2 the gain is increased by a factor of 16 to give faster convergence.

8.7. TEST-BENCH CONVERSION TO 230V

A table with the components that were changed or reconfigured is shown below:

Component/Voltage	480V	230V
Dyno drive	SP1404 (480V)	SP1204 (230V)
Motor drive	-	SP1203 (230V)*
Line regenerator	Wired for 480V	Rewired for 230V
Fuses (1-3)	8A	15A
Fuses (4-5)	4A	8A
Dynamotor and motor	Wired for 480V	Rewired for 230V
Transformer	Taped for 480V	Re-taped for 230V

Table 8.7.1: changed components of test-bench

* this component wasn't in the original configuration; it was added later as part of the project.

8.8. TEST-BENCH MODIFICATIONS AFTER CONVERSION

The connections made in the test-bench and changes of the original configuration are documented below. Compare them with page 2 and 3 of the schematics section (pp 9 and 10) (Horlick, 2009), for further reference.

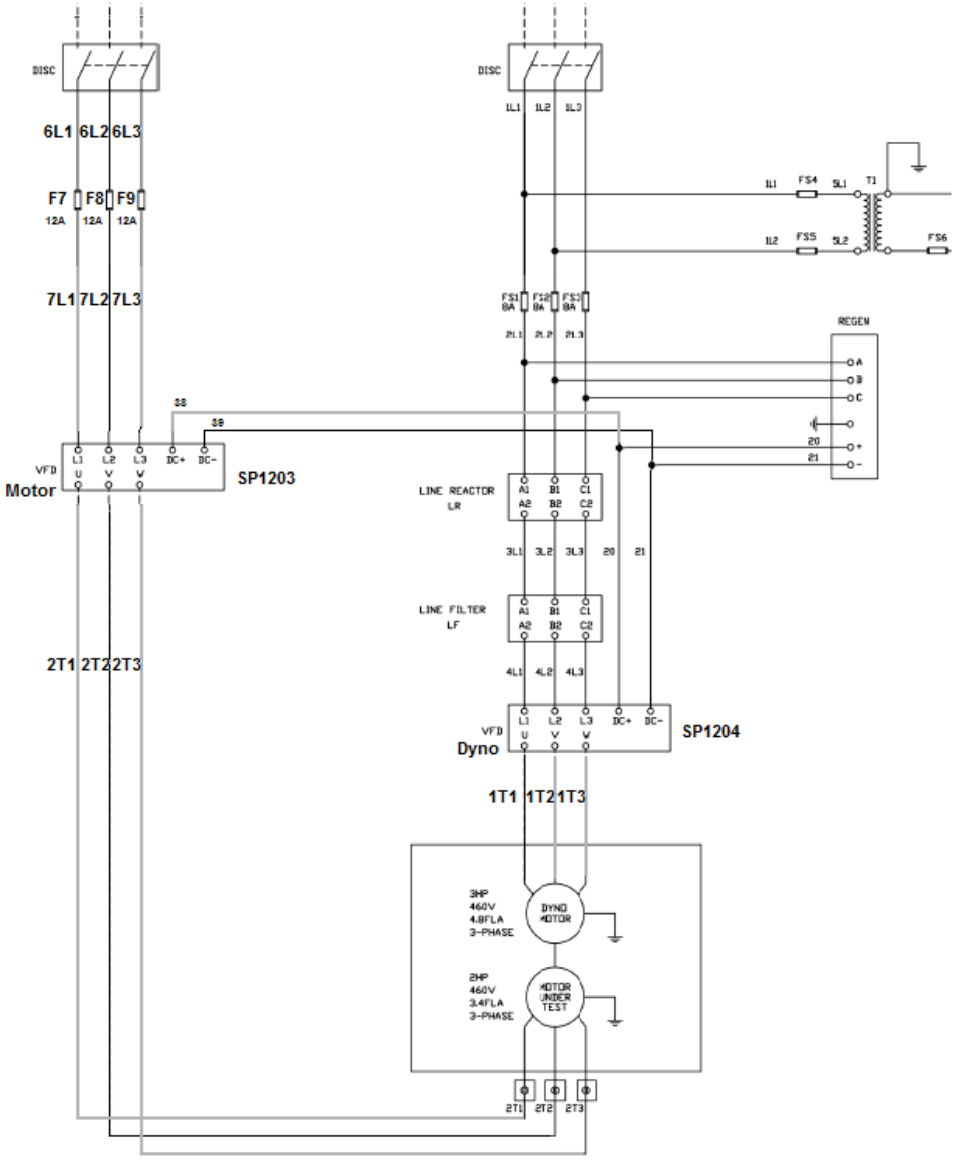

Figure 8.8.1: modified main control enclosure sheet 1

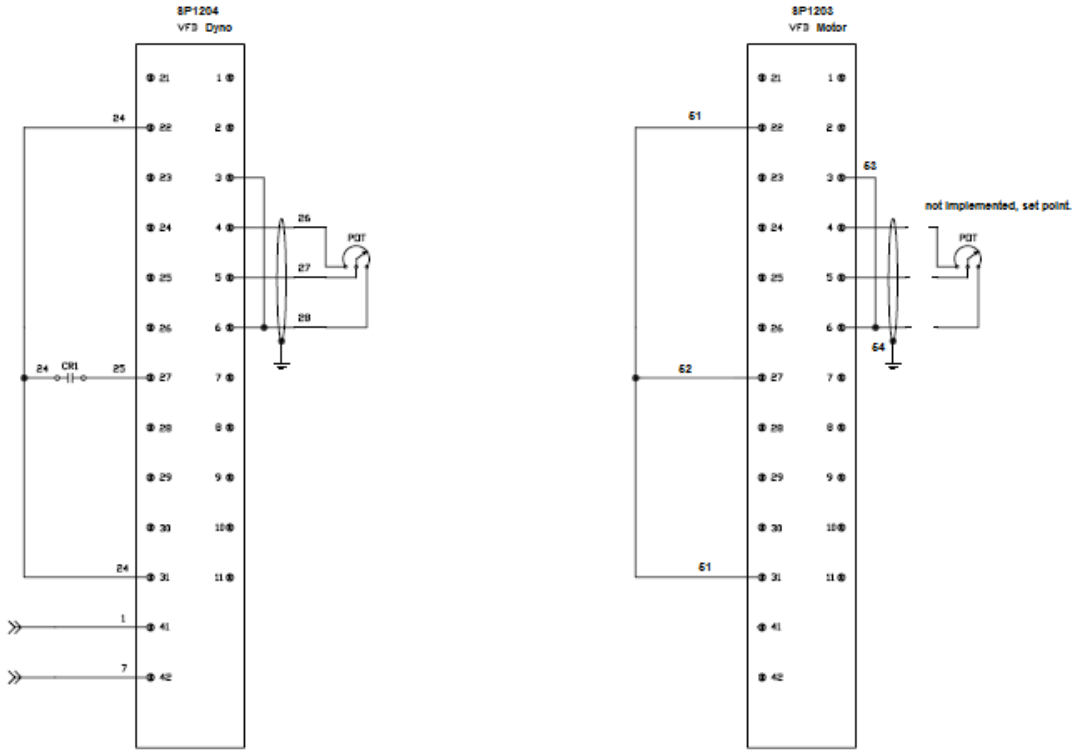

Figure 8.8.2: modified main control enclosure sheet 2

BIBLIOGRAPHY

Bose, B. K. (2002). *Modern Power Electronics and AC Drives.* New York: Prentice-Hall.

Carlton, J. (1994). *Marine Propellers and Propulsion.* Oxford: Butterworth & Heinmann.

Chapman, S. (2005). *Electric machinery fundamentals.* New York: Mc Graw-Hill.

Control Techniques. (2007). *User Guide Unidrive Models size 0 to 6.* Emerson Industrial Automation.

De Doncker, R. N. (1989). The universal Field oriented controller applied to tapped stator windings induction motors. *University of Wisconsin*.

Denal, M. A. (2002). Fuzzy and neural control of an induction motor. *University of Science and Technology of Oran, Algeria* , Vol.12, No.2, pp 221–233.

Figari, M. a. (2007). Dynamic behavior and stability of marine propulsion systems. *Naval Architecture and Marine Engineering (DINAV), Università di Genova.*

Fitzgerald, R. a. (2003). *Electric Machinery.* New York: McGraw-Hill.

Horlick, C. (2009). *3 HP Submital Documentation job # 21117.* Horlick Co. Inc.

Krause, P. W. (2002). Analysis of Electric Machinery. *IEEE Pres*.

Kuo, B. (1987). *Automatic Control Systems.* NJ: Prentice-Hall.

Machat, M. (2005). Neural networks in motor control. *Dept. of power electrical and electronic engineering*.

Matlab. (2008). Asynchronous machine blocks SimPowerSystems help. *Matlab R2008a, version 7.6.*

Mohan, N. U. (1995). *Power Electronics: Converters, Applications, and Design.* New York: John Wiley & Sons.

Novotney, D. W. (1986). Introduction to Field Orientation and High Performance AC Drives. *IEEE IAS Tutorial Course*.

Ohm, D. Y. (2000). Dynamic model of induction motors for vector control. *Drivetech, Inc.*

Tempo, R. B. (1996). Robustness analysis with real Parametric Uncertainty. *Contrl Engineering Handbook* , pp 495-505.

Trzynadlowski, A. M. (1994). The Field Orientation Principle in Control of Induction Motors. *Kluwer Academic Publishers*.

Wikipedia. (n.d.). *Electric motor.* Retrieved 6 29, 2010, from Wikipedia: http://en.wikipedia.org/wiki/Electric_motor

Wikipedia. (n.d.). *Induction motor.* Retrieved 6 29, 2010, from Wikipedia: http://en.wikipedia.org/wiki/Induction_motor

Wikipedia. (n.d.). *PID controller.* Retrieved 6 29, 2010, from Wikipedia: ttp://en.wikipedia.org/wiki/PID_controller

Xiros, N. (2004). PID marine engine speed regulation under full load conditions for sensitivity H∞-norm specifications against propeller disturbance. *Proceedings of the Institute of Marine Engineering, Science and Technology. Part A, Journal of marine engineering and technology.* , no5, pp. 3-11.

Zaiping, C. a. (2003). Neural networks based electric motor drive for transportation systems. *Intelligent Transportation Systems. IEEE.* , vol.2., pp. 1378- 1383.

Zhaoming, Z. M. (2007). FPGA-Based Smart Induction Motor Controller Design. *Nios II Embedded Processor Design Contest, Yuan Ze University*.