ADDIS ABABA UNIVERSITY

ADDIS ABABA INSTITUTE OF TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

Investigation of Saturation Effect in Induction Machine Parameter Using LabVIEW CompactRIO

Kirubel Rorisa Seifu

A Thesis submitted to the School of Graduate Studies of the Addis Ababa University in a partial Fulfillment of the Requirement for the Degree of Master of Science in Control Engineering

Advisor: Dr. Mengesha M.

August, 2019
Addis Ababa

ADDIS ABABA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

Investigation of Saturation Effect in Induction Machine Parameter Using LabVIEW CompactRIO

By: Kirubel Rorisa Seifu

APPROVED BY BOARD OF EXAMINERS

Dr. Mengesha Mamo Advisor	_____ Signature	_____ Date
_____ Internal Examiner	_____ Signature	_____ Date
_____ External Examiner	_____ Signature	_____ Date
_____ Dean, School of Electrical and Computer Engineering	_____ Signature	_____ Date

DECLARATION

I, the undersigned, declare that this thesis is my original work and has not been presented to any university for similar or any other degree award. All sources of materials used for this thesis have been fully acknowledged.

Name – Kirubel Rorisa

Signature - _____
kirubelr321@gmail.com

Place – Addis Ababa University
Addis Ababa Institute of Technology
School of Graduate Studies
School of Electrical and Computer Engineering
Control Engineering
Addis Ababa
Ethiopia

Date - August, 2019

This thesis is performed under the supervision of my advisor, Dr. Mengesha Mamo, and submitted with his approval.

Advisor's Name – Dr. Mengesha Mamo

Signature - _____

Date - August, 2019

ACKNOWLEDGEMENT

First and foremost, I am thankful for all the opportunities as well as the challenges that I encountered throughout my post graduate studies.

My gratitude goes to my advisor Dr. Mengesha Mamo for inspiring me to do a thesis on this interesting topic.

ABSTRACT

Magnetic saturation in electrical machines has major influence on operational characteristics of the machine. To apply precise control mechanism on an induction machine, the development of accurate and reliable knowledge of parameter variation in induction machine's operating region during magnetic saturation is needed.

This thesis deals with bringing an alternative approach on investigation of the impact of magnetic saturation, on the steady-state operation of the controlled induction motor, using CompactRIO (a powerful control and acquisition system introduced by national instrument). The reason behind the selection of CompactRIO is because it is a rugged, reconfigurable embedded system that can be used for reliable stand-alone control applications. Control engineers can be benefited from FPGA technology without any knowledge of hardware description language, because CompactRIO can be developed using LabVIEW development platform.

The core essence of the presented work is to obtain required steady-state torque with minimal stator current. The influence of the mutual inductance mismatch (difference between the value of parameter L_m used in the controller and the real value of L_m in the machine) due to saturation is analytically verified in a simple mathematical-model modification that reduces the influence on the drive performance. Methods for the selection of magnetizing reference current are provided and effectiveness and accuracy of this investigation method is verified by LabVIEW simulation.

Key Words: Saturation Effect, Induction Machine, LabVIEW, CompactRIO

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AC	Alternating Current
ADC	Analogue Digital Converter
ASIC	Application Specific Integrated Circuit
cRIO	Compact Reconfigurable Input Output
DC	Direct Current
DIO	Digital Output
DQ	Direct Quadrature
FOC	Field Oriented Control
FPGA	Field Programmable Gate Array
IM	Induction Machine
LabVIEW	Laboratory Virtual Instrument Engineering Workbench
NI	National Instrument
PC	Personal Computer
PWM	Pulse Width Modulation
RFOC	Rotor Field Oriented Control
RIO	Reconfigurable Input-Output
SRAM	Static Random Access Memory
VI	Virtual Instrument

CHAPTER ONE

1. INTRODUCTION

1.1 Background

It is widely recognized that induction motor is becoming the main actuator for industrial purposes. Indeed, as compared to the DC machine, its maintenance is simpler (as it includes no mechanical commutators) and its cost is relatively cheaper. However, the problem of controlling induction motor is more complex because of its nonlinear nature.

The problem of induction motor control and observation has been given a great deal of interest over the last decade. Basically there are two control approaches of AC drives, namely scalar control and vector control. Scalar control is the term used to describe a simpler form of AC motor control. It is characterized by the adjustable magnitude of stator voltages and frequency in such a way that the air gap flux is always maintained at the desired value at the steady-state. It is a simpler form of motor control but is not adequate for high performance drives. Vector control (field oriented control) approach however, is used where a precise control of AC motors is required.

Direct field oriented control and indirect field oriented control are the two approaches to vector control. In the case of direct field oriented control, the rotation angle of the i_{qs}^{e} vector with respect to the stator flux is being directly determined (e.g. by measuring air gap flux) but in the indirect field oriented control technique the rotor angle is being measured indirectly, such as by measuring slip speed

The indirect field-oriented control scheme for induction machines has gained substantial popularity during last few years due to its relative simplicity, compared to other methods which require either main flux and stator currents or stator voltages and currents sensing for main flux calculation. The main problem which arises in the application of this method is the variation of induction machine parameters [1].

The estimation of induction motor performance traditionally based on constant parameter models has yielded good engineering results. However it has been observed that some of the machine parameters may not be treated as constant due to saturation in magnetic circuit. Further, it is

observed that highly saturated conditions may affect the static and dynamic performance of induction motor drives [2]. Therefore, Parameter variation effects in field oriented induction machines can cause significant performance deterioration. Due to this reason accurate control of induction machine requires sufficient knowledge on the parameters variation in all possible operating conditions. The essence of this study is to investigate the core saturation and resulting inductance characteristics for variation of the main electrical operating parameters being the magnitude of stator and rotor current and their resulting phase angle. In other words the main purpose of this study is to analyze effects caused by mismatch between actual value of magnetizing inductance and the value used by the controller. Actual effects of variable saturation level on machine's performance of the drive are assessed.

1.2 Statement of the Problem

Variation of induction machine parameters during magnetic saturation has brought difficulties in the application of modern control schemes. If a vector controlled induction motor drive is assigned to operate in all operating regions, it becomes an imperative that parameter variation due to main flux saturation is considered.

A mismatch between machine parameter used in a controller and the actual machine parameter which occurs due to saturation causes significant performance deterioration.

The most frequently applied control approach towards addressing parameter variation due to magnetic flux saturation needs knowledge of model reference adaptive control (MRAC) with the combination of MATLAB software. However, as an engineer there is a need to provide alternative mechanisms where professionals and any other individuals can choose, as per their interest, for investigating and understanding parameter variation as a result of saturation.

1.3 Objectives

1.1.1 General Objective

The general objective of this thesis work is to introduce CompactRIO based method of investigation for the possible parameter variation of induction machine, working at a steady state, with the consideration of magnetic saturation of iron core.

1.1.2 Specific Objectives

The thesis has the following specific objectives.

1. To show the limitation in performance of induction machine during magnetic saturation.
2. To model induction motor considering saturation of iron core using labVIEW cRIO.
3. To implement indirect vector control algorithm of induction motor considering saturation effect.

1.4 Scope and limitation of the study

This thesis analyses the parameter variation in induction machine using the national instrument CompactRIO and LabVIEW software.

Although there seems to be abundant resources on internet regarding the software and the instrument, no clear clue was put to direct a novice how he/she can start modeling and controlling an induction machine. There were several difficulties encountered in the way to find and to integrate all the necessary toolkits and add-ons of LabVIEW software relating to the focus area of the thesis. Much time was spent understanding the software and the working principle of the national instrument product, (such as cRIO, LabVIEW, LabVIEW FPGA, LabVIEW Real-Time, LabVIEW Motor and Simulation).

This serious lack of information and guide has limited the preparation of this thesis to the envisaged level.

CHAPTER TWO

2. LITERATURE REVIEW

2.1 Introduction

Magnetic saturation in electrical machines is of major influence on the operating characteristics. For induction machines, the assumption of linear magnetic behavior of the iron core in classical machine models may be adequate for the calculation of steady-state rated operation [3].

The total leakage inductance of an inductance motor is much smaller at standstill compared to operation at full-load or pull-out torque. This is partly due to the reduction of the slot inductance in the rotor bars as a result of the skin effect at higher frequencies of the rotor current. Another important factor is the magnetic saturation of the material affecting only those inductances describing magnetic flux within the iron core. Especially in the case of machine transients under fault conditions (e.g. three or two phase short-circuits) or following out-of-phase synchronization with machine currents sometimes exceeding 20 pu, the reduction in the magnetic presence as a result of occurring saturation is the major factor for reduced leakage inductances leading to higher currents [3].

Numerous methods of sensor-less vector control of induction machines are nowadays available. The majority of methods rely in the process of speed estimation on utilization of the induction machine model. As mechanical sub-systems of the control part of the drive and of the machine itself are in any sensor-less drive effectively decoupled, and as standard d-q axis constant parameter induction machine model is utilized, accuracy of speed estimation strongly depends on parameter variation effects in the machine [4].

Compensation of parameter variation effects in sensor-less vector controlled induction motor drives with model based speed estimation has been a subject of considerable interest in the past two decades. In majority of cases attempts are made to provide on-line identification of either stator resistance or rotor resistance. As changes in stator and rotor resistance are temperature dependent, the only way to compensate for their variation is to employ a method of on-line identification [4].

2.2 Induction Machine

2.2.1 Definition

Induction motors are often termed the 'Work horse of the industry'. This is because it is one of the most widely used motors in the world. It is used in transportation and industries, and also in household appliances as a means of converting electric power to mechanical work [5].

Induction Motors account for more than 85% of all motors used in industry and domestic applications. This motor is also called as asynchronous motor because it runs at a speed less than its synchronous speed. There are basically two types of induction motor that depend upon the input supply - single phase induction motor and three phase induction motor. Single phase induction motor is not a self-starting motor but three phase induction motor is a self-starting motor. The speed of the rotor will depend upon the AC supply and the speed can be controlled by varying the input supply [6].

The three-phase induction motors are most commonly used in industrial applications. In particular, the squirrel-Cage induction motors are widely used in home and industrial applications because these machines are very economical, rugged and reliable [6]. Thus the three phase induction motor is Self-starting, less armature reaction, robust in construction, economical and easier to maintain. The advantages of much motor are:

- Simple in design, rugged, low-price and easy to maintain
- They have wide range of power ratings.
- They run essentially at constant speed from no-load to full load
- Its speed depends on the frequency of the power source
- Requires a variable-frequency power-electronic drive for optimal speed control

The speed of induction motor depends on various factors, like supplied voltage, frequency, number of poles. There are different methods for controlling the speed from stator side and rotor side.

An induction motor has two main parts. A Stationary Stator: This has stator winding in them which induces a magnetism which causes the Rotor part of the motor to revolve.

A Rotor: This is the revolving part in the induction motor that produces torque with respect to slip.

The windings in the rotor are short circuited, unlike the windings in the stator which are independent to the three phases.

There are two types of rotors which can be placed inside a stator; cage rotor and wound rotor.

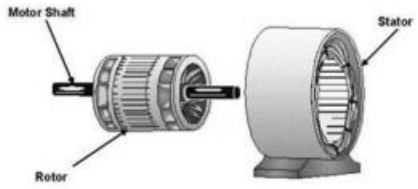

Figure 2-1: Rotor and stator of Induction Motor

An induction motor generally works when a three phase AC power is applied to the induction motor and its rotor starts rotating. When a three phase supply is connected to an induction motor, it is generally connected to the three phase stator field winding. This induces set of currents flowing inside the stator. These windings start behaving as electro magnets and produce three alternating and independent magnetic fluxes. The resultant of these fluxes is called the net flux [7]. The magnitude remains constant but position keeps on changing in circular manner.

2.2.2 Relation of rotor speed and slip

If rotor runs at the synchronous speed, which is the same speed of the rotating magnetic field, then the rotor will appear stationary to the rotating magnetic field and the rotating magnetic field will not cut the rotor and no induced current will flow in the rotor. Thus no torque is generated and the rotor speed falls below the synchronous speed. When the speed falls, the rotating magnetic field cuts the rotor windings and a torque is produced. So the induction motor will always run at a speed lower than the synchronous speed [7]. The difference between the motor speed and the synchronous speed is called the slip.

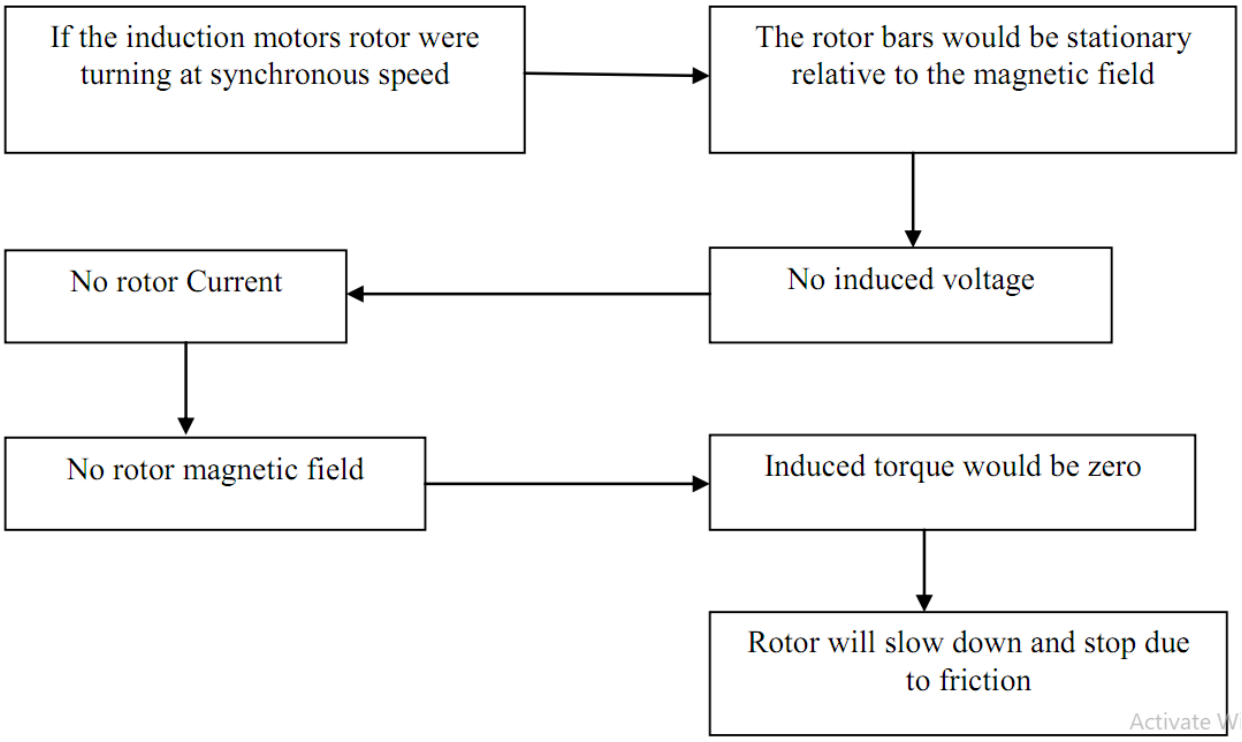

Figure 2-2: Relations of rotor speed and slip

The currents produced by each phase can be expressed by the following equations.

$$i_A = I_m \cos(\omega_e t) \quad \text{.... (2.1)}$$

$$i_B = I_m \cos(\omega_e t - 120) \quad \text{.... (2.2)}$$

$$i_C = I_m \cos(\omega_e t - 240) \quad \text{.... (2.3)}$$

The magnetic field produced in the stator winding by these three phase voltages can be expressed as follows:

$$MF_A = K i_A \cos(\theta_{ae}) \quad \text{.... (2.4)}$$

$$MF_B = K i_B \cos(\theta_{ae} - 120) \quad \text{.... (2.5)}$$

$$MF_C = K i_C \cos(\theta_{ae} - 240) \quad \text{.... (2.6)}$$

Substituting values for *"i" in* MF equations:

$$MF_A = F_{max} \cos(\theta_{ae}) \cos(\omega_e t) \quad \text{.... (2.7)}$$

$$MF_B = F_{max} \cos(\theta_{ae} - 120) \cos(\omega_e t - 120) \quad \text{.... (2.8)}$$

$$MF_C = F_{max}\cos(\theta_{ae} - 240)\cos(\omega_e t - 240) \quad \text{.... (2.9)}$$

Solving the equations with the cosine expansion

$$MF_A = \frac{F_{max}}{2}[\cos(\theta_{ae} - \omega_e t) + \cos(\theta_{ae} + \omega_e t)] \quad \text{.... (2.10)}$$

$$MF_B = \frac{F_{max}}{2}[\cos(\theta_{ae} - \omega_e t) + \cos(\theta_{ae} + \omega_e t + 120)] \quad \text{.... (2.11)}$$

$$MF_C = \frac{F_{max}}{2}[\cos(\theta_{ae} - \omega_e t) + \cos(\theta_{ae} + \omega_e t + 240)] \quad \text{.... (2.12)}$$

The above equations produce pulsating magnetic fields in the stator windings. Each pulsating magnetic field can be split in two portions. One with the $-\omega_e t$ value is revolving in the anticlockwise direction and the other is revolving in the clockwise direction.

So thus, we get the combined revolving magnetic field by the following equation:

$$MF = \left(MF_A{}^{+} + MF_B{}^{+} + MF_C{}^{+}\right) + \left(MF_A{}^{-} + MF_B{}^{-} + MF_C{}^{-}\right) = 3MF_A{}^{+} \quad \text{.... (2.13)}$$

$$MF = \frac{3F_{max}}{2}\cos(\theta_{ae} - \omega_e t) \quad \text{.... (2.14)}$$

2.2.3 Control of Induction Motor

A system employed for motion control using electric motor as a prime mover is called electric drive. The function of an electric drive system is the controlled conversion of electrical energy to a mechanical form and vice versa through a magnetic field. Electric drive is a multi-disciplinary field of study requiring proper integration of knowledge of electrical machines, actuators, power electronic converters, sensors and instrumentation, control hardware and software and communication links.

The control of AC machine is basically classified into scalar and vector control. The scalar controls are easy to implement though the dynamics are sluggish. The objective of FOC is to achieve a similar type of controller with an inner torque control loop which makes the motor respond very fast to the torque demands from the outer speed control loop. In FOC, the principle of decoupled torque and flux control are applied and it relies on the instantaneous control of stator current space vectors. Control of induction motor is complicated due to the control of decoupled torque and flux producing components of the stator phase currents. There is no direct access to the rotor quantities such as rotor fluxes and currents. To overcome these difficulties,

high performance vector control algorithms are developed which can decouple the stator phase currents by using only the measured stator current, flux and rotor speed.

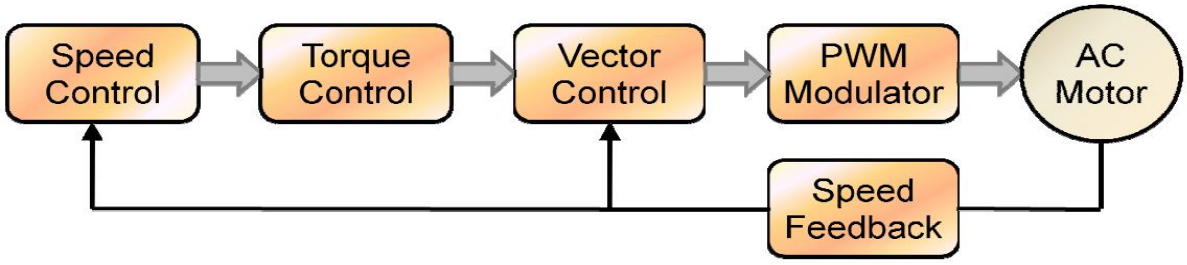

Figure 2-3: Block diagram representation of FOC

The principle of vector control of AC machine can be controlled to give dynamic performance comparable to the separately excited DC motor. There are at least three fluxes, rotor, air gap and stator and three currents, stator, rotor and magnetizing in an induction motor. For high dynamic response, interactions among current, fluxes, and speed must be taken into account in determining appropriate control strategies. Independent control of motor flux and torque can be obtained by this method and it is possible by connecting coordinate system with rotor flux vector. Figure 2-4 shows the vector diagram of induction motor in stationary α-β and rotating d-q coordinates.

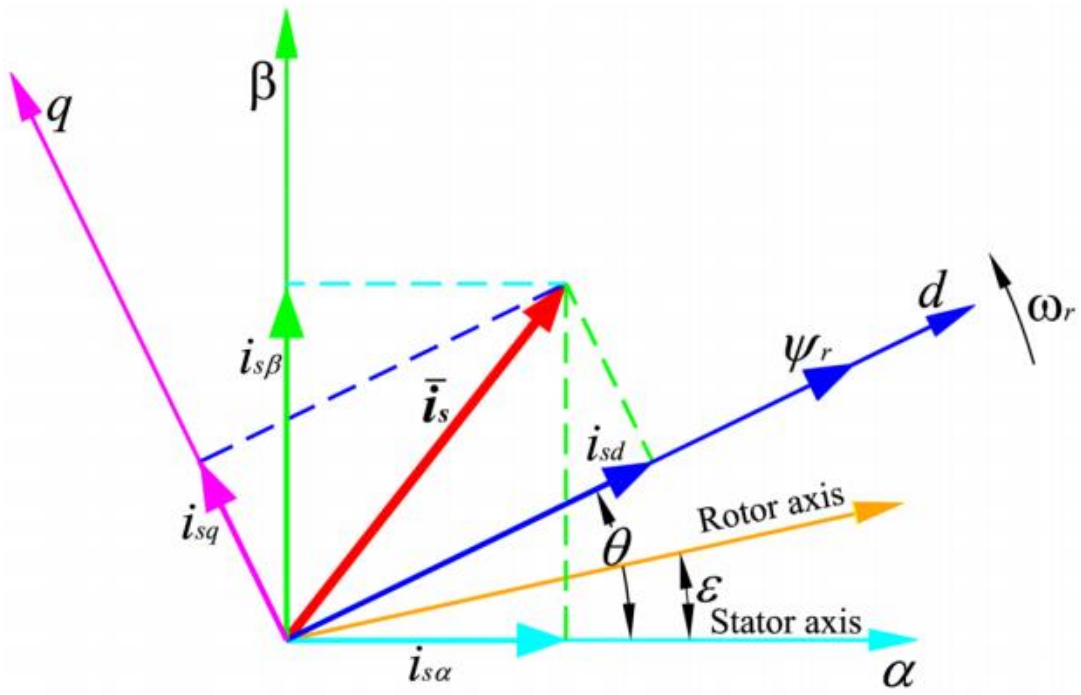

Figure 2-4: Vector diagram in stationary and rotating reference frame

2.2.4 Saturation Effect

In many industrial or other applications where electrical machines are used, strong emphasis is placed on lower energy consumption. This can be achieved by energy saving control strategies, if the electrical machine is used as a drive. Nowadays, in many applications, the controlled electric drives are required to obtain various speed and better dynamic parameters. In this case, the accurate control systems for IM are more complicated, because of mutual inductance dependence on saturation. Induction machines are usually modeled with the assumption of linear magnetics. However, in many variable torque applications, it is desirable to operate in saturation, allowing an induction machine to produce higher torque. Hence, precise modeling of saturation becomes essential for control purposes and for understanding the limitations imposed by saturation.

CHAPTER THREE

3. LABVIEW COMPACTRIO

3.1 Introduction

The National Instrument CompactRIO Control and Mechatronics Bundle is an embedded control platform. The CompactRIO Controller is a high-performance embedded controller that features industrial I/O modules, extreme ruggedness, industry-standard certifications, and integrated vision, motion, industrial communication, and human machine interface (HMI) capabilities [9].

CompactRIO is a real-time embedded industrial controller made by National Instruments for industrial control systems. NI CompactRIO incorporates a real-time processor and reconfigurable FPGA for reliable stand-alone embedded or distributed applications, and hot-swappable industrial I/O modules with built-in signal conditioning for direct connection to sensors and actuators. With NI CompactRIO, one can rapidly build embedded control or acquisition systems that rival the performance and optimization of custom-designed hardware circuitry. CompactRIO embedded systems are developed using LabVIEW, the LabVIEW Real-Time (RT) Module and the LabVIEW FPGA Module [10].

3.2 CompactRIO Embedded System

When the era of computers started, they were sometimes referred to as special purpose machines i.e. dedicated to only single task, but were far too large and expensive for most kinds of tasks performed by embedded computers of today's era. However, over the time, the concept of programmable controllers evolved from traditional electromechanical sequencers to the use of computer technology.

Because the embedded system is special purpose, design engineers can always optimize it, thereby, reducing the size as well as cost of the product, and increasing the reliability and performance.

Embedded systems are not always standalone devices. Many embedded systems consist of various other peripherals within a larger device that serves a more general purpose. For example, the Gibson Robot Guitar features an embedded system for tuning the strings, but the overall purpose of the Robot Guitar is, of course, to play music.

A CompactRIO embedded system features a real-time embedded processor, 4 or 8-slot reconfigurable chassis containing a user-programmable FPGA, and hot-swappable industrial I/O modules. This low-cost embedded architecture delivers open access to low-level hardware resources for rapid development of custom stand-alone or distributed control and acquisition systems. Figure 3-1 displays a CompactRIO embedded system [10].

Figure 3-1: cRIO 9082 Embedded Controller

3.3 Field Programmable Gate Array (FPGA)

FPGA is a reprogrammable and reconfigurable silicon chip i.e. it can be configured in the field. It of programmable interconnects which provides a high degree of flexibility while interconnecting the logic block ns programmable logic components that can be configured to emulate the functionality of basic logic gates or other complex functions.

In contrast to processors in PCs, programming an FPGA rewires the chip itself to implement your functionality rather than run a software application [9].

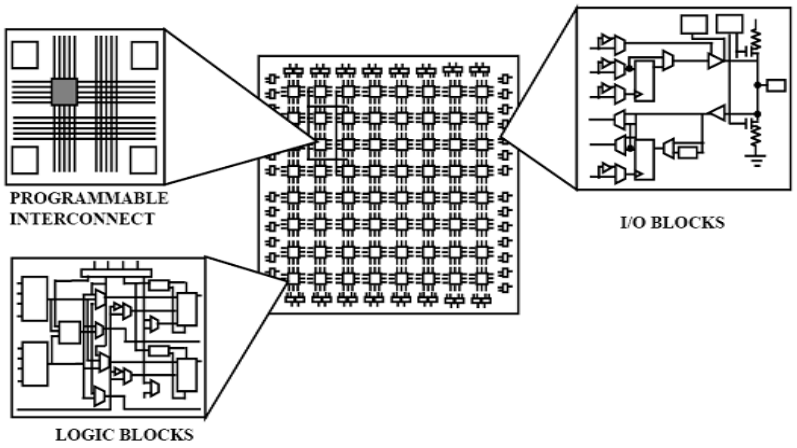

Figure 3-2: FPGA Structure

To program an FPGA, one must specify that how you want the chip to work with the help of a logic circuit diagram or a source code in a hardware description language (HDL). FPGAs can be used to implement any logical function that an ASIC could perform, but the ability to update the functionality after shipping proves to be advantageous.

FPGAs contain programmable logic components called "Logic blocks", also referred to as ―configurable logic blocks, and a hierarchy of reconfigurable interconnects that allow the blocks to be wired together resembling a one-chip programmable breadboard. The logic blocks can implement simple logic gates like AND and XOR or can also be configured to perform complex combinational functions. In most FPGAs, the logic block also incorporates memory elements, which may be simple flip-flops or more complete memory blocks.

FPGA provides its user a way to configure:

- The interconnections between the logic blocks and
- The function of each logic block

3.3.1 Logic blocks

Logic block of an FPGA can be configured according to the required functionality. It can work as a simple gate or a complex microprocessor. Logic blocks of an FPGA can be implemented by any of the following:

- ❖ Transistor Pair: In this transistor pairs run in parallel lines as shown in Figure3-3.

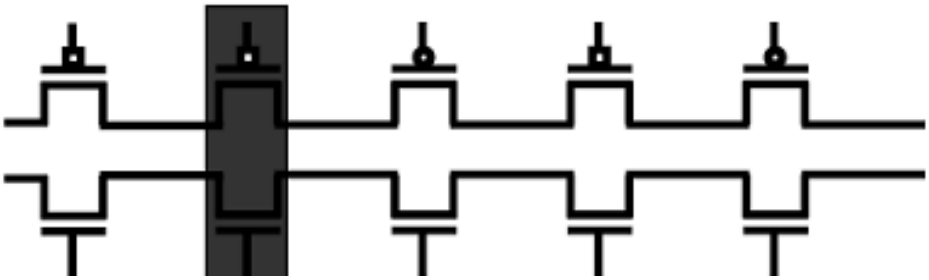

Figure 3-3: Logic block having transistor pair tiles

- ❖ Plessey logic block: Basic building block here is 2-input NAND gate which is connected to each other to implement desired function. The layout is shown in Figure 3-4.

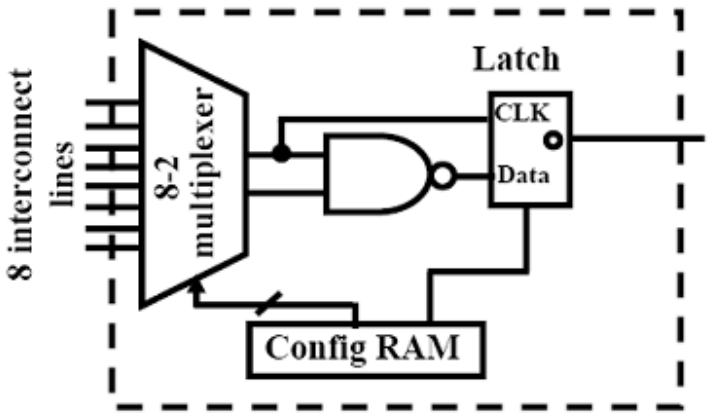

Figure 3-4: Plessey logic block

- Actel logic block: Typically, an Actel logic block consists of multiple numbers of multiplexers and logic gates. Figure 3-5 shows a typical Actel logic block.

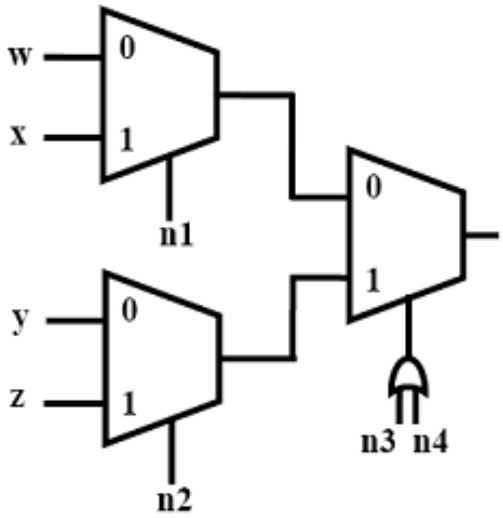

Figure 3-5: Actel Logic Block

- Xilinx logic block: Xilinx logic blocks are based on the concept of look up tables that can implement any number of different functionality. A typical Xilinx logic block is shown in Figure 3-6. The input lines are fed to the input and enable pin of lookup table. The output of the Lookup Table (LUT) gives the result of the function that it implements.

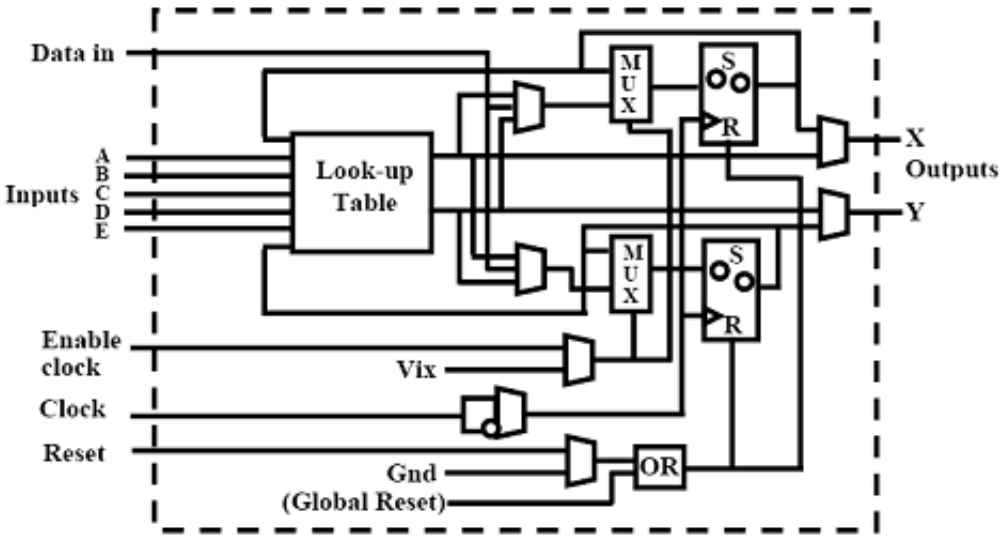

Figure 3-6: SRAM controlled programmable switches

3.4 Building a CompactRIO Embedded system

There are four steps to build a CompactRIO embedded system.

i. Choose cRIO real-time controller
ii. Choose cRIO FPGA chassis.
iii. Chooses C-Series Input-Output module(s).
iv. Choose a software (LabVIEW for Windows, LabVIEW Real-Time and LabVIEW FPGA).

3.5 Real Time Controller

The CompactRIO real time controller has an industrial processor or that balances low power consumption and powerful real-time control algorithms, signal analysis, and data logging. CompactRIO real-time controllers enable you to directly deploy your control algorithm and run the code untethered on the real-time processor without a code generation step for the user [I/O Module [10].

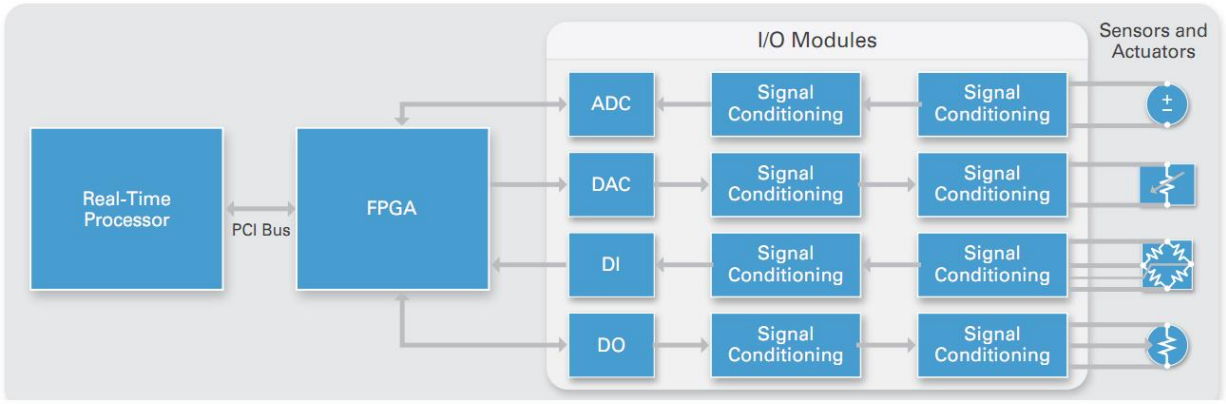

Figure 3-7: Reconfigurable Embedded System

CompactRIO combines reliable stand-alone embedded capability with extreme industrial certifications and ratings for operation in harsh industrial environments. Some of its striking features are listed below:

- –40 to 70 °C (–40 to 158 °F) operating temperature
- Up to 2,300 V_{rms} isolation (withstand)
- 50 g shock rating
- International safety, Electromagnetic Compatibility (EMC), and Environmental Certifications
- Class I, Division 2 rating for hazardous locations

3.6 I/O Module

Each CompactRIO I/O modules are self-contained measurement modules. All circuitry required for measurement is contained in the module itself. All A/D and D/A conversion occur in the module before the data reaches the chassis [9].

Each I/O module contains built-in signal conditioning and screw terminal, BNC, or D-Sub connectors. A variety of I/O types is available, including ±80 mV thermocouple inputs, ±10 V simultaneous-sampling analog inputs/outputs, 24 V industrial digital I/O with up to 1 A current drive, differential/TTL digital inputs with 5 V regulated supply output for encoders, and 250 V_{rms} universal digital inputs [10].

The NI CompactRIO Control and Mechatronics Bundle include three general-purpose C Series modules that should cover most control and mechatronics experiments. The modules are an analog input module, an analog output module, and a bidirectional digital module.

Figure 3-8: An I/O Module

3.7 Real-Time Processor

The CompactRIO embedded system features an industrial 200 MHz Pentium class processor that reliably and deterministically executes the LabVIEW Real-Time applications. One can choose from thousands of built-in LabVIEW functions to build a multithreaded embedded system for real-time control, analysis, data logging, and communication. The controller also features a 10/100 Mb/s Ethernet port for programmatic communication over the network (including e-mail) and built in Web (HTTP) and file (FTP) servers. Using the remote panel Web server, you can automatically publish the front-panel graphical user interface of your embedded application for multi-client remote monitoring or control. The real-time processor also features dual 11 to 30 VDC supply inputs, a user DIP switch, LED status indicators, a real-time clock, watchdog timers, and other high-reliability features. Figure3-9 depicts a cRIO real-time processor.

Figure 3-9: Real Time Processor

3.8 Reconfigurable Chassis

The reconfigurable chassis, as shown in Figure 3-10, is the heart of NI CompactRIO embedded systems, containing the RIO FPGA core. This user-defined RIO FPGA is a custom hardware implementation of your control logic, input/output, timing, triggering, and synchronization design. The RIO FPGA chip is connected to the I/O modules in a star topology, for direct access to each module for precise control and unlimited flexibility in timing, triggering, and synchronization. A local PCI bus connection provides a high-performance interface between the RIO FPGA and the real-time processor. The reconfigurable chassis features the same rugged metal construction that characterizes the entire CompactRIO platform

Figure 3-10: The reconfigurable chassis

3.9 Reconfigurable I/O (RIO) Technology

With NI RIO technology, one can define their own custom measurement hardware circuitry using reconfigurable FPGA chips and LabVIEW graphical development tools. Reconfigurable FPGA technology proves advantageous to automatically synthesize a highly optimized electrical

circuit implementation of our input/output, communication, or control application. (10) FPGA devices are widely used by control and acquisition system vendors because of their performance, re-configurability, small size, and low engineering development costs. One can take advantage of user-programmable FPGAs to create highly optimized reconfigurable control and acquisition systems with no knowledge of specialized hardware design languages such as VHDL. With CompactRIO, one can design their own custom control or acquisition circuitry in silicon with 25 ns timing/triggering resolution.

3.10 Applications

Because of its low cost, reliability, and suitability for high-volume embedded measurement and control applications, CompactRIO can be adapted to solve the needs of a wide variety of industries and applications. Examples include:

- Batch control
- Discrete control
- Motion control
- In-vehicle data acquisition
- Machine condition monitoring
- Rapid control prototyping (RCP)
- Industrial control and acquisition
- Distributed data acquisition and control
- Mobile/portable noise, vibration, and harshness analysis

CompactRIO is designed for advanced developers who will use LabVIEW graphical development tools to adapt the reconfigurable hardware for a wide variety of industries and applications. Customers such as MTS, Roush, Göpel, Process Automation and Virginia Tech have already successfully developed CompactRIO embedded systems for heavy machine control, in-vehicle data acquisition, acoustics and vibration analysis and electric motor drive characterization.

3.10.1 NI 9401: An 8-channel TTL Digital I/O Module

The NI 9401 shown in Figure3-11 has a DSUB connector that provides connections for the eight digital channels. Each channel has a DIO pin to which a digital input or output device can be

connected. The DIO channels of the NI 9401 are grouped in two ports, one containing channels 0, 1, 2, and 3, and other containing channels 4, 5, 6, and 7. Each port in software can be configured for input or output. All the four channels in a port must be configured in same direction.

Figure 3-11: NI 9401 digital I/O Module

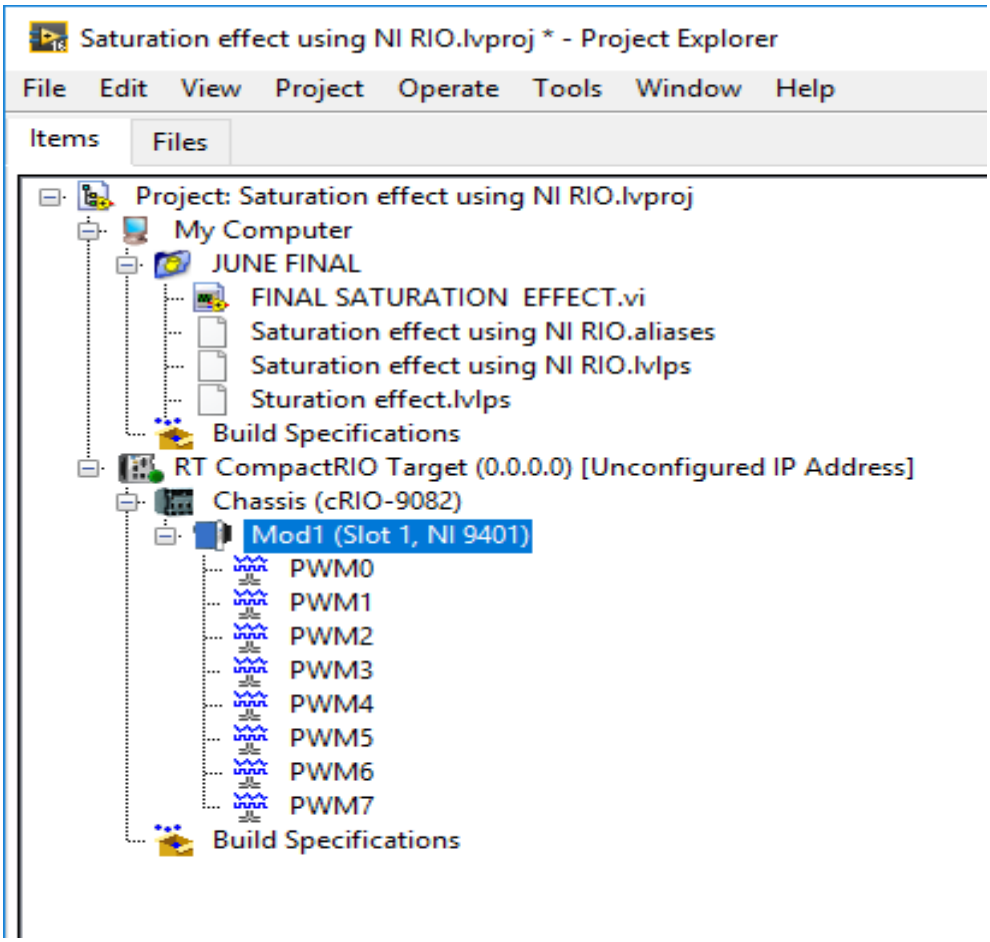

Figure 3-12: Project Explorer Windows for PWM

3.11 Required Software

3.12 LabVIEW

Programmers develop software applications every day in order to increase efficiency and productivity in various situations. LabVIEW, as a programming language, is a powerful tool that can be used to help achieve these goals. LabVIEW (Laboratory Virtual Instrument Engineering Workbench) is a graphically-based programming language developed by National Instruments. Its graphical nature makes it ideal for test and measurement (T&M), automation, instrument control, data acquisition, and data analysis applications. This results in significant productivity improvements over conventional programming languages. National Instruments focuses on products for T&M, giving them a good insight into developing LabVIEW [11].

3.12.1 VIRTUAL INSTRUMENTS

Simply put, a Virtual Instrument (VI) is a LabVIEW programming element. A VI consists of a front panel, block diagram, and an icon that represents the program. The front panel is used to display controls and indicators for the user, and the block diagram contains the code for the VI. The icon, which is a visual representation of the VI, has connectors for program inputs and outputs. Programming languages, such as C and BASIC, use functions and subroutines as programming elements [11]. LabVIEW uses the VI. The front panel of a VI handles the function inputs and outputs, and the code diagram performs the work of the VI.

Multiple VIs can be used to create large scale applications; in fact, large scale applications may have several hundred VIs. A VI may be used as the user interface or as a subroutine in an application. User interface elements such as graphs are easily accessed, as drag-and-drop units in LabVIEW.

3.12.2 Frontal Panel

Figure above illustrates the front panel of a LabVIEW VI. It contains a knob for selecting the number of measurements per average, a control for selecting the measurement type, a digital indicator to display the output value, and a stop button.

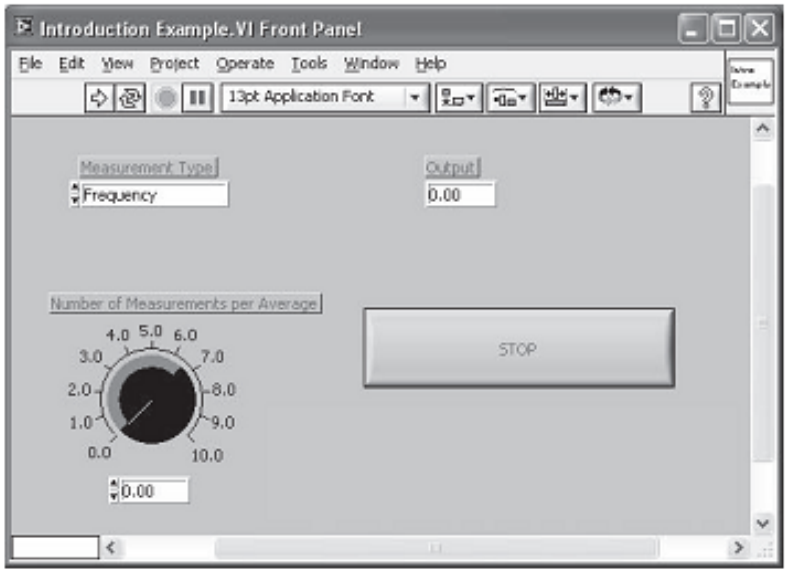

Figure 3-13: Frontal Panel

3.12.3 BLOCK DIAGRAM

Figure 3-14 depicts the block diagram, or source code, that accompanies the front panel in Figure 3.13. The outer rectangular structure represents a While loop, and the inner one is a case structure. The icon in the center is a VI, or subroutine, that takes the number of measurements per average as input and returns the frequency value as the output. The orange line, or wire, represents the data being passed from the control into the VI. The selection for the measurement type is connected, or wired to the case statement to determine which case is executed. [11]When the stop button is pressed, the While loop stops execution. This example demonstrates the graphical nature of LabVIEW and gives you the first look at the front panel, block diagram, and icon that make up a Virtual Instrument.

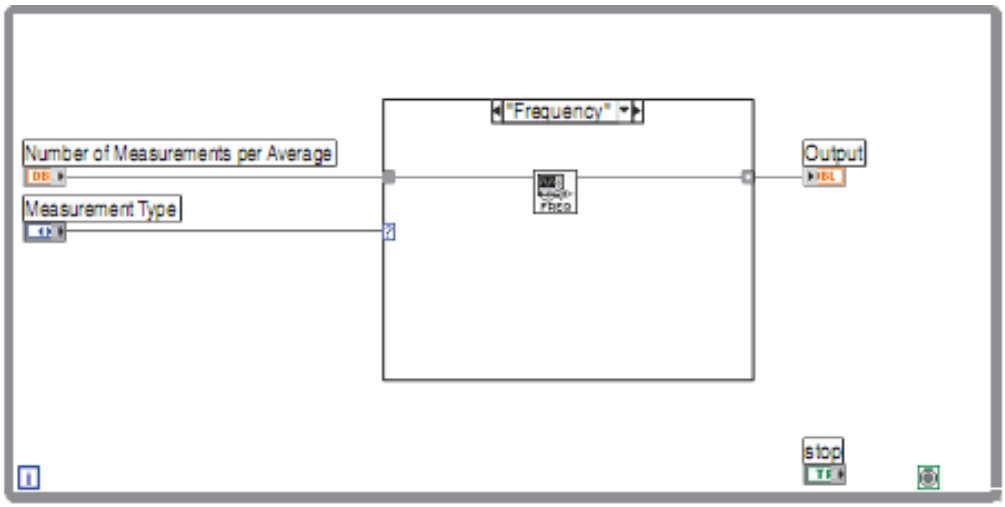

Figure 3-14: Block Diagram

LabVIEW is not an interpreted language; it is compiled behind the scenes by LabVIEW's execution engine. Similar to Java, the VIs are compiled into an executable code that LabVIEW's execution engine processes during runtime. Every time a change is made to a VI, LabVIEW constructs a wire table for the VI. This wire table identifies element in the block diagram; that have inputs needed for that element to run. Elements can be primitive operators such as addition, or more complex such as a subVI. If LabVIEW successfully constructs all the wire tables, you are presented a solid arrow indicating that the VIs can be executed. If the wire table cannot be created, then a broken arrow is presented for the VIs with a problem. And also for each VI loaded in memory that requires that VI for execution. LabVIEW runs in several subsystems,

which will be described throughout this book. All that we need to understand now is that the main execution subsystem compiles diagrams while you write them. This allows programmers to write code and test it without needing to wait for a compiling process, and programmers do not need to worry about execution speed because the language is not interpreted [11].

The wire diagrams that are constructed do not define an order in which elements are executed. This is an important concept for advanced programmers to understand. LabVIEW is a dataflow-based language, which means that elements will be executed in a somewhat arbitrary order. LabVIEW does not guarantee which order a series of elements is executed in if they are not dependent on each other. A process called arbitrary interleaving is used to determine the order elements are executed in. You may force an order of execution by requiring that elements require output from another element before execution. This is a fairly common practice, and most programmers do not recognize that they are forcing the order of execution. When programming, it will become obvious that some operations must take place before others can. It is the programmer's responsibility to provide a mechanism to force the order of execution in the code design [11].

3.12.4 Icon and Connector Pane

After a front panel and block diagram is built, icon and connector pane shall be built so one can use the VI as a sub VI. The icon and connector pane correspond to the function prototype in text-based programming languages. Every VI displays an icon, such as the one shown in Figure 3-15, in the upper right corner of the front panel and block diagram windows [12].

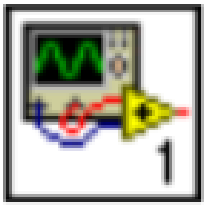

Figure 3-15: Icon

An icon is a graphical representation of a VI. It can contain text, images or a combination of both. If one uses a VI as SubVI, the icon identifies the SubVI on the block diagram of the VI. One can double click the icon to customize or edit it [12].

One also needs to build a connector pane, as shown in Figure 3-16, to use the VI as a SubVI.

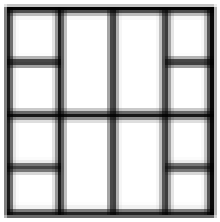

Figure 3-16: Connector Pane

The connector pane is a set of terminals that correspond to the controls and indicators of that VI, similar to the parameter list of a function call in text-based programming languages. The connector pane defines the inputs and outputs one can wire to the VI so one can use it as a SubVI. A connector pane receives data at its input terminals and passes the data to the block diagram code through the front panel controls and receives the results at its output terminals from the front panel indicators [12].

3.13 LabVIEW Real-time

The National Instruments LabVIEW Real-Time Module is an add-on component for the LabVIEW Development System. When installed, this software compiles NI LabVIEW graphical code and optimizes it for the selected real-time target. Using the LabVIEW Real-Time Module, applications can be develop and deploy to all NI real-time hardware targets like CompactRIO.

3.14 LabVIEW FPGA

National Instruments LabVIEW and the LabVIEW FPGA Module deliver graphical development for field-programmable gate array (FPGA) chips on NI Reconfigurable I/O (RIO) hardware targets. With an NI LabVIEW FPGA Module, FPGA VIs can be develop on a host computer running Windows, and LabVIEW compiles and implements the code in hardware. Embedded FPGA VIs can be developed that combine direct access to I/O with user-defined LabVIEW logic to define custom hardware for different applications.

3.15 Reconfigurable I/O FPGA

The reconfigurable I/O FPGA chassis is the center of the embedded system architecture. It is directly connected to the I/O for high-performance access to the I/O circuitry of each module and

timing, triggering, and synchronization. Because each module is connected directly to the FPGA rather than through a bus, you experience almost no control latency for system response compared to other controller architectures. By default, this FPGA automatically communicates with I/O modules and provides deterministic I/O to the real-time processor. Out of the box, the FPGA enables programs on the real-time controller to access I/O with less than 500 ns of jitter between loops. You can also directly program this FPGA to further customize the system. Because of the FPGA speed, this chassis is frequently used to create controller systems that incorporate high-speed buffered I/O, fast control loops, or custom signal filtering. For instance, using the FPGA, a single chassis can execute more than 20 analog proportional integral derivative (PID) control loops simultaneously at a rate of 100 kHz. Additionally, because the FPGA runs all code in hardware, it provides the highest reliability and determinism, which is ideal for hardware-based interlocks, custom timing and triggering, or the elimination of custom circuitry normally required with nonstandard sensors and buses.

CHAPTER FOUR

4. METHODOLOGY AND MODELING

4.1 Introduction

As described in chapter one, the main objective of this thesis is to investigate the possible parameter variation in induction machine under a saturation condition. Particularly, the paper gives high emphasis for the case in steady state torque characteristics of the induction motor in rotor field oriented control.

4.2 A Saturated Induction Motor Model in Rotor Field Oriented Control

To work on the objectives listed under chapter one, section 1.3 the following mathematical modeling is developed.

In rotor field oriented control reference frame ($\varphi_{rq}= 0$) the dynamic induction motor has the subsequent forms.

$$v_{sd} = R_s i_{sd} + \frac{d\varphi_{sd}}{dt} - \omega_{mr}\varphi_{sq} \quad \ldots\ldots (4.1)$$

$$v_{sq} = R_s i_{sq} + \frac{d\varphi_{sq}}{dt} + \omega_{mr}\varphi_{sd} \quad \ldots\ldots (4.2)$$

$$v_{rd} = R_r i_{rd} + \frac{d\varphi_{rd}}{dt} = 0 \quad \ldots\ldots (4.3)$$

$$v_{rq} = R_r i_{rq} + (\omega_{mr} - \omega_r)\varphi_{rd} \quad \ldots\ldots (4.4)$$

$$\tau_e = PL_m\left(i_{sq} i_{rd} - i_{sd} i_{rq}\right) = P\frac{L_m}{L_r}\varphi_{rd} i_{sq} \quad \ldots\ldots (4.5)$$

$$J\frac{d_{\omega r}}{dt} = t_e - t_l - B\omega_r \quad \ldots\ldots (4.6)$$

$$\varphi_{sd} = L_s i_{sd} + L_m i_{rd} \quad \ldots\ldots (4.7)$$

$$\varphi_{sq} = L_s i_{sq} + L_m i_{rq} \quad \ldots\ldots (4.8)$$

$$\varphi_{rd} = L_m i_{sd} + L_r i_{rd} \quad \ldots\ldots (4.9)$$

$$\varphi_{qr} = L_m i_{sq} + L_r i_{rq} \quad \ldots\ldots (4.10)$$

Where: v_{sd} *and* v_{sq} *are stator voltages.*

$v_{rd\ and}\ v_{rq}$ are rotor voltages.

i_{sd} and i_{sq} are d-axis and q-axis rotor currents respectively.

φ_{sd} and φ_{sq} are stator flux linkages.

φ_{rd} and φ_{rq} are rotor flux linkages.

R_s and R_r are stator and rotor resistances respectively.

L_r and L_s are rotor and stator Self-Inductance respectively.

J is drive inertia.

B = coefficient of viscous friction.

t_e and t_l are electrical and load torque respectively.

P is the number of pole pairs.

If *d-axis* is aligned with the rotor field, the *q-component* of the rotor field in the rotor reference frame is zero *(i.e. φ_{rq}= 0)*. From *(4.10);*

$$\varphi_{qr} = L_m i_{sq} + L_r i_{rq} = 0$$

$$i_{rq} = -\frac{L_m}{L_r} i_{sq} \quad \ldots\ldots (4.11)$$

When the field is properly oriented i_{rd} is zero. Therefore at a steady state, equation *(4.9)* would be rewritten as *(4.12);*

$$\varphi_{rd} = L_m I_{sd} \quad \ldots\ldots (4.12)$$

Putting *(4.12)* into the torque equation *(4.5)* for steady state operation of IM:

$$\tau_e = P\frac{{L_m}^2}{L_r} I_{sd} I_{sq} \quad \ldots\ldots (4.13)$$

Now we can rewrite (4.13) as a function of *(I_{sq}/I_s) as shown in (4.14).*

$$\tau_e = P\frac{{L_m}^2}{L_r} I_{sd} I_s \left(\frac{I_{sq}}{I_s}\right) \quad \ldots\ldots (4.14)$$

Where: $$I_s = \sqrt{{I_{sq}}^2 + {I_{sd}}^2} \quad \ldots\ldots (4.15)$$

Further *(4.14)* can be written as *(4.16).*

$$\tau_e = P\frac{{L_m}^2}{L_r}{I_s}^2\left(\frac{I_{sq}}{I_s}\right)\sqrt{1-\left(\frac{I_{sq}}{I_s}\right)^2} \quad \ldots\ldots (4.16)$$

4.3 Control scheme

The control scheme used in this thesis work is shown in Figure 4-1 and Figure 4-2. The main idea is to control the torque and speed of the induction motor using cRIO. And also investigating the impact of magnetic saturation from torque response of the motor is the core point.

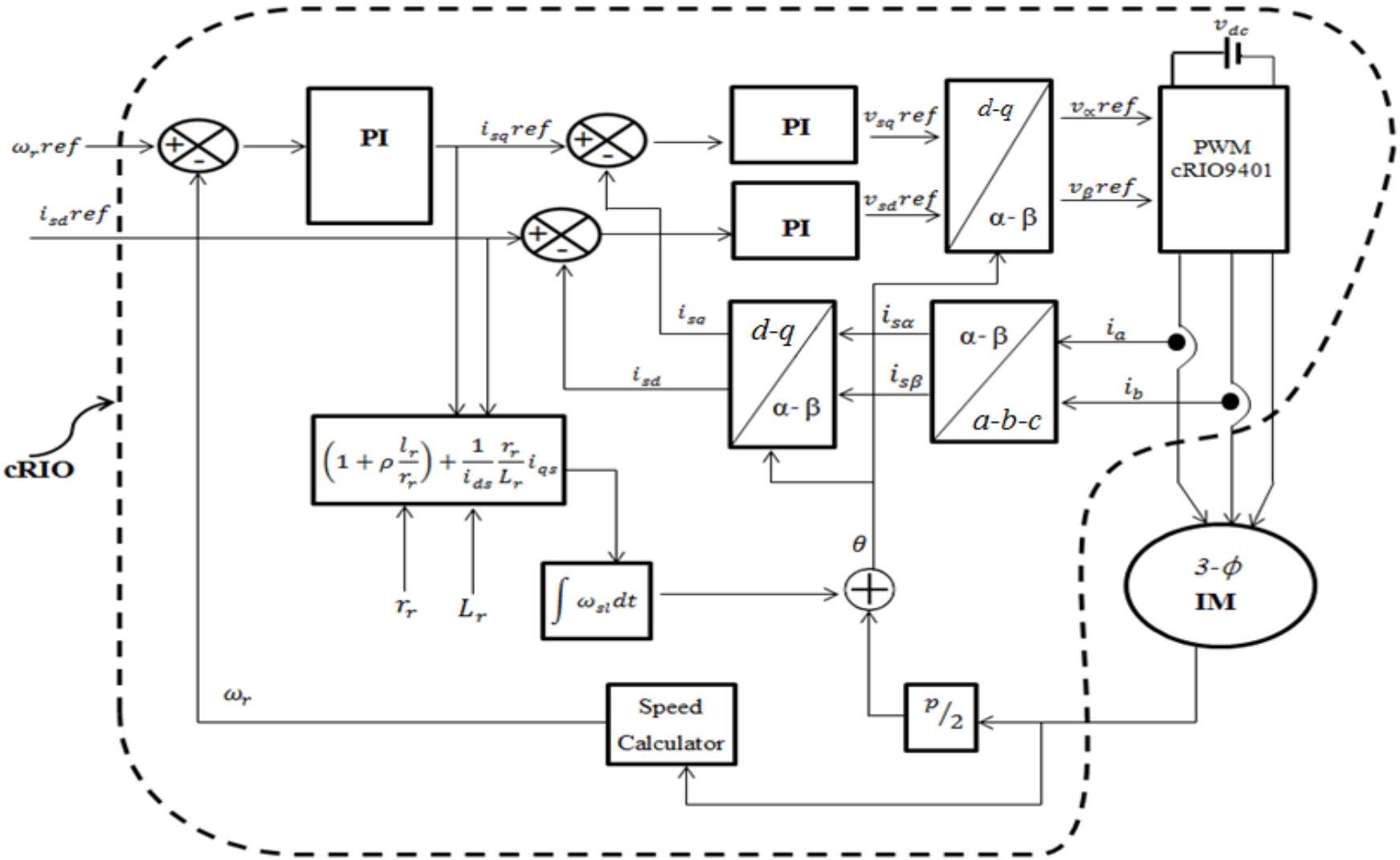

Figure 4-1: Block diagram representation of the Control Scheme

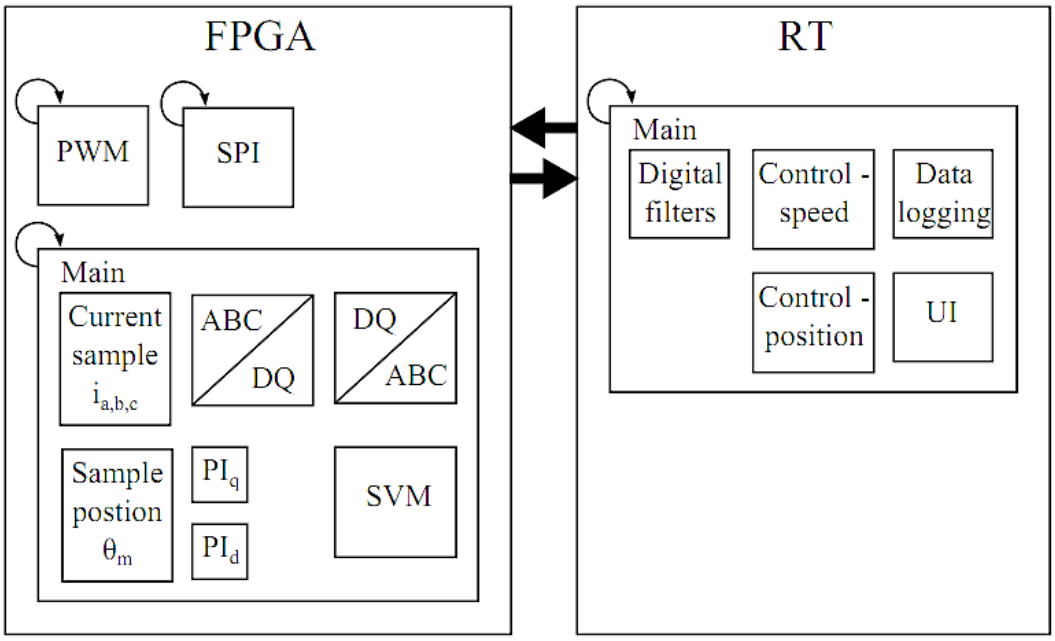

Figure 4-2: Block diagram of Program allocation on FPGA and RT respectively

4.4 Programming LabVIEW

4.4.1 Selecting the programing mode for the application

CompactRIO is programmed using NI LabVIEW, a comprehensive development environment that provides engineers and scientists unprecedented hardware integration and wide ranging compatibility. Because CompactRIO is a distributed real-time system, it also uses the LabVIEW Real-time Module and optionally, the LabVIEW FPGA module.

In order to fully utilize the capacity of CompactRIO, the following software is a must to use.

- LabVIEW Development System
- LabVIEW Real-Time Module
- LabVIEW FPGA Module
- LabVIEW Control Design and simulation Module
- LabVIEW MathScript RT Module
- NI_RIO driver

A LabVIEW project is organized in a hierarchical fashion. There are two processing units. One is called ‘My Computer’ and is ‘window based-PC’ where development takes place, simulation

code can be run and project algorithms can be tested before implementation. This processing unit is often called the "host". The other processing unit is the CompactRIO system. It is important to note that simulation code can run on either processing units but development can take place only on the host.

4.5 Configuring LabVIEW

In this thesis work a 3-KW induction motor is used for analysis. All the data for the motor to which all results of this thesis refer are given in Table 4-1.

Table 4-1: Parameters used for 3-KW Induction Motor

Input Parameters	Values
$L_{\sigma s}$	0.0105
$L_{\sigma r}$	0.0105
L_m	0.223
R_s	1.976
R_r	2.91
J	0.031
B	0.0007
T_n	15

4.6 Simulation Model

LabVIEW programming is essentially the same for any target, be it on PC or any type national instrument hardware, the same environment and programming principles apply even to Field Programming Gate Array (FPGA).

4.6.1 Field Oriented Control algorithm VI

The principle of rotor field oriented control system of an induction motor is that d-q coordinate reference frame is locked to the rotor flux vector. This results in decoupling of variables so that

flux and torque can be separately controlled by stator direct axis current, i_{sd} and quadrature axis current, i_{sq} respectively. Like in separately excited DC machine.

Using the mathematical model listed under section 4.2, the following RFOC shown in Figure 4-2 is formulated.

Figure 4-3 shows the project explorer window created to program PWM. All eight digital output pins seen are under the folder named "Mode1 (slot1-NI 9401)"

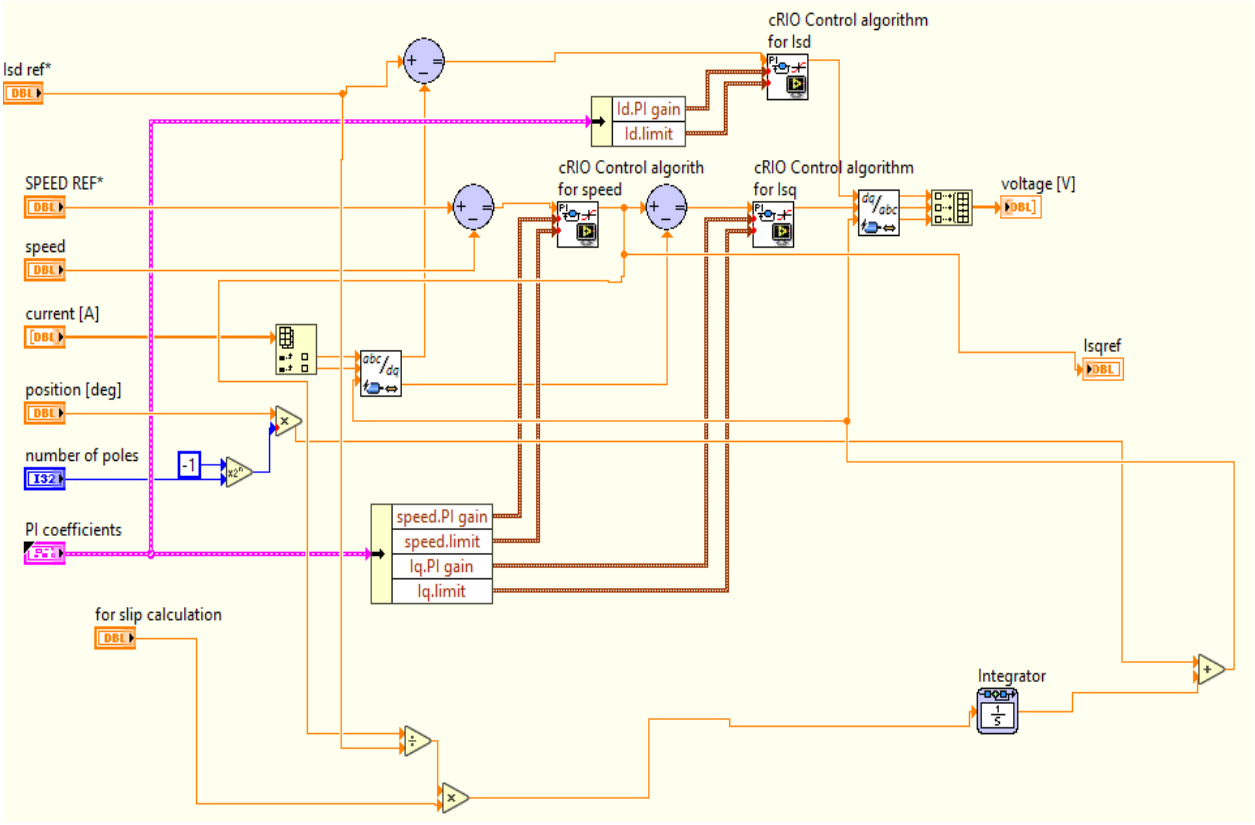

Figure 4-3: Rotor Field Oriented Control VI

In this thesis a labVIEW CompactRIO based algorithm is designed, and simulated on a host VI. A host VI is a VI that is targeted to a processor, in this case a PC.

The VI in the 'JUNE FINAL' folder of the Figure 4-4 runs on the host PC because it is located under my computer in the project hierarchy. This VI contains code that is not connected to I/O and is meant to be used to validate cRIO algorithms in simulation before being connected to physical inputs and outputs.

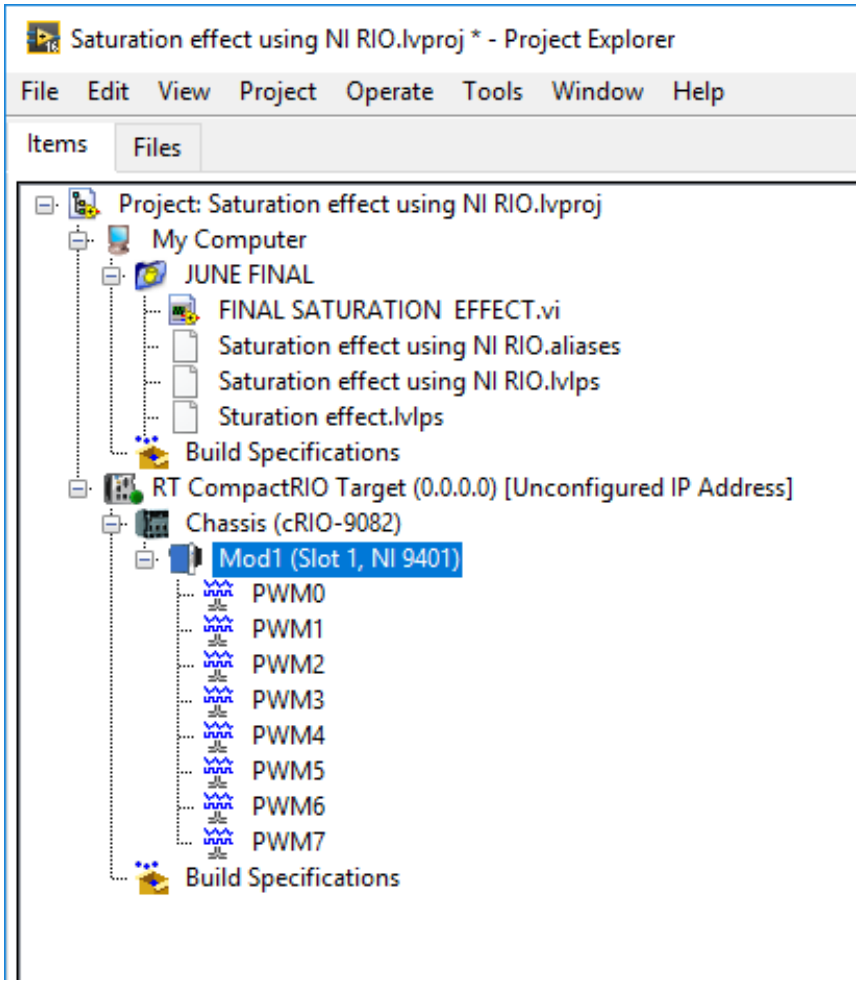

Figure 4-4: Project Explorer Windows for PWM

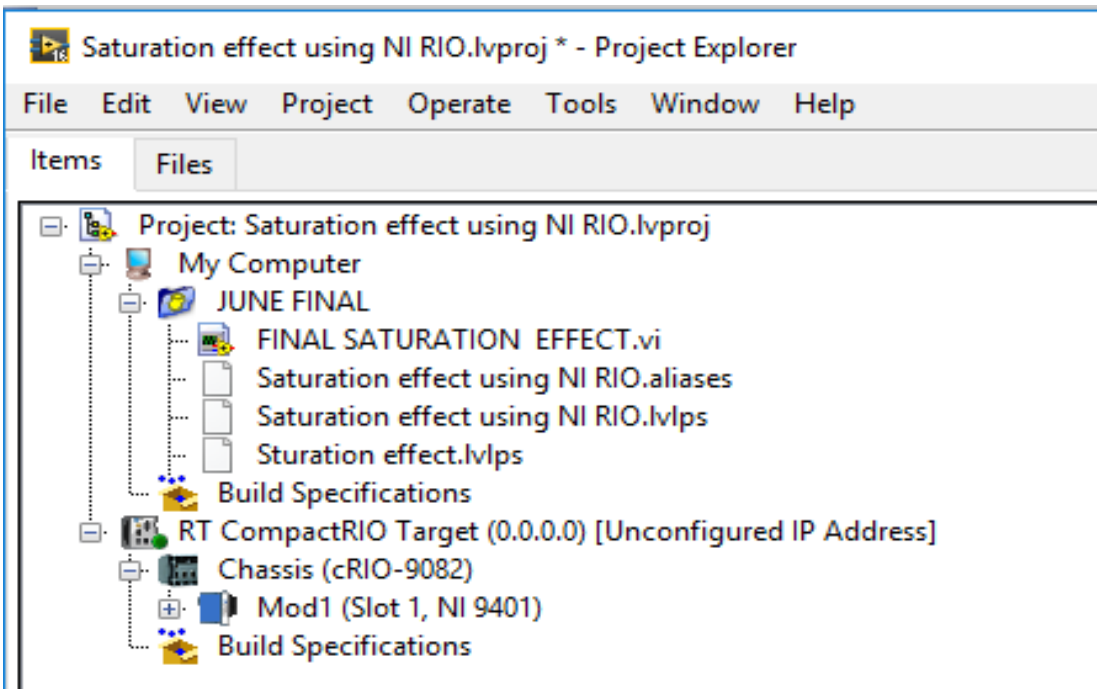

Figure 4-5: Project Explorer Window

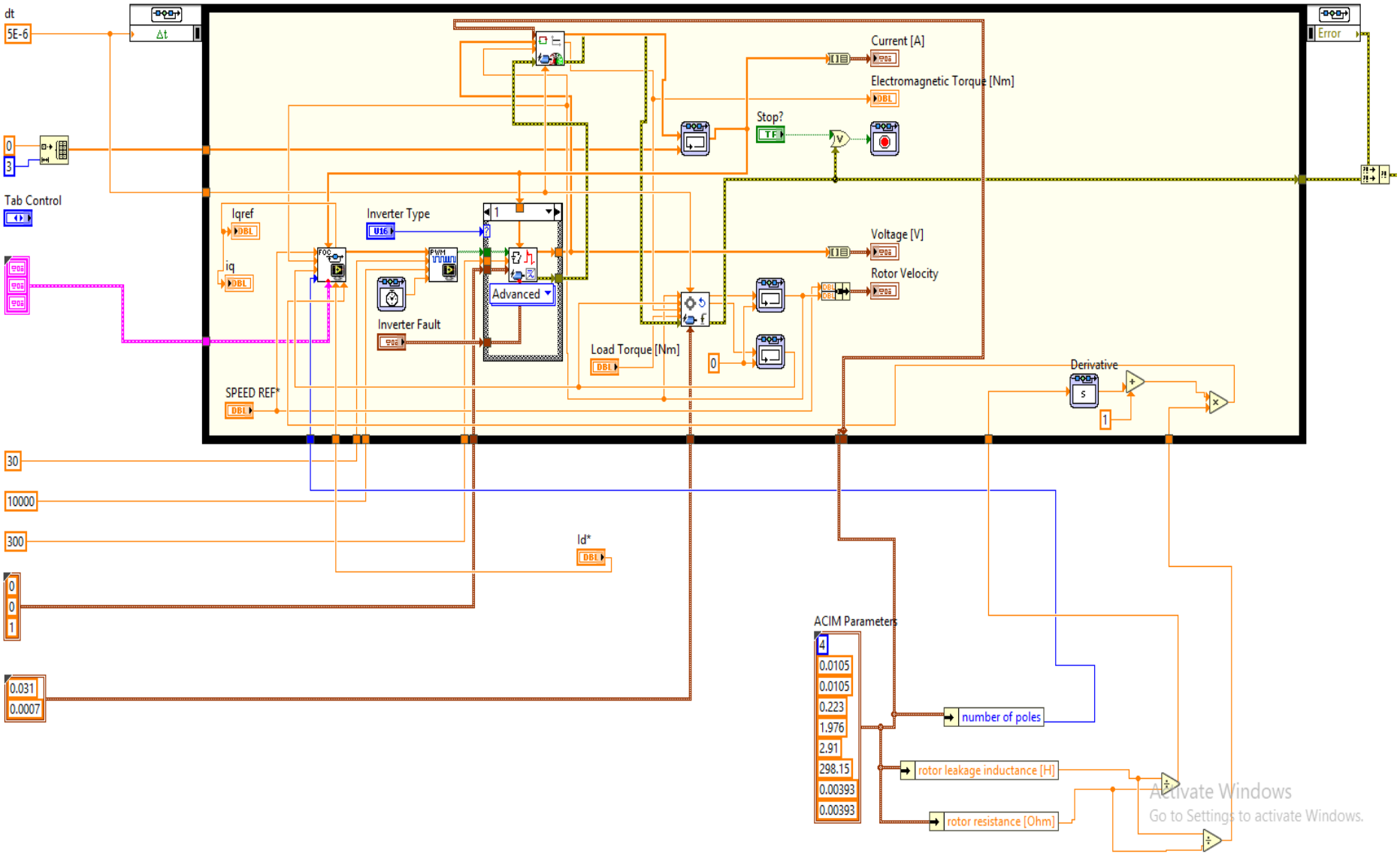

Figure 4-6: Simulation Model of the Overall System

4.7 Analysis of Parameter Variation

Recall *(4.14)*, to ensure good dynamic performance servo drives usually operate with constant I_{sd}. Now let us consider two scenarios using the mentioned equation.

Case I - Torque Characteristics without consideration of Saturation Effect

Case II - Torque Characteristics with consideration of Saturation Effect

Case I - Torque Characteristics without consideration of Saturation Effect

The induction motor can also operate with variable d-axis stator current I_{sd}. With the constraint of the operation range of input current mentioned on (4.15), the torque for constant I_s using (4.16) is expressed in Table 4-2.

Table 4-2: Torque without Considering Saturation for Constant Is = 12.

I_{sd} (A)	11.76	11.00	9.6	7.2
I_{sq} (A)	2.4	4.8	7.2	9.6
I_s (A)	12	12	12	12
I_{sq}/ I_s (A)	0.2	0.4	0.6	0.8
τ_e (Nm)	18.06	33.79	44.23	73.73

Table 4-3: Torque without considering saturation for constant Is = 9.

τ_L (Nm)	5	7.5	10	15
I_{sd} (A)	8.80	8.41	7.2	5.4
I_{sq} (A)	1.9	3.6	5.4	7.2
I_s (A)	9	9	9	9
I_{sq}/ I_s (A)	0.2	0.4	0.6	0.8
τ_e (Nm)	10.7	19.38	24.88	24.88

Table 4-4: Torque without considering saturation for constant Is = 6.

τ_L (Nm)	5	7.5	10	15
I_{sd} (A)	5.88	5.50	4.80	3.6
I_{sq} (A)	1.2	2.4	3.6	4.8
I_s (A)	6	6	6	6
I_{sq}/ I_s (A)	0.2	0.4	0.6	0.8
τ_e (Nm)	4.52	8.448	11.06	11.06

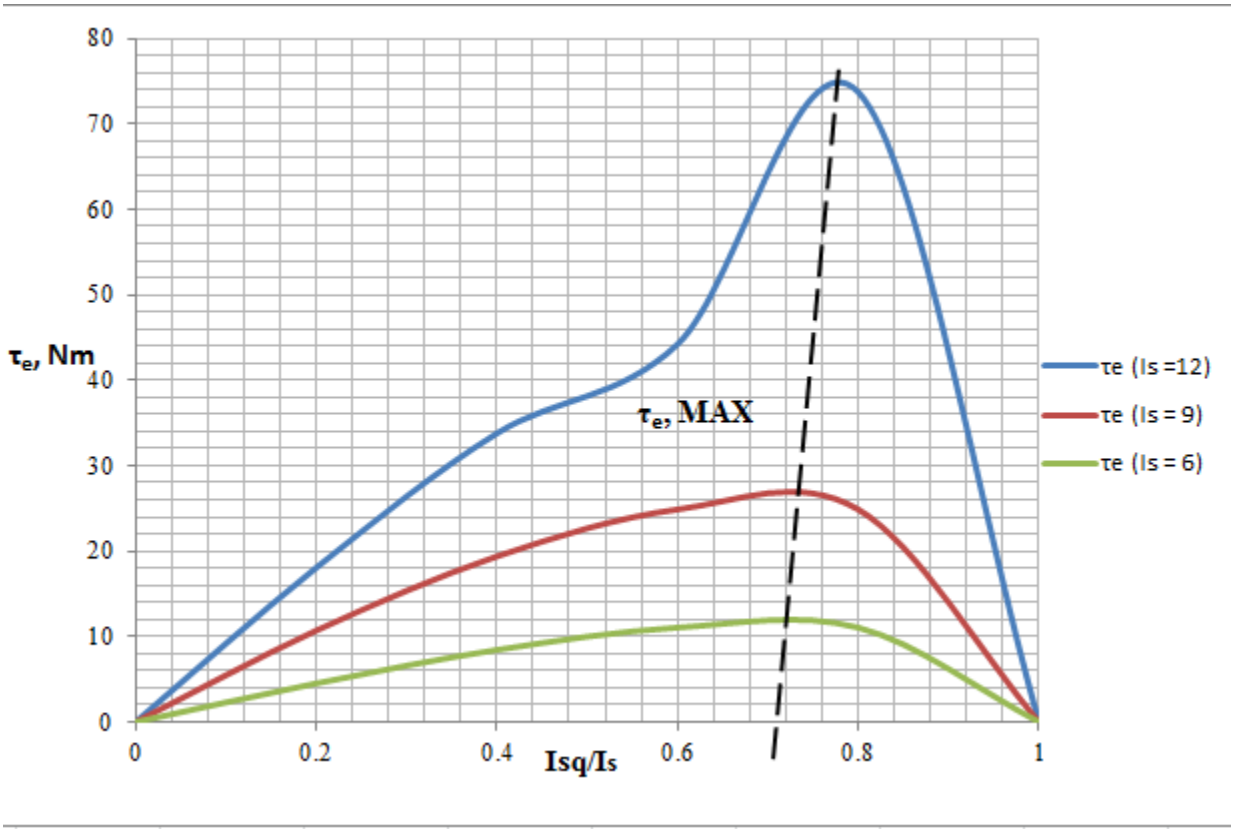

Figure 4-7: Torque characteristics for different Is values (without magnetic saturation)

It can be seen from the above results that the sets of torque without considering magnetic saturation of the same induction motor is unusually high for constant values of I_s. And the maximum torque occurs at Isq/Is = 0.707.

Case II - Torque Characteristics with consideration of Saturation Effect

In the contemporary high performance AC motor drives, the most widely used scheme is an indirect field oriented control (IFOC) with current controllers in d-q reference frame tied to the rotor flux vector. In this type of drives, a mismatch between machine parameters used in the controller and the actual machine parameters, which occurs due to changes in temperature or saturation, results in the following:

- The flux level is not properly maintained, and the rotor flux amplitude is not equal to the expected value.
- The resulting steady state torque is not equal to the commanded values.
- The torque response is not instantaneous.
- The correlation between I_{qs} and torque, as well as the correlation between ids and flux is not linear anymore.

In the indirect field oriented control schemes with current controllers in d-q reference frame tied the rotor flux vector, the parameter of greatest interest is the rotor time constant τr, $\tau r = L_r / R_r = (L_m + L_{\sigma r})/R_r$, where L_r, R_r and $L_{\sigma r}$ are rotor inductance, rotor resistance and rotor leakage inductance, respectively. L_m denotes mutual inductance between stator and rotor.

Experience and intuition suggests that large magnetizing current will be required to create flux levels significantly larger than the nominal rated flux. It is obvious that from the understanding of magnetic behavior, if the saturation effect of the iron core is considered, the true value of torque we get would decrease due to the behavior of the magnetizing curve as shown in Figure 4-9.

A wrongly set parameter in the model yields the difference between the commanded and obtained values of the currents even if the current controllers are ideal. Both, the torque-producing (I_{qs}) and the flux-producing (I_{ds}) currents are affected.

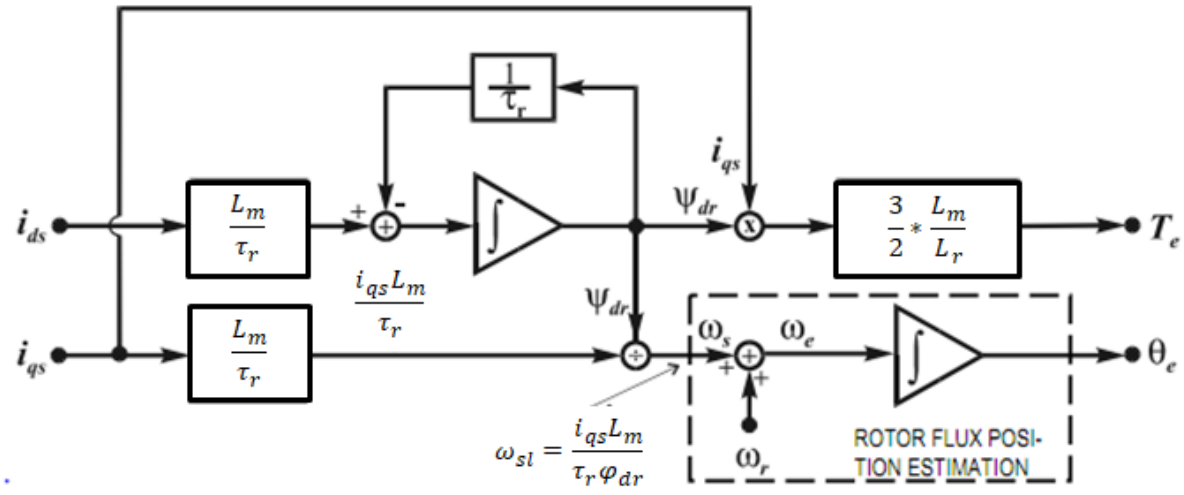

Figure 4-8: Induction machine dynamics model in the d-q frame.

This model assumes a correct orientation of the d-q reference frame. To get the correct orientation, the controller that controls the drive must have the correct information about the current rotor flux angle θe because this angle is used in the transformation of the measured currents. The result of the transformation is the value of the stator current components I_{ds} and I_{qs}, used as the input in the model. For this reason (to get current θ_e) the controller contains a software copy of the model. The software model runs in parallel with the machine following the dynamics of all variables [13].

The torque producing current reference I_{qs}* (q-component) controls the torque while the d-component reference (or flux-producing current) I_{ds}* controls the amplitude of the rotor flux. By means of I_{qs}* the controller calculates the angular slip frequency as:

$$\omega_{sl}^{*} = \frac{i_{qs}^{*} L_m^{*}}{\tau_r^{*} \varphi_{dr}^{*}}$$

Stars in superscript for motor parameters identify the values used in mathematical model (in the controller) and are not necessarily equal to actual parameters in the machine.

The integral of ω_{sl}* presents the "slip contribution" to the rotor flux position, and is used to estimate (by adding to measured rotor position) current rotor flux position (θ_e*), essential for coordinate transformations. Since the controller calculates ω_s* using wrong parameter values, the estimated angle θ_e* is not equal to the real rotor flux position (angle θ_e) in the machine.

Unfortunately, the controller uses this, wrongly estimated, angle θ_e^* to rotate (transform from the stationary to rotated frame) the measured stator currents $i_{\alpha s}$ and $i_{\beta s}$. The transformation (rotation) is necessary to get the d and q component of the stator current because the current controllers work in the synchronous rotating d-q frame [13].

Critical parameters for the correct θ_e estimation are the rotor resistance R_r and the mutual inductance Lm. The leakage inductance $L_{\sigma r}$ which is also included in τ_r expression, is usually much smaller than Lm, appears always in sum with much larger L_m, and does not vary too much. Rotor resistance varies a lot with the machine heating, and the mutual inductance L_m depends on current ids (or, more precisely, on magnetizing current, I_m, which is equal to ids in steady state). This paper discusses only the influence of mutual inductance variations. The rotor resistance R_r is considered as a known constant in the simulations.

Even if the reference value I_{ds}^* never exceeds the nominal, the actual I_{ds} can be several times larger than expected if the parameter R_r^* used in the controller differs from R_r in the machine r these reasons, the variation of L_m cannot be neglected [13].

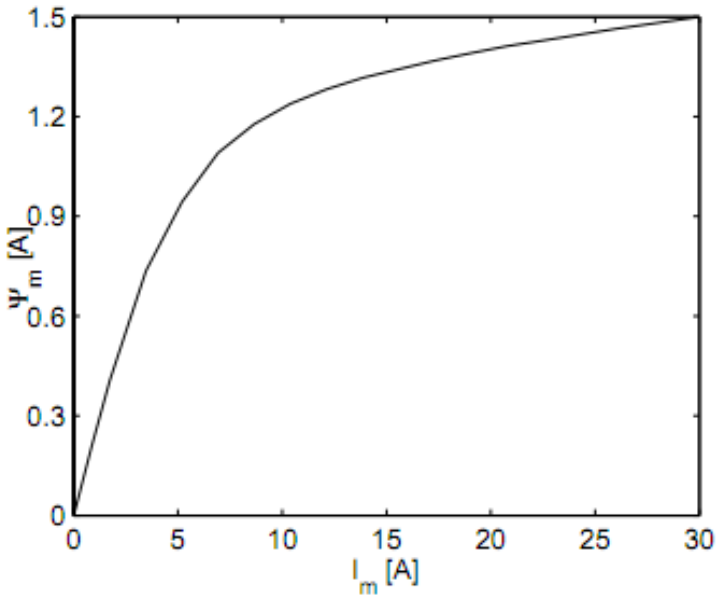

Figure 4-9: Magnetizing Curve

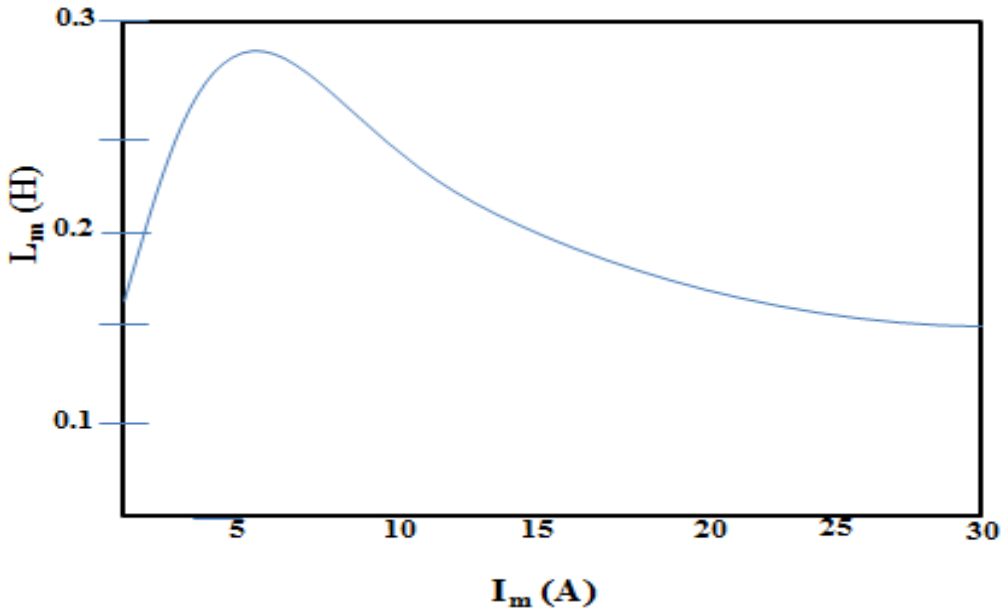

Figure 4-10: Mutual Inductance of Investigated Motor (Approximate Curve)

The maximum value of the torque we obtain for a given value of I_s can be obtained from the first derivative of (4.16) with respect to the ratio I_{sq}/I_s.

$$\frac{d\tau_e}{dt} = 0 \rightarrow r_{max} \qquad \text{.. ..(4.17)}$$

If we neglect magnetic saturation (4.17) have the following forms

$$2{r_{max}}^2\text{-}1=0 \qquad \text{.....(4.18)}$$

The roots $r_{max}1,2 = \pm {\sqrt{2}}/{2}$ which shows that the maximal torque of the controlled induction motor drive produced when $I_{sq}/I_s = 0.707$, that is $I_{sd}=I_{sq}$.

But if we take saturation effect into consideration we have

$$a_4{r_{max}}^4 + a_2{r_{max}}^2 + a_0 = 0 \qquad \text{..... (4.19)}$$

$$a_4 = -\left(2L_m - \frac{{L_m}^2}{L_r}\right)\left(2L_mL_r - {L_m}^2\right)(L_m - L)$$

$$- 2\frac{{L_m}^2}{L_r}\left[-L_r\left(2L_mL_r - {L_m}^2\right) + \left(2L_mL_r - {L_m}^2\right)(L_m - L)\right]$$

$$a_2 = \left(2L_m - \frac{{L_m}^2}{L_r}\right)\left(2L_mL_r - {L_m}^2\right)(L_m - L) + \frac{{L_m}^2}{L_r}\left[-L_r\left(2L_mL_r - {L_m}^2\right) + \left(2L_mL_r - {L_m}^2 - {L_r}^2\right)(L_m - L)\right] - 2{L_m}^2{L_r}^2$$

$$a_o = {L_m}^2{L_r}^2 \quad \text{..... (4.20)}$$

Then coefficients should be calculated.

Sets of the torque characteristics for several constant stator currents I_s, with consideration of magnetic saturation for the same induction motor is tabulated in the subsequent tables.

Table 4-5: Torque value for I_s = 12A, Considering Magnetic Saturation

I_s	12	12	12	12	12
I_{ds}	11.758	10.998	8.459	7.200	0.000
I_{qs}	2.400	4.800	8.512	9.600	12.000
φ_{dr}	1.500	1.500	1.500	1.500	1.500
$r = (I_{sq}/I_s)$	0.200	0.400	0.709	0.800	1.000
L_m	0.128	0.136	0.177	0.208	0.000
$L_{\sigma r}$	0.0105	0.0105	0.0105	0.0105	0.0105
L_r	0.138	0.147	0.188	0.219	0.011
L	0.011	0.012	0.021	0.029	0.000
τ_e	9.979	20.056	36.161	41.127	0.000

Table 4-6: Torque value for I_s = 9A, Considering Magnetic Saturation

I_s	9	9	9	9	9
I_{ds}	8.818	8.249	7.200	5.400	0.000
I_{qs}	1.800	3.600	5.400	7.200	9.000
φ_{dr}	1.500	1.500	1.500	1.500	1.500
$r = (I_{sq}/I_s)$	0.200	0.400	0.600	0.800	1.000
L_m	0.170	0.182	0.208	0.278	0.000
$L_{\sigma r}$	0.0105	0.0105	0.0105	0.0105	0.0105
L_r	0.181	0.192	0.219	0.288	0.011
L	0.019	0.022	0.029	0.051	0.000
τ_e	7.629	15.316	23.134	31.220	0.000

Table 4-7: Torque value for I_s = 6A, Considering Magnetic Saturation

I_s	6	6	6	6	6
I_{ds}	5.879	5.499	4.800	3.600	0.000
I_{qs}	1.200	2.400	3.600	4.800	6.000
φ_{dr}	1.500	1.500	1.500	1.500	1.500
$r = (I_{sq}/I_s)$	0.200	0.400	0.600	0.800	1.000
L_m	0.255	0.273	0.313	0.417	0.000
$L_{\sigma r}$	0.0105	0.0105	0.0105	0.0105	0.0105
L_r	0.266	0.283	0.323	0.427	0.011
L	0.043	0.050	0.065	0.116	0.000
τ_e	5.187	10.400	15.673	21.069	0.000

Table 4-8: Torque value for I_s = 3A, Considering Magnetic Saturation

I_s	3	3	3	3	3
I_{ds}	2.939	2.750	2.400	1.800	0.000
I_{qs}	0.600	1.200	1.800	2.400	3.000
φ_{dr}	1.500	1.500	1.500	1.500	1.500
$r = (I_{sq}/I_s)$	0.200	0.400	0.600	0.800	1.000
L_m	0.510	0.546	0.625	0.833	0.000
$L_{\sigma r}$	0.0105	0.0105	0.0105	0.0105	0.0105
L_r	0.521	0.556	0.636	0.844	0.011
L	0.174	0.198	0.260	0.463	0.000
τ_e	2.646	5.298	7.966	10.666	0.000

The corresponding torque results can be shown in Figure 4-11.

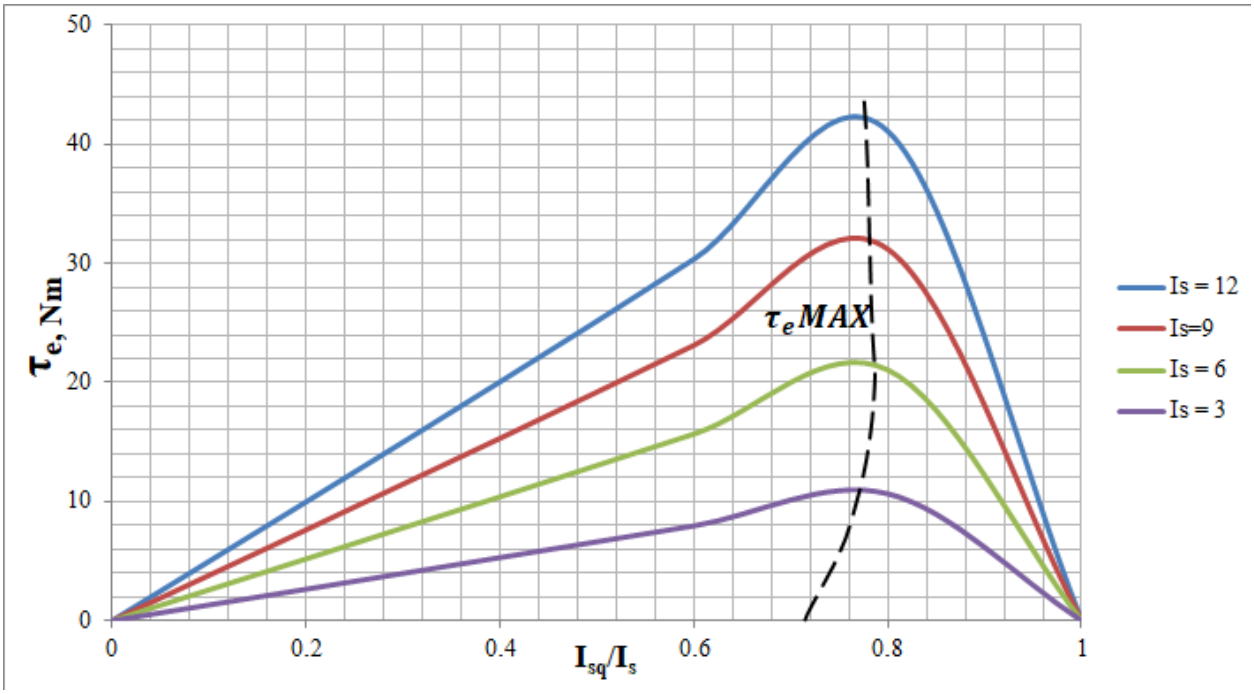

Figure 4-11: Torque characteristics for different I_s values (with Magnetic Saturation)

It is obvious from Figure 4-11that the calculated torque taking into account magnetic saturation is reduced as compared to Figure 4-7. This is because of the reduction of mutual inductance value L_m in *(4.16)*, according to the magnetizing curve in Figure 4-9.

The maximum torque curve can be obtained by changing the magnetizing current in the range $0 \leq Im \leq Imax$. from the magnetizing curve shown in (Figure 4-8) the inductances L_m, L and L_r are calculated for the corresponding level of saturation. Then, equation (4.19) is solved for the root r_{max} which satisfies $0 \leq r_{max} \leq 1$.For the corresponding value of magnetizing current I_m, the minimal steady state current is obtained as:

$$I_m = \sqrt{(I_{sd} + I_{rd})^2 + \left(I_{sq} + I_{rq}\right)^2}$$

$$I_m = I_s\sqrt{1 + \left(1 - 2\frac{L_m}{L_r} + \frac{{L_m}^2}{{L_r}^2} - 1\right)\left(\frac{I_{sq}}{I_s}\right)^2}$$

$$I_m = I_s\sqrt{\left(1 - 2\frac{L_m}{L_r} + \frac{{L_m}^2}{{L_r}^2}\right)(r)^2} \quad \text{.. ..(4.21)}$$

$$I_{smin} = \frac{I_m}{\sqrt{1-\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right){r^2}_{max}}} \quad \text{.. ..(4.22)}$$

The appropriate magnetizing stator current reference is given by

$$I_{sd,ref} = \sqrt{I^2{}_{smin} - I^2{}_{sq}} = I_{smin}\sqrt{1 - r^2{}_{max}} \quad \text{.. ..(4.23)}$$

This gives us the steady state operation of IM drive with minimal stator current for the required load torque.

CHAPTER FIVE

5. RESULT AND DISCUSSION

The effect of parameter variation in IM is discussed hereunder. Generated reports from simulation results using the 3-KW induction motor whose parameters given in Table 3-1 are presented and briefly discussed.

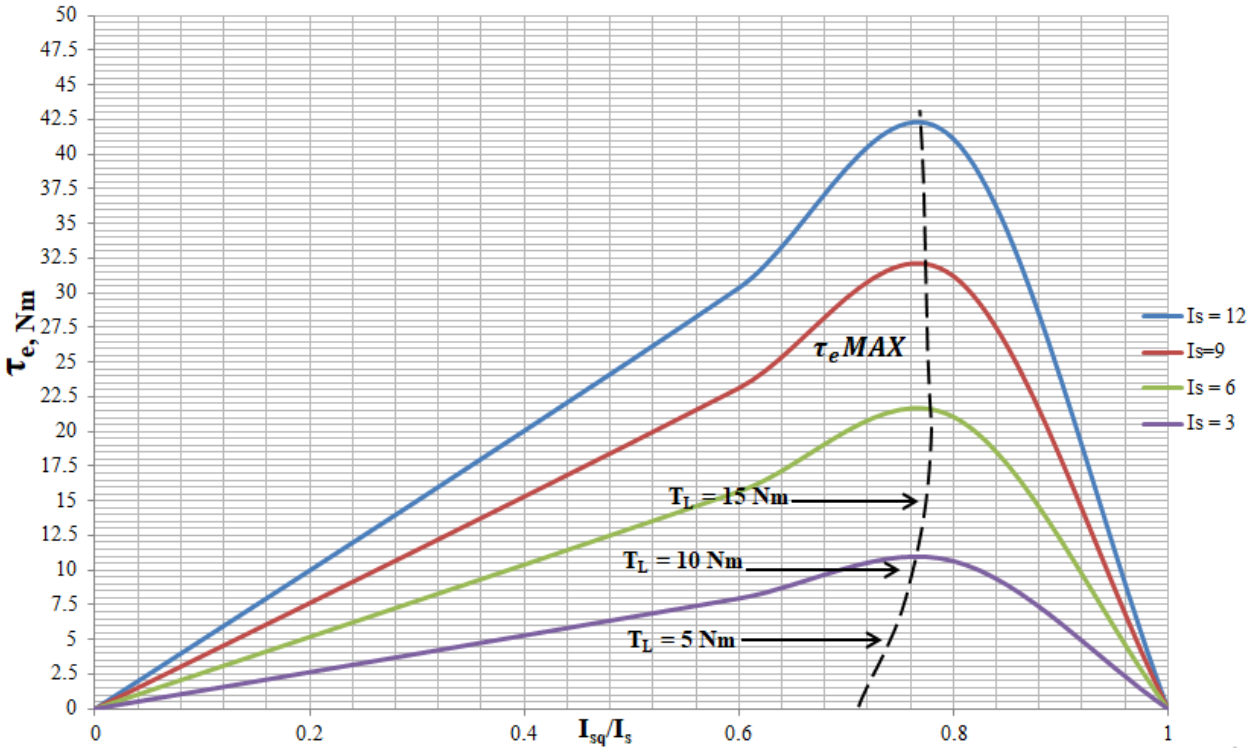

Figure 5-1: Selection of Minimum Input Current for the Corresponding Load Torque

Figure 5-1 extracted from the analytical analysis should be confirmed through cRIO simulation with the minimum input current obtained from equations (*4.21-4.23*).

The aim is to run the motor at a constant rotor speed in the domain where the motor torque is nearly equal to the load torque. This can be seen from the following simulation results.

Using the methodology and modeling of the system stated in chapter four, maximum torque is determined for different values of load torque. The successful result is obtained from cRIO and RFOC algorithm constructed using LabVIEW platform.

Now L_m is a dynamically adjusted variable which are obtained from the tabulated values of Table 4-5 to Table 4-8. These values are made using the magnetizing curve shown at Figure 4-9 with the modified mathematical model.

At the beginning of the test the motor is set to accelerate to 50 rad/s with $I_{sd,ref}$ = 9A. Then the load torque with the value of τ_L=5Nm, τ_L=10Nm, and τ_L=15 Nm are applied on it. The value of d-axis stator current is slowly decreased up to the minimum stator current for the corresponding load torque. Then the speed controller increased the q-axis stator current command to indemnify the reduced I_m.

The result of the reduced stator current is shown in Table 5-1.

Table 5-1: The Result of Minimum Stator Current

τ_L (Nm)	5	10	15
Reduced Is-min values			
I_{sd} (A)	3.6	4.7	5.7
I_{sq}(A)	4.0	5.1	7.2
I_{smin}(A)	5.4	6.9	9.1

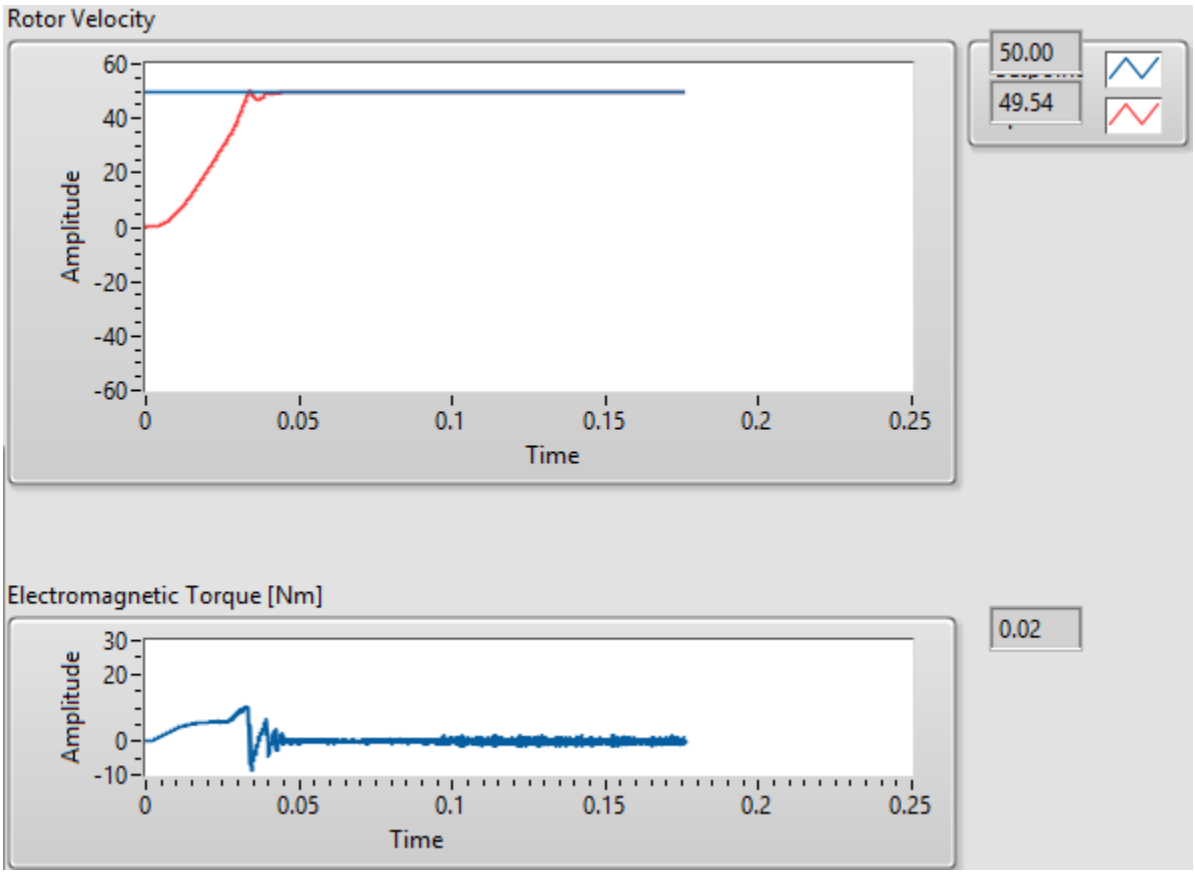

Figure 5-2: Initial response of the Motor with no-load

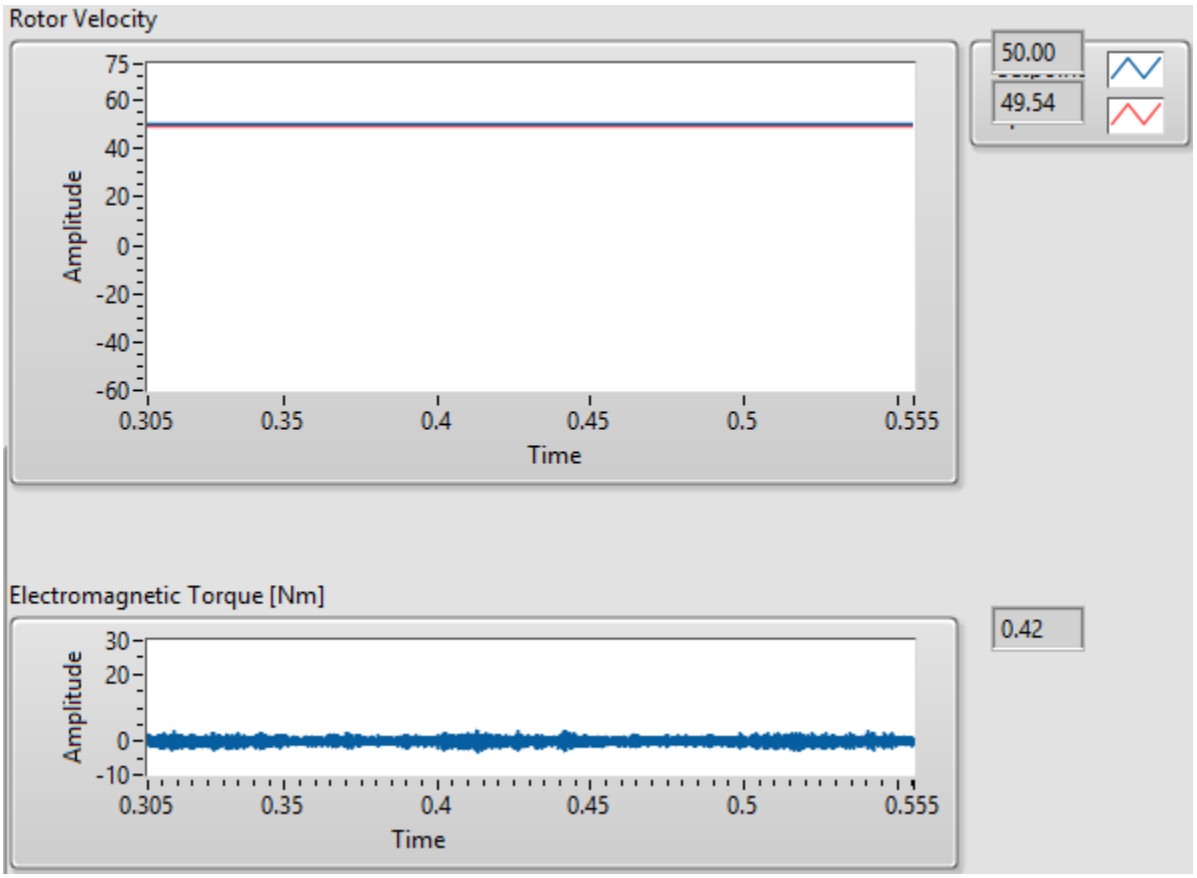

Figure 5-3: Rotor Velocity Settled to the desired speed

Now after 1.09 sec, a load of 5Nm is applied and the following results are extracted.

Applying 5NM load torque, $\tau_{L.}$

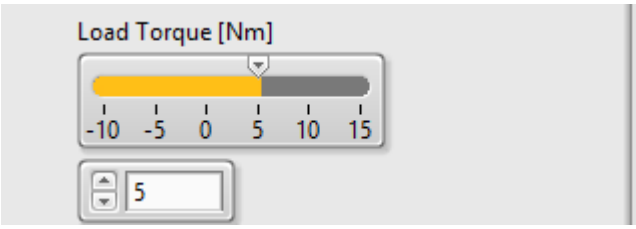

Figure 5-4: 5Nm Load Torque

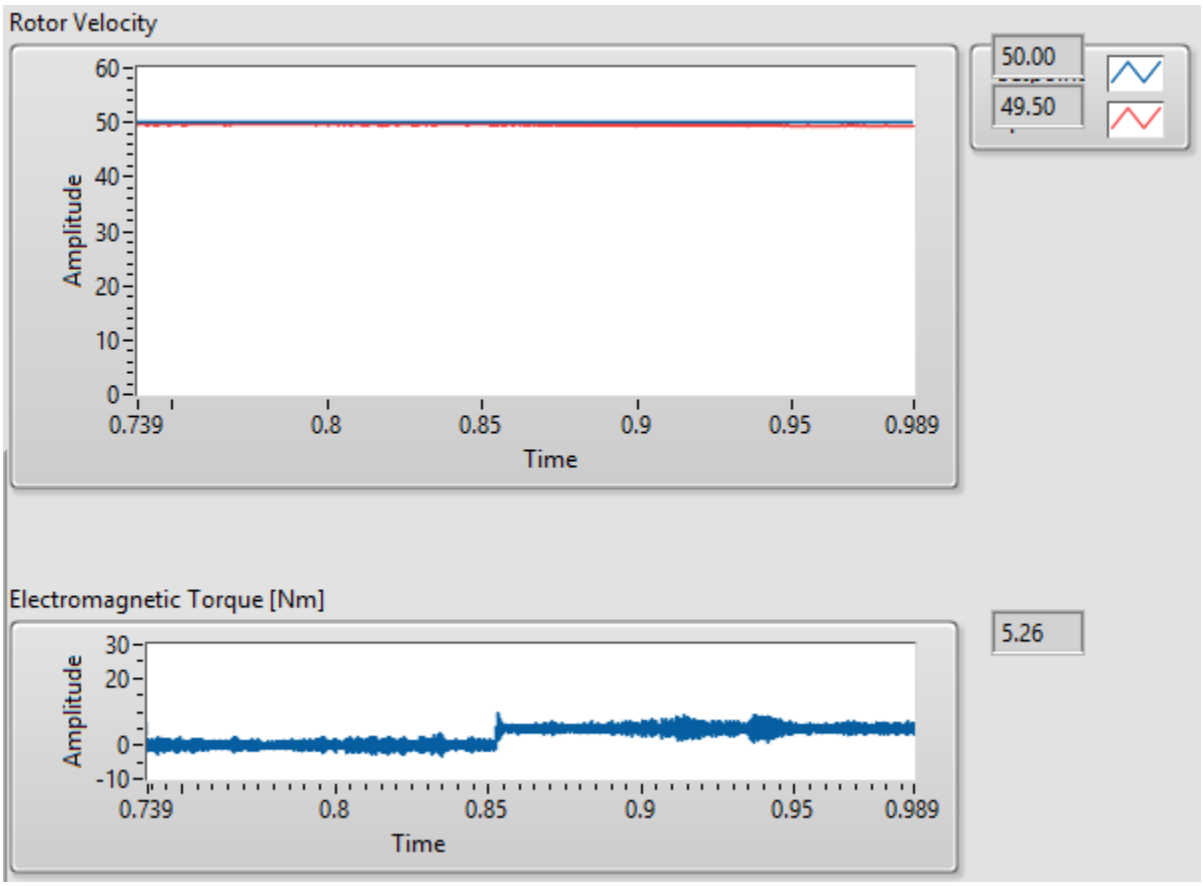

Figure 5-5: Response after 5Nm applied Load Torque

As can be seen from Figure 5-5, the aim is achieved successfully that the rotor runs at constant speed in the domain where τ_e is nearly equals to τ_L. The speed is not affected by the applied load. This implies that with saturation effect is considered and with the application of minimum input stator current, we can get enough torque as required by the load. Then again a load torque of 10Nm and 15Nm are applied and the following responses are observed.

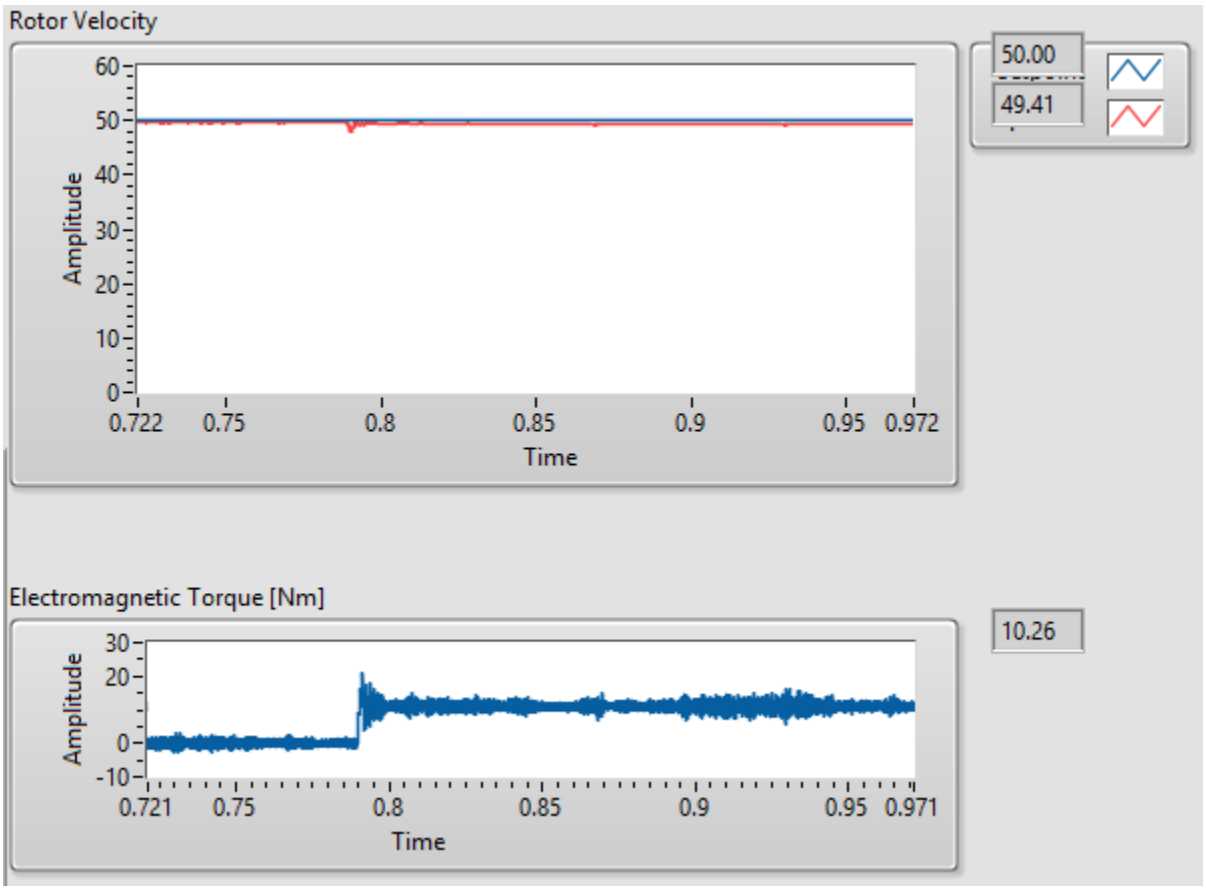

Figure 5-6: After applying 10 Nm Load Torque

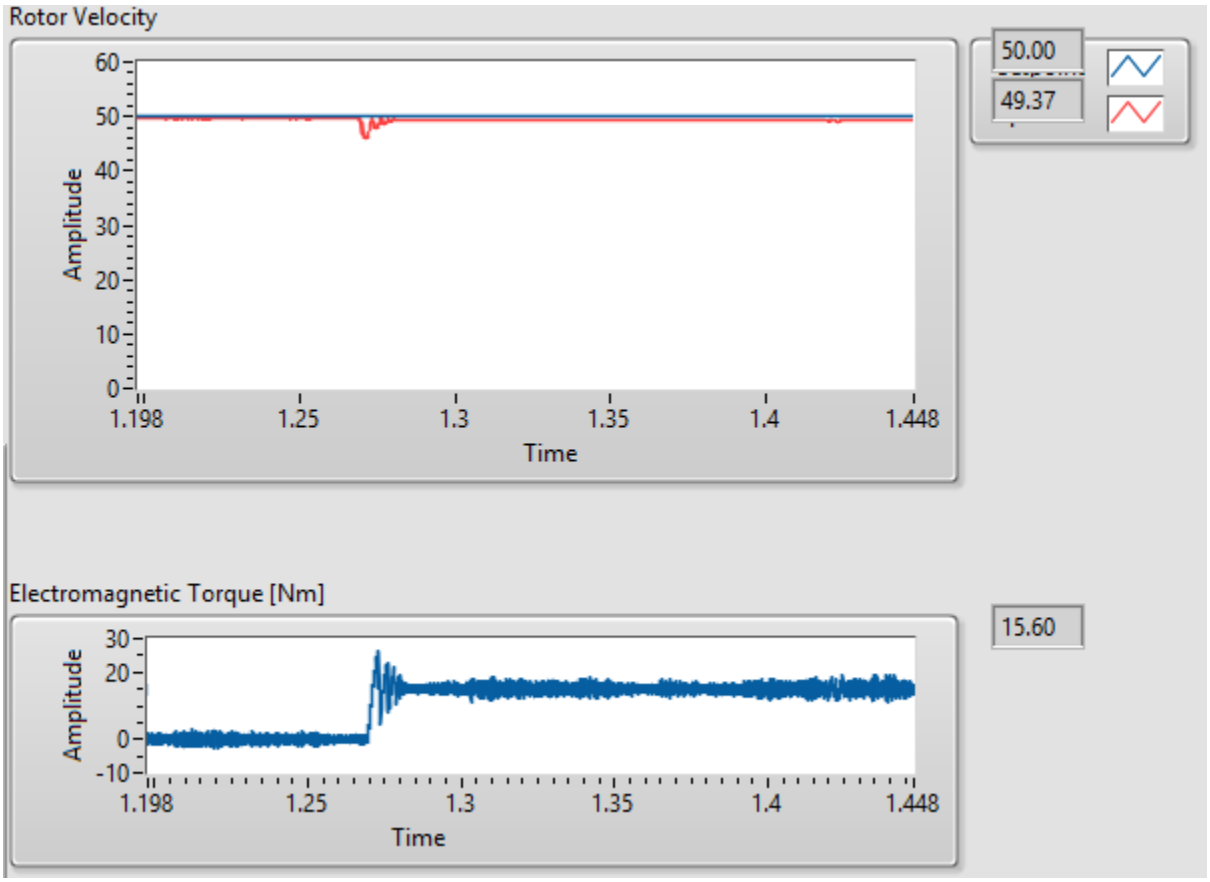

Figure 5-7: After applying 15Nm Load Torque

Now, the effect of loss minimization is explained hereunder.

One of the main power losses of an induction machine is copper loss which is given by;

$$P_{cu} = \frac{3}{2}\left({i_{sd}}^2 + {i_{sq}}^2\right)R_s \qquad \ldots\ldots (5.1)$$

Using the result obtained in Table 4-1 and Table 5-1, the power losses are calculated independently for both without saturation condition and with saturation condition in Table 5-2 and Table 5-3 respectively.

Table 5-2: Power loss Calculation without considering saturation

Case-I Without considering Saturation			
I_{sd} (A)	11	9.6	7.2
I_{sq} (A)	4.8	7.2	9.6
R_s	1.976	1.976	1.976
P_{cu}	**426.93**	**426.82**	**426.82**

Table 5-3: Power loss Calculation considering saturation

Case-II Considering Saturation			
I_{sd} (A)	3.6	4.7	5.7
I_{sq} (A)	4.0	5.1	7.2
R_s	1.976	1.976	1.976
P_{cu}	**85.84**	**142.57**	**249.95**

CHAPTER SIX

6. CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

After the accomplishment of this thesis work, the following conclusions are made.

Induction motor drive with RFOC algorithm is very sensitive to parameter variation. The critical parameter is rotor time constant which is expressed as a function of mutual inductance and rotor resistance. This thesis work deals with the effect of the mutual inductance variation. The mutual inductance variation is due to main flux path saturation.

As seen from the effect of the variation of mutual inductance, the produced electrical torque is different than expected. It is possible to get a good result which minimizes the effect of this torque variation by a means of an improved model with a tabulated magnetizing curve.

LabVIEW cRIO with rotor field oriented control techniques can be used as a useful tool for control and investigation of parameter variation of induction motor.

Taking into account magnetic saturation, the operating region of the maximum input current of the induction motor drive can be limited. Therefore the torque production loss $\Delta\tau e$ of the motor is evident that unsuitable magnetizing reference value could be used if saturation effect is not considered. This contributes significantly to the increment of Copper Loss (P_{cu}) power losses under steady-state, which in turn contributes on decrease on efficiency of the motor.

6.2 Recommendations

Based on this thesis work the following points are recommended.

- Further studies can be done to analyze other parameter variations and effects of saturation.
- Since cRIO offers powerful standalone embedded execution for deterministic LabVIEW real-time applications, practitioners and academicians should give due focus.
- Because of its strong graphical interface, flexibility of its programming language, and data flow visualization, academicians in higher education should give enough priorities for LabVIEW.

REFERENCES

[1]. *Study of Main Flux Saturation Effects in Field-Oriented Induction Motor Drives.* **Levi, Emil and Vuckovic, V.** 1989, IEEE, pp. 219-224.

[2]. *A Novel Approach to Incorporate the Main Flux Saturation Effect in a Three-Phase Induction Machine during Motoring and Plugging.* **Pahwa, Vivek and Sandhu, K S.** 2011 : s.n., 2011, International Journal of Computer and Electrical Engineering, pp. 443-448.

[3]. *Investigation on the Effects of Magnetic Saturation in Induction Machines During Transients.* **Conradi, Alexander , Schmülling, Christoph and Schmülling , Stefan.** 2013, GSTF Journal of Engineering Technology (JET).

[4]. *Main Flux Saturation Compensation in Sensorless Vector Controlled Induction Machines for Operation in the Field Weakening Region.* **Levi, E. and Wang, M.** 1999, EPE, pp. 1-9.

[5]. **Paul C. Krause, Oleg Wasynczuk, Scott D. Sudhoff.** *Analysis of Electric Machinery and Drive Systems.* United States of America : Wiley Interscience, 2002.

[6]. *FPGA Implementation for Speed Monitoring and Speed Control of AC Motor using V/F Control.* **C. Nandhini, M. Jagadeeswari.** 5, India : s.n., May 2016, International Journal Of Innovative Research in Science, Engineering and Technology, Vol. 5.

[7]. **Surti, Ammar Quaaid.** *Speed Control of Induction Motor Using FPGA based NI-GPIC Board.* Northridge : California State University, 2015.

[8]. *Saturation Effects in Field-Oriented Induction.* **Robert D. Lorenz, Donald W. Novotny.** 1990, IEEE TRANSACTIONS ON INDUSTRY APPLICATIONS, pp. 283-289.

[9]. **National Instruments.** *Getting Started with NI CompactRIO Control and Mechatronics Bundle.* s.l. : National Instruments, April 10, 2012.

[10]. **Kaur, Navneet.** Development of CompactRIO based PID Tempreture Controller. *Thesis submitted in partial fulfillment of the requirement for the award of degree of Master of ENgineering in Electronics Instrumentatio and Control.* s.l. : Thapar University, June 2009.

[11]. **Bitter, Rick, Mohiuddin, Taqi and Nawrocki, Matt.** *LabVIEW Advanced Programming Techniques.* New York : Taylor & Francis Group, 2007.

[12]. **National Instruments.** *LabVIEW Fundamentals.* Texas : National Instruments Corporation, 2007.

[13]. *Mutual Inductance Variation Influence on Induction Motor IFOC Drive.* **Milan Mijalkovic, Petar Petrovic.** 2010, Electronics, pp. 71-76.

[14]. *Impact of Magnetic Saturation on the Steady-State Operation of the Controlled Induction Motor.* **Ljusev, Petar, Dollinar, Drago and Stumberger, Gorazd.** 2003, Electrotechnical Review, pp. 190-195.

[15]. **Okoro, Ogbonnaya Inya.** *Dynamic and Thermal Modelling of Induction Machine with Non-Linear Effects.* Kassel : Kassel University Press, 2002.

[16]. *An improved methodology for dynamic modelling and simulation of electromechanically coupled drive systems: An experimental validation.* **ERDOGAN, NUH , HENAO, HUMBERTO and GRISEL, RICHARD.** 2015, Indian Academy of Sciences, pp. 2021-2043.

[17]. *A Novel Approach to Incorporate the Main Flux Saturation Effect in a Three-phase Induction Machine during Motoring and Plugging.* **Sandhu, K. S. and Pahwa, Vivek.** 2011, International Journal of Computer and Electrical Engineering, pp. 443-448.

APPENDIX

A. Derivation

The steady-state rotor currents I_{rd} and I_{rq} in RFOC (φ_{rq} =0) are:

$$I_{rd} = -\frac{1}{R_r}\frac{d\varphi_{rd}}{dt}$$

$$I_{rq} = -\frac{L_m}{L_r} I_{sq} \qquad \text{…. (A.1)}$$

The value of steady-state magnetizing current I_m is according to (A.1) calculated by equation (A.2).

$$I_m = \sqrt{(I_{sd} + I_{rd})^2 + \left(I_{sq} + I_{rq}\right)^2}$$

$$= I_s\sqrt{1 + \left(1 - 2\frac{L_m}{L_r} + \frac{{L_m}^2}{{L_r}^2} - 1\right)\left(\frac{I_{sq}}{I_s}\right)^2}$$

$$= I_s\sqrt{\left(1 - 2\frac{L_m}{L_r} + \frac{{L_m}^2}{{L_r}^2}\right)(r)^2} \qquad \text{…. (A.2)}$$

Solve (4.19) for the root r_{max} which satisfies$0 \le r_{max} \le 1$. For the corresponding value of magnetizing current *Im*, the minimal steady-state stator current according to (A.2) is given by (A.3)

$$I_{smin} = \frac{I_m}{\sqrt{1-\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right){r^2}_{max}}} \qquad \text{…. (A.3)}$$

Derivation of (4.14) yields (A.4):

$$\left(\frac{dT_e}{dr}\middle| r_{max}\right) = \frac{d}{dr}\left(P\frac{L^2 m}{L_r} I^2{}_s r\sqrt{1-r^2}\middle| r_{max}\right)$$

$$= PI^2{}_s\left[\left(\frac{d}{dr}\left(\frac{{L_m}^2}{L_r}\right)\middle| r_{max} r_{max}\sqrt{1-{r_{max}}^2} + \frac{{L_m}^2}{L_r}\left(\frac{1-2{r^2}_{max}}{\sqrt{1-{r^2}_{max}}}\right)\right)\right] = 0 \qquad \text{…. (A.4)}$$

Derivation of L_m/L_r in (A.4) yields (A.5)

$$\frac{d}{dr}\left(\frac{{L_m}^2}{L_r}\right) = \frac{2L_m\frac{dL_m}{dr} - \frac{dL_r}{dr}{L_m}^2}{{L_r}^2} = \left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right)\frac{dL_m}{dr} \quad \text{…. (A.5)}$$

Because; $L_r = L_{r1} + L_m$

$$\frac{dL_r}{dr} = \frac{dL_m}{dr} \quad \text{… (A.6)}$$

Derivation of the mutual inductance L_m with respect to the ratio 'r' yields (A.7);

$$\frac{dL_m}{dr} = \frac{dL_m}{dI_m}\frac{dL_m}{dr} = \frac{d}{dI_m}\left(\frac{\varphi_m}{I_m}\right)\frac{dI_m}{dr}$$

$$= \frac{d\varphi_m}{dI_m}\frac{I_m - \varphi_m}{{I^2}_m}\frac{dI_m}{dr} = -\frac{L_m - L}{I_m}\frac{dI_m}{dr} \quad \text{…. (A.7)}$$

Derivation of I_m with respect to the r, according to (A.2), yields (A.8)

$$\frac{dI_m}{dr} = I_s\frac{1}{\sqrt{1-\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right)r^2}}\frac{d}{dr}\left[1-\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right)r^2\right]$$

$$= {I_s}^2\frac{1}{2I_m}\left[-\frac{d}{dr}\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right)r^2 - 2r\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right)\right]$$

$$= \frac{{I_s}^2}{I_m}\left[\frac{2L_mL_r - {L_m}^2 - {L_r}^2}{{L_m}^3}\frac{dL_m}{dr}r^2 - r^2\frac{2L_mL_r - {L_m}^2}{{L_r}^2}\right] \quad \text{…. (A.8)}$$

Equation (A.8) is inserted in (A.7) and equation (A.9) is obtained.

$$\frac{dI_m}{dr} = -\frac{L_m - L}{I_m}\frac{{I_s}^2}{I_m}\left[\frac{2L_mL_r - {L_m}^2 - {L_r}^2}{{L_m}^3}\frac{dL_m}{dr}r^2 - \frac{2L_mL_r - {L_m}^2}{{L_r}^2}r\right] \quad \text{…. (A.9)}$$

From equation (A.9) dL_m/d_r is expressed as;

$$\frac{dL_m}{dr} = \left[1 + \frac{2L_mL_r - {L_m}^2 - {L_r}^2}{{L_r}^3}(L_m - L)\frac{{I_s}^2}{{I_m}^2}r^2\right]\frac{dI_m}{dr} = \frac{2L_mL_r - {L_m}^2}{{L_r}^2}(L_m - L)\frac{{I_s}^2}{{I_m}^2}r$$

…. (A.10)

i.e.

$$\frac{dL_m}{dr} = \frac{\frac{2L_mL_r - {L_m}^2}{{L_r}^2}(L_m - L)\frac{{I_s}^2}{{I_m}^2}r}{1+\frac{2L_mL_r - {L_m}^2 - {L_r}^2}{{L_r}^3}(L_m - L)\frac{{I_s}^2}{{I_m}^2}r^2} \quad \ldots (A.11)$$

The ratio I^2_s / I^2_m is determined from (A.2)

$$\frac{{I_s}^2}{{I_m}^2} = \frac{1}{1-\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\right)r^2} = \frac{{L_r}^2}{{L_r}^2 - (2L_mL_r - {L_m}^2)r^2} \quad \ldots. (A.12)$$

and inserted in (A.11) to obtain:

$$\frac{dL_m}{dr} = \left[L_r\left(2L_mL_r - {L_m}^2\right)(L_m - L)r\right] \Big/ \left\{{L_r}^3 + \left[-\left(2L_mL_r - {L_m}^2\right)L_r + (2L_mL_r - {L_m}^2 - {L_r}^2)(L_m - L)\right]r^2\right\}$$

…. (A.13)

The equation (A.4) obtains the following form:

$$\left(\left(2\frac{L_m}{L_r} - \frac{{L_m}^2}{{L_r}^2}\left(\frac{dL_m}{dr}\right)\middle| r_{max}\right) r_{max}\sqrt{1 - {r_{max}}^2} + \frac{{L_m}^2}{L_r}\frac{1-2{r_{max}}^2}{\sqrt{1-{r_{max}}^2}} = 0\right)$$

…. (A.14)

$$r_{max} = \frac{I_{sq}}{I_s}$$

$$= \left\{\left(2L_m - \frac{{L_m}^2}{L_r}\right)\left(2L_mL_r - {L_m}^2\right)(L_m - L) + 2\frac{{L_m}^2}{L_r}\left[-L_r\left(2L_mL_r - {L_m}^2\right) + (2L_mL_r - {L_m}^2 - {L_r}^2)(L_m - L)\right]\right\}{r^4}_{max} +$$

$$\left\{\left(2L_m - \frac{{L_m}^2}{L_r}\right)\left(2L_mL_r - {L_m}^2\right)(L_m - L) + \frac{{L_m}^2}{L_r}\left[-L_r\left(2L_mL_r - {L_m}^2\right) + (2L_mL_r - {L_m}^2 - {L_r}^2)(L_m - L)\right] - 2{L_m}^2{L_r}^2\right\}{r^2}_{max} + {L_m}^2{L_r}^2 = 0 \quad (A.15)$$

For the required load torque τ_l, minimal stator Current I_{smin} is calculated according to (4.14) with (A.16).

$$I^2{}_{smin} = {T_1}\Big/\left[P\frac{L_m{}^2}{L_r}\sqrt{1-r^2{}_{max}}r_{max}\right]$$

$$I_{sd,ref} = \sqrt{I^2{}_{smin} - I^2{}_{sq}} = I_{smin}\sqrt{1-r^2{}_{max}} \qquad \text{(A.16)}$$

[14]