UNIVERSITY OF

SUSSEX

AT BRIGHTON

STATOR FAULT DIAGNOSIS IN INDUCTION MOTORS

Müslüm ARKAN

Submitted for the degree of Doctor of Philosophy

School of Engineering

University of Sussex

Brighton

January 2000

To

My Mother and Father

Declaration

I hereby declare that this thesis has not been submitted, either in the same or different form, to this or any other university for a degree.

Signed ....

Müslüm ARKAN

THE UNIVERSITY OF SUSSEX

D.Phil Thesis

I <u>Müslüm ARKAN</u> author of the thesis **<u>Stator Fault Diagnosis in Induction Motors</u>** give permission to the Librarian of the University of Sussex to make available for inter-library loan and for photocopying, the above mentioned thesis which is deposited in the University Library. I agree that the abstract of the thesis may be published in such lists of sources as may be approved from time to time by the Librarian.

Signed ....

Date ....

Acknowledgement

I would like to express my sincere gratitude to my supervisor Dr. P. J. Unsworth for his suggestions, encouragement, patience and help throughout the duration of my study at the University of Sussex.

My many thanks also go to my fellow members of the intelligent motor control group Shiping, Dora, Hakan, Sarat and my current fellows Yanzhen, Simon, Dragica, Srdjan and Mongol for being so supportive. I would also like to thanks Dr. P. Lindon for his helpful discussions.

I wish to thank Prof. E. Spooner for his valuable comments.

I would also like to express my gratitude to my parents, two sisters and two brothers for their understanding, support, and love during my study.

I wish to thank my friends Nusret, Abdulkadir, Ibrahim, Burhan and other Turkish students at Sussex, and Dr. F. Yalcinkaya and Dr. M.E. Tagluk for their help and guidance in my study at Sussex University.

I would like to thank the technical support staff in the School of Engineering for their help. My special thanks go to Mr. B. Jackson.

I am grateful to Prof. Dr. M. Koksal and Dr. S. Herdem for all their help, encouragement, and effort in scholarship arrangement. I also wish to thanks Prof. Dr. A. Nacaroglu for his support and timely advice.

Finally, I gratefully acknowledge the help from the Government of Turkey, and Inonu University for financing my studies at the University of Sussex. I would also like to thank Allen-Bradley Co. Inc., Milwaukee, Wisconson, USA, for their collaboration in this research project and finance for laboratory equipment.

THE UNIVERSITY OF SUSSEX

STATOR FAULT DIAGNOSIS IN INDUCTION MOTORS

Submitted for the degree of Doctor of Philosophy

Müslüm ARKAN

ABSTRACT

This thesis presents a non-invasive and on-line method for detection of stator winding faults in 3-phase induction motors from observation of *negative sequence supply current*.

The consequences of unexpected failure of critical equipment include extended process shutdown, costly machinery repair, and health and safety problems. It is known that approximately 36% of induction motor failures are caused by failure of the stator winding, and it is believed that these faults start as undetected turn-to-turn faults within a coil, which lead to catastrophic phase-to-phase or phase-to-ground short circuit faults. To achieve prior warning of failure, shorted turns within a stator winding coil must be detected or predicted via on-line monitoring. The brief time between an initial turn-to-turn fault and final failure means that the fault diagnosis must be on-line.

A Power Decomposition Technique (PDT) has been used to derive positive and negative sequence components of measured voltages and currents. A fault detection algorithm has been developed to characterize the effects of supply unbalance and non-linear motor effects (motor iron saturation, winding unbalance, rotor static eccentricity), which also generate negative sequence current. The characterization can be performed when the motor is manufactured, or by an automated learning process during the early life of the motor. Compensation for this negative sequence current before making the fault decision enables a high fault sensitivity to be achieved.

Computer simulation models of an induction motor with stator asymmetry are presented, together with hardware and software designs for implementation of the fault detection system. Experimental results are included.

Contents

List of Symbols

f_s	Supply frequency
f_r	Rotation frequency
α	Operator for symmetrical components ($e^{j\frac{2\pi}{3}}$)
a, b, c	Subscript symbols for the quantities in *abc* three-phase system
0, p, n	Subscript symbols for zero, positive, and negative sequence components
i, v or u	Instantaneous current and voltage
α , β	Subscript symbols for the quantities in $\alpha - \beta$ reference frame
q, d	Subscript symbols for the vector quantities in *q-d* reference frame
p, q	Instantaneous active and reactive powers
P, Q	Fundamental active and reactive powers
T_s	Supply period
I, V, Z	Magnitude values for current, voltage, and impedance
I, V, Z	Phasor values for current, voltage, and impedance
i, v	Superfix symbols for current and voltage related
s, r	Subscripts symbols for the quantities of stator and rotor
g	Subscript symbol for the air gap of motor
p	Operator for *d/dt*
$\dot{\mathbf{I}}, \dot{\mathbf{V}}$	Spiral vector quantities for current and voltage
ω_s , ω_m	Angular synchronous and rotor speed
s_p , s_n	Positive and negative sequence slip values
ext	Subscript symbol for the external quantities

N_s	Number of samples in per supply cycle
ref	Subscript symbol for reference quantities
av	Subscript symbol for average values
x, y	Real and imaginary part of phasor (or complex number)
δ	delay angle
h	Subscript symbol for the healthy condition of motor
m, s, l, v	Subscript symbols denoting motor, supply, load and voltage related quantities
T	3 to 2 transformation matrix
N or N_s, N_r	Number of stator and rotor winding turns
μ_0	Permeability of free space
ψ	Flux linkages
sh	Subscript and superscript symbol for shorted turn

List of Figures

List of Tables

Chapter 1

Introduction

1.1 General Introduction

A trend in modern industry is to focus attention and resources on fault detection and preventive maintenance strategies for industrial plant. Preventive maintenance of electric machines is essential for any industry. The consequences of unexpected critical equipment failure include costly machinery repair, extended process down time, and health and safety problems. On the other hand, frequent and unnecessary protective shutdowns may be more harmful than losing a machine itself through lost production. Today, the availability of low cost computing power and the desire to improve the reliability of electrical drives has led to the development of more powerful diagnostic techniques for electrical machines. As a result, more and more industries are now embarking upon the evolution of modern continuous condition monitoring techniques which can provide early warning of developing faults in critical machines, to enable a pre-planned and well organized shut-down of the plant for repair or replacement of faulty machines.

In this chapter, a general introduction to motor failure prediction is given, and an overview of the aim of this research work, and a preview of the contents of the thesis.

1.2 Induction Motor Faults

Although induction motors are rugged and robust, they are not 100% fault free. As with all mechanical devices, failures do occur. In induction motors, failure can be electrical or mechanical or a combination of both. Electrical or mechanical faults are usually proceeded by deterioration of one of the mechanical, electrical, magnetic, insulation or cooling components of the machine.

The most common faults in squirrel-cage induction motors are *bearing, stator* and *rotor faults* [5,6,7,8,9,10]. Surveys conducted by the IEEE [9] and EPRI [10] have given extremely useful information. The IEEE survey classified the "*failure contributions*" in electrical related, mechanical related, environmentally and maintenance related, and other categories. On other hand, the EPRI survey identified "*percentage failure by components*". Table 1.1 shows the result of these surveys. The surveys show that the most common electrical fault in induction motor is stator failure (36%).

The majority of all stator and rotor failures are caused by a combination of various stresses that act on these components [8]. For the *stator*, these can be grouped as follows

- *Thermal*: ageing, overloading, cycling
- *Electrical*: dielectric, tracking, corona, transients
- *Mechanical*: coil movement, rotor strike, miscellaneous
- *Environmental*: moisture, chemical, abrasion, foreign objects

Under ideal conditions, modern insulation systems can be expected to have a life cycles in excess of 100,000 operating hours [20]. In most any industrial application however, motor insulation systems are routinely subjected to a combination of the above stress factors that shorten their life expectancy.

Survey name	EPRI Survey 1982 [10]	IEEE IAS Survey 1985 [9-Part I]	IEEE IAS Survey 1985 [9-Part II]
Survey size	1052 failures	380 failures	304 failures
Bearing failures (Sleeve bearings, Anti-friction bearings, Thrust bearings, Bearing seals, Oil leakage)	41%	44%	50%
Stator failures (Ground Insulation, Turn insulation, Bracing, Frame, Core)	36%	26%	25%
Rotor failures (Cage, Shaft, Core)	9%	8%	9%
Other	14%	22%	16%

Table 1. 1: **Motor failure survey for large motors [9, 10]**

Thermal stress, can be caused by thermal ageing, voltage variation, unbalanced phase voltage, cycling (frequent direct online starting), failure of ventilation, and rise in ambient temperature [8,16]. The insulation life of a machine halves for every 10 ^{o}C increase in temperature. For every 3.5% voltage unbalance per phase, the winding temperature rise will increase 25% in the phase with the highest current [8]. Frequent starting also put insulation under high current that causes temperature rises. Within certain limits, it can be estimated that the winding temperature rise will increase as the square of the load [8].

During recent years, substantial evidence has come to light indicating that a significant number of motors are exposed to *transient voltage* conditions, which result in reduced winding life or premature failures (either turn-to-turn or turn-to-ground).

Low voltage motors that are built to operate at voltage levels below 1 kV and power levels below 600 horsepower are generally random wound. Random wound coils are not made with particular attention to the order in which the turns lay in the coil. Therefore, the first turn in the random wound coil can be adjacent with the last turn, resulting in conductors with the full coil voltage potential between them in direct contact. Corona phenomena, which is a localized discharge resulting from transient gaseous ionization in insulation systems where the voltage stress has exceeded a critical value, is normally a serious problem only for operation at 4 kV and above. It could also be a problem for a small motor with seriously degraded insulation.

Mechanical stresses proportional to the square of the current, act on the coil windings, and can cause coil vibration at twice line frequency, with movement both in the radial and tangential directions. This movement can cause severe wear to the coil insulation, and loosen the topsticks.

Regardless of the cause of failure, the actual mode of stator failure can be broken down into five groups: *turn-to-turn, coil-to-coil, open circuit, phase-to-phase,* and *coil-to-ground* [8]. A turn-to-turn fault is one of the most difficult failure to detect. Depending on the type of motor protection, the motor may continue to run, but the heating in the short circuited turns will later cause severe faults like phase-to-phase or coil to ground faults. Pre-warning of severe damage can only be achieved if turn-to-turn faults within a coil can be initially diagnosed via on-line diagnostic techniques [5].

1.3 Condition Monitoring and Fault Detection Techniques

Various monitoring and diagnostic strategies have been proposed for the diagnosis of problems in induction motors. Although they can be classified in a different ways, in this subsection the best known stator related techniques will be discussed.

1.3.1 Condition Monitoring Techniques

While many approaches to fault detection have been tried, condition monitoring of induction motors is usually provided by some combination of *thermal, vibration, flux*, and *motor current* monitoring techniques. This section provides a review of the three monitoring techniques with emphasis on research into electrical condition monitoring techniques.

1.3.1.1 Vibration Techniques

A widely used and powerful technique in vibration analysis is the display of vibration amplitude as a function of frequency. In that case, it is possible to identify the sources of vibration. Identification is commonly used for rotor, bearing, misalignment, and eccentricity faults [5, 13, 17, 21]. Although stator faults effect the vibration signal at f_s, $2f_s$, and $4f_s$, dynamic unbalance and coupling misalignment also produce this effect [13]. Leonard et al [32] have also suggested that second harmonic of the fundamental frequency occur in the frame vibration spectrum due to inter-turn winding faults. It has also been shown that this frequency component can be expected when stator frame vibration is monitored during the occurrence of supply unbalance (including single phasing).

1.3.1.2 Thermal Techniques

Thermal monitoring of electrical machines is accomplished by measuring local temperatures in the stator core, windings, and at the motor bearings [13]. The measurement of temperatures of cooling air or circulating lubrication are alternatives. These temperatures provide indications of the overall condition of machine, and may warn about overload, or coolant failure. Temperature measurement is done using thermocouples or resistance temperature detectors (RTD). These sensors are installed permanently in the machine structure during manufacture. Due to the expense of the system, they are installed only in larger machines. A typical thermal monitoring system comprises a sensor, an amplifier, a meter and an alarm unit. If the temperature exceeds a predetermined value, an alarm is sounded to warn that a gross change is taking place within the machine.

1.3.1.3 Electrical Techniques

Electrical monitoring of electric machines is predominantly accomplished through the measurement of two quantities: *stator current* and *magnetic flux*. These provide indications of both the electrical and the mechanical condition of the machine.

Within the machine, magnetic flux and electric field vary circumferentially in the airgap, periodically in space, and periodically with time. Under ideal conditions these magnetic flux and electric field waveforms should be symmetrical, but electrical defects in the machine will distort them [13]. Defects in either rotor or stator disrupt the radial and circumferential patterns of flux in the machine causing changes, which can be detected outside the machine. These magnetic and electric field distortions will also alter the machine terminal voltage and current, which can be measured to give an indication of machine condition.

Motor current monitoring can provide a non-invasive, cost effective means for on line monitoring of electric machinery, without interrupting production. Recent investigations have focused on current monitoring to avoid the need for vibration transducers [5, 17]. In this approach, the motor itself acts as a bilateral transducer, converting mechanical vibrations into electrical signal perturbations. It offers means to monitor both the electrical and the mechanical condition of the machine. The current spectrum is influenced by many effects such as supply harmonics, motor quality, static and dynamic load conditions, and fault conditions. Any sudden change in the monitored frequency components in the current spectrum (not associated with change in the supply voltage and load torque) can be used as an indication of a possible failure condition.

1.3.2 Fault Detection Techniques

Research investigations have concentrated on using four methods of analysis to provide an indication of the health of the machine. These methods are parameter estimation, spectrum analysis, artificial neural networks, and symmetrical components.

1.3.2.1 Parameter Estimation Technique

The parameter estimation technique is widely used, even though it requires an accurate mathematical model and an elaborate understanding of the system dynamics based on a set of system parameters. These system parameters are usually chosen to reflect the motor conditions. A stator fault will affect the motor parameters. For example, parameter estimation of the Park's model can be used for fault detection [14]. An induction machine can be characterized by a Park's model. Firstly, a set of experiments and identifications with different operating points and loads can be used in order to constitute a parameter data base of the healthy motor. Then, during normal operation, a new set of data is calculated during operation. Any deviation from healthy motor parameter values can be used as a fault indication. The major difficulty with the parameter estimation technique is that an accurate mathematical system model (e.g., Park's model, the motor's mechanical dynamic equation etc) is required, which in most cases is difficult to obtain. Most machine dynamics are non-linear and stochastic, and many assumptions must be made regarding the system to arrive at a simple and reasonable mathematical model of the machine. Such a fault detection or diagnosis system is not robust enough in the presence of noise and perturbations [15].

1.3.2.2 Spectrum Analysis

Spectrum or signature analysis, which transforms the monitored quantity e.g. current into individual frequency components derived by Fast Fourier Transform (FFT), is commonly used to detect induction machine rotor faults. Most motor faults affect either the air gap permeance or the MMF, which cause variation in the air gap flux density. These flux variations produce stator currents at frequencies related to the fault condition. The important step is reducing the complex signal to simple frequency components which are characteristic of different fault mechanisms. Advantage of working in the frequency domain, is that different faults tend to influence different frequency components, so that the nature of the faults can be diagnosed, even in the presence of more than one fault. In spectrum analysis, the main purpose is to identify the defect frequency components and to monitor their amplitudes for trending. Detection of any failure in the frequency domain is done by comparing frequency spectra to see any amplitude changes. For this purpose, a reference spectrum is taken when the machine is in good condition. It is difficult to distinguish the frequency components by looking at the time signal, and is why spectral analysis is so useful. However, it does not mean that spectral analysis is suitable for all kinds of applications. For instance, spectral analysis is highly sensitive to minor changes in the normal operating conditions of the machine. Also there are some problems associated with signal processing techniques such as leakage, aliasing and the effect of the different weighting windows [17].

1.3.2.3 Symmetrical Components

The use of electrical models based on symmetrical components has been the traditional practice in analyzing the induction motor characteristics during unbalanced conditions. The method was introduced early in this century by Fortescue [31]. He showed that an unbalanced set of n phasors may be resolved into n-1 balanced n phase systems of different phase sequence and one zero phase sequence system. This mathematical tool has become very useful in investigating the response of any unbalanced power system. A three phase unbalanced system can be analysed in term of the positive, negative, and zero sequence components (see Chapter 2 and 3). The method of symmetrical components is analogous in some respects to the resolution of a periodic function into its fundamental and higher harmonics by Fourier series. By the symmetrical components method a set of unbalanced currents (or voltages) may be resolved into system of balanced sequence currents (or voltages) equal in number to the number of phases involved. Both zero sequence current (ground fault) and negative sequence currents (inter-turn faults) or impedances of the machine can be used to show the presence of stator faults.

1.3.2.4 Artificial Intelligence Techniques

Artificial Intelligence (AI) techniques are mainly intended to perform the tasks, which are best accomplished by human experts. Within the field of artificial intelligence, there are several different techniques, which are being actively used for providing effective and alternative solutions to the problems, which cannot be solved by traditional computing algorithms. *Artificial neural networks*, *expert systems*, and *fuzzy logic* can be considered to be the most successful AI techniques in recent technological endeavours.

The knowledge of human experts can be embodied into an expert system through a set of 'IF-THEN' rules. Once successfully tested, the resulting expert systems can be copied or reproduced to be employed at a number of similar applications. Though the resources and time needed for the development of a full-scale expert system for condition monitoring of critical machinery may be very high, the benefits that could be gained by using such systems can repay the time and cost effect [18].

Artificial neural networks can be trained to perform motor fault detection by learning experts' knowledge using a representative set of data. At the beginning of a neural network's learning (or training) session, the detection and diagnosis of the motor's condition made by the neural network fault detector will not be correct. Based on the difference between the correct decision made by the expert and the one made by the neural network, an error quantity is generated and used to adjust the neural network's internal parameters (called network weights) to yields better output that is closer to the correct decision. Once the neural network is trained appropriately, the network weights will contain the knowledge needed to perform the fault detection [15, 18].

1.4 Appraisal of Previous Works

Since there are many published papers on stator fault detection by using above mentioned techniques, the most relevant and recent papers will be cited here.

An early indicator of electrical faults in a rotating machine stator is usually an increase in electrical discharge activity. For motor and generator rated 4 kV an above, it has been known for decades that stator failure processes give rise to partial discharges (PD). There are many published papers about using PD to detected stator faults [35, 36, 37, 38, 39]. Partial discharge tests can now be performed either when the motor is not operating (off-line) or during normal operation of the motor (on-line). To reduce expertise necessary for distinguishing between PD signal and electrical noise occurring elsewhere in the plant, the on-line PD test methods are preferred by utility industry. Stator winding PD are separated from noise based on the shape of the detected pulses, that is, PD has a fast risetime, whereas noise has slow risetime. Borsi et al [41] used neural networks to separate PD from pulse shaped noise signals.

Wang [12] proposed a new method, the Zero-Crossing Times technique (ZCT) to monitor stator fault detection. In this method, instead of sampling voltage and currents with an

A/D converter, timings are recorded as waveforms cross zero. Fluctuation in the intervals between successive zero-crossings of the three phase current waveforms are analysed using Digital Fourier Transforms (DFT). Stator faults were observed from amplitude change of the spectral component at $2f_s$, where f_s is the supply frequency.

Cardoso et al [24, 28] has used the Park's Vector approach for diagnosing malfunctions, such as, single phasing, inter-turn stator fault, airgap eccentricity etc. Under ideal conditions, the motor current Park's Vector representation is a circular locus its center at the origin of coordinates. And under faulty conditions the observed picture will be differed from the reference pattern. However, they do not mentioned anything about load, voltage and supply imbalance changes, which also effect the shape of current locus.

An artificial-intelligence-based diagnostic method for induction machines has been presented by Filippetti et al [43, 44]. First they use an unsupervised NN approach and utilise for the learning process suitable diagnostic indexes. For second step they use different approaches, backpropagation NN, fuzzy ruled based system, and neuro-fuzzy adaptive NN. The attribute they used for stator fault was the component ($-f_s$) of current space vector, which corresponds to the negative sequence current component.

Penman et al [23] have used axial leakage flux resulting from stator winding inter-turn shorts for fault detection. To monitor axial leakage flux, they install a search coil placed concentric with the drive shaft of the motor. And in order to locate the fault position, they mounted four smaller coils symmetrically in four quadrants of the motor at a radius of about half the distance from the shaft to the stator endwinding. The occurrence of a fault on the motor results in a change in the air gap space harmonic distribution and search coil can detect the time harmonics of the axial flux. They used low harmonics (for their motor 48 Hz, 72 Hz, and 82 Hz) to detect shorted turns.

Thomson et al [27] have investigated inter-turn stator faults by looking at rotor slot passing frequencies in the current spectrum. Their results showed that one of the rotor slot passing frequency components in the current spectrum could be used to detect shorted turns in a stator coil. Their recent result has been mentioned in [5] for a larger motor with one turn shorted in a 15 turn coil. They observed 20 dB increase for one of rotor slot passing frequency components (1274 Hz at 1440 r.p.m.- the full load speed) in the current signal. They used high pass filter to get clear picture of the slot passing frequencies.

Natarajan [25] has used a zero sequence current detector for stator fault identification. The most common method of detecting faults is the "ground-fault detector" [19]. Kirchoff's current law dictates that the instantaneous sum of the currents in a three-phase system (which is zero sequence current), without neutral connection, must always be zero. Hence a current transformer arranged with all three leads must always indicate zero unless there is an additional return path such as ground. Although he claimed that by this method it is also possible to detect stator turn-to-turn fault, supply unbalance fault etc, as it was mentioned in the discussion section that was because of an additional motor neutral connection via ammeter.

Kohler et al [2] have used an effective negative sequence impedance, which is the ratio of negative sequence voltage to negative sequence current, as an indication of stator winding faults. This impedance method is potentially useful since it does not require motor modelling or knowledge of machine parameters and is relatively insensitive to speed changes. They used symmetrical components theory to derive negative sequence voltage and current. A fundamental limitation of this predictor is the requirement for a balanced voltage supply to the motor. Because slightly unbalanced supplies, changing in time, are relatively common in industrial power systems, this constraint on the predictor was serious.

Results of [2] and [66] have been further developed by Kliman et al [3] to detect stator turn faults. To come over the supply balance requirement, they used actual negative sequence impedance to separate negative sequence current into two parts: injected negative sequence current (fault current) and negative sequence current due to supply voltage imbalance. Within limits of the approximation, they assumed that the actual negative sequence impedance is constant.

Cash et al [26] presented a method using line-neutral voltages to detect stator insulation failure. The paper presents experimental and theoretical validation of the method. They used sum of the line-neutral voltages. Because line-neutral voltages are multiplication of phase currents and phase impedances, and the sum of the phase currents for three wire system is zero, any deviation of line-neutral voltage sum from zero is result of imbalance between phase impedances. To improve the sensitivity of method they used a band-pass filter around the fundamental. Limitation of this method is an accessibility requirement to the machine electrical neutral in its wye configuration.

1.5 Objectives of This Research

The aim of the research described in this thesis is to detect stator faults by on-line voltage and current measurement. The approach is by using positive and negative sequence voltages and currents, calculated by using a Power Decomposition Technique (PDT).

Monitoring negative sequence current offers a non-invasive method to detect stator faults. Before making any decision about motor condition from negative sequence current, other effects that may cause negative sequence current have to be eliminated. In the presented thesis these effects and problems will be discussed and elimination of them will be presented.

The following can be identified as the main tasks which have been addressed in developing the proposed fault diagnosis scheme:

- A new technique has been used to obtain negative and positive sequence components of voltage and current. Because the amount of data needed is very small, the PDT process can be carried out in real time.
- A compensation method has been introduced for non-simultaneous sampling which removes the hardware complexity and costs of simultaneous sampling.
- Semi-empirical formulae have been proposed to eliminate the effects of supply unbalance, load, and voltage variation.
- The effects of motor heating and variation in negative sequence resistance with slip change are minimised by using only negative sequence reactance to calculate supply negative sequence current.

- Change in the motor negative sequence impedance under supply unbalance with load variation has been reported. The experimental results on several motors shown that negative sequence impedance can vary between 10% to 50%.
- To reduce the amount of data to be stored for on-line system identification, a Recursive Least Square method has been used for motor characterization.
- To minimise the effects of fluctuation in current at the shaft rotation frequency, instead of sampling over one cycle, sampling over a number of cycles equal to the number of pole pairs has been used.
- With the proposed algorithm, negative sequence current as low as 0.1 percent of full load current due to shorted turns can be detected in the presence of variation in load, voltage, and unbalance of 100%, $\pm 10\%$, and $\pm 4\%$ respectively.

1.6 About the Thesis

There are *seven chapters* in the thesis. The *Chapter 2* discusses symmetrical components and their application to the induction motor. The theoretical background is given in this chapter. Equivalent circuits of induction motors for positive and negative sequences under different condition are presented. The power decomposition technique is also described in Chapter 2.

Chapter 3 describes the basic principles of the stator fault diagnosis algorithm presented in this thesis. The proposed algorithm and its requirements are presented in detail.

Chapter 4 provides the results of computer simulations of asymmetrical induction motors. In this chapter two models for asymmetrical induction motors model are discussed and simulation results have been analysed using the algorithm developed in Chapter 3.

Hardware and software used to implement the proposed method for stator fault diagnosis is presented in *Chapter 5*. The implementation of algorithm has been carried out with two configurations of hardware and software. This chapter gives details of the hardware and software developed for data processing and analysis.

Chapter 6 describes the experimental set-up, and experimental results are presented in this chapter. The sensitivity requirement is also discussed.

Chapter 7 present conclusions and suggestions for future work.

Chapter 2

Symmetrical Components and Their Application to Induction Motors

2.1 Introduction

The symmetrical component theory was introduced by Fortescue [31] in his classical paper, ”*Method of Symmetrical Co-ordinates Applied to the Solution of Polyphase Networks*”. In this work, Fortescue proved that an unbalanced system of related phasors can be resolved into a system of balanced phasors called *the symmetrical components* of the original phasors. The conventional method can be applied to any multi-phase system containing any number of phases.

In this chapter the theoretical background of symmetrical components for three-phase system will be discussed, as well as a new method that has been proposed by Huang [1] to derive sequence components of arbitrary three-phase signals in the time domain, and their application to the induction motors.

2.2 Symmetrical Components

The solution of *balanced* polyphase circuits is usually accomplished by converting the currents and applied voltages to per phase values and solving for one of the phases in a manner similar to single-phase circuits. The mutual reactions between phases may be represented by equivalent self-impedances because the symmetry of the problem determines the magnitude and phase position of the other phase currents. The currents and voltages for the other phases are equal in magnitude to those of the first but displaced symmetrically in phase position.

The solution of unbalanced polyphase circuits or balanced circuits with unbalanced terminal conditions does not permit the same simplification. By the older methods of analysis it was necessary to assign symbols to the quantities in all the phases and carry through the phase solutions simultaneously. This complicates the problem enormously. The method of *symmetrical component analysis* reduces the solution of such problems to a systematic form and results in considerable simplification, especially for balanced systems operating under some kind of unbalance such as a line-to-line, single line-to-ground, or double line-to-ground fault.

One purpose of the method of symmetrical components is to show how three unbalanced voltages can be built up in an analytical manner from the three fundamental symmetrical balanced sequence components. The question naturally arises, why should the resolution of three phasors into nine phasors necessarily simplify the solution. The answer lies in the fact that the resolution results in three symmetrical systems, each one of which can be treated independently, which may be solved by reducing the constants and voltages to per phase values and solving on a single-phase basis.

In symmetrical circuits, currents and voltages of different sequences do not react upon each other; positive-sequence voltages produce only positive-sequence currents, and negative-sequence voltages only produce negative sequence currents, and similarly, zero-sequence voltages produce only zero-sequence currents. This considerably simplifies problems involving asymmetry such as that introduced by short-circuiting conductors of a system either together or to ground, singly or in pairs, or by the open-circuiting of a conductor.

The resolution of the problem into the sequence components has a further advantage in that it isolates the quantities into components, which have a more direct influence on system behavior. A large variety of torque-speed-characteristics can be obtained by applying asymmetrical supply voltages to induction motors. The net torque of an induction motor is affected by the negative sequence voltage. Ground relays and grounding phenomena in general are very closely associated with zero-sequence components. Power quantities can also be resolved into components associated with the different sequences [49, 50, 51, 52]. These and many other subjects attest to the value of resolving the problem into the sequence components; the solution of the problem itself is not only simpler, but the results are resolved into factors which can be investigated separately.

Most of the apparatus used in practice such as generators, induction motors, transmission lines and cables (to within a sufficient degree for practical purposes) are of the symmetrical type. In fact special precautions are taken to ensure that they are symmetrical, otherwise undesirable features, such as telephone interference and extra power losses, may be introduced. It may be seen, therefore, that this simplification is applicable to the vast majority of systems in commercial use.

In unbalanced systems, currents of one sequence produce not only voltage drops of the same sequence but may produce voltage drops of the other sequences as well. In the method of symmetrical components, rules are developed for solving the fundamental relation between the sequences in such systems.

Although mostly applied to three-phase systems because of their common use, the fundamental conceptions are applicable to systems of any number of phases.

2.2.1 Meaning of Positive, Negative, and Zero Sequence; Phasors

The theory of Symmetrical Components states that, by the superposition principle in a linear network, any set of asymmetrical three-phase phasors of *abc* can replaced by the sum of three sets:

1. ***Positive-phase-sequence***: This is a system of three phasors equal in magnitude and $\frac{2\pi}{3}$ radians apart in phase, in which the time order of arrival of the phase phasors at a fixed axis of reference is the same as that of the generated voltages. The phases have been arbitrarily named so that the phase order of the generated voltages is *abc*. In Figure 2.1 (a), the system of phasors is a *positive-sequence* system.

2. ***Negative phase sequence***: The system of three phasors which are also equal in magnitude and $\frac{2\pi}{3}$ radians apart in phase, but which arrive at a fixed axis in the phase order *acb* instead of *abc*, as in Figure 2.1 (b), has consequently been called a *negative-sequence* system.

3. ***Zero-phase sequence***: If the phasors representing currents or voltages in the three phases are not separated in time phase, there will be no sequence of phases, and the currents or voltages, as the case may be, will vary identically in each phase. A phasor diagram for this system, which has been called *zero sequence,* is shown in Figure 2.1 (c). The phasors, which form the zero-sequence system, are equal in magnitude and in phase.

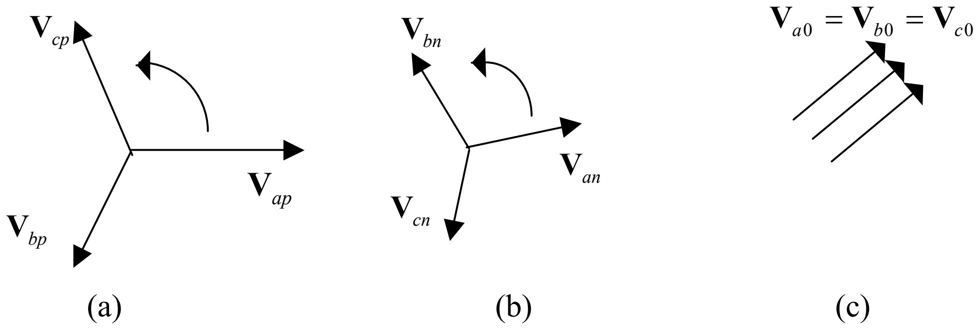


Figure 2. 1: (a) Positive, **(b)** Negative, and **(c)** Zero-sequence systems of phasors.

Any sinusoidal voltage or current can be represented in a complex exponential function of time, $v(t) = V_m e^{j(\omega t+\varphi)}$. The *phasor* of voltage or current that is used here is a complex quantity, which is the value of the complex function at *t=0.* The phasor describes the complex functions in their amplitude and phase relationship completely except for the frequency, so that the phasor of $v(t)$ is $\mathbf{V} = V_m e^{j\varphi}$ [53].

2.2.2 Symmetrical Component Equations: Resolution of Three Phasors into Their Symmetrical Components

Three given phasors $\mathbf{V}_a$, $\mathbf{V}_b$, and $\mathbf{V}_c$ can be expressed in terms of their symmetrical components by the equations

$$\mathbf{V}_a = \mathbf{V}_{a0} + \mathbf{V}_{ap} + \mathbf{V}_{an}$$

$$\mathbf{V}_b = \mathbf{V}_{b0} + \mathbf{V}_{bp} + \mathbf{V}_{bn} \qquad 2.1$$

$$\mathbf{V}_c = \mathbf{V}_{c0} + \mathbf{V}_{cp} + \mathbf{V}_{cn}$$

where known relations exist between the phasors of each of the three symmetrical sets of phasors, $\mathbf{V}_{a0}, \mathbf{V}_{b0}, \mathbf{V}_{c0}$; $\mathbf{V}_{ap}, \mathbf{V}_{bp}, \mathbf{V}_{cp}$; $\mathbf{V}_{an}, \mathbf{V}_{bn}, \mathbf{V}_{cn}$. The double subscript notation has been employed, the first subscript indicating the phase, the second the sequence. Small letters a, b and c are used to indicate the phases, while *0*, *p*, and *n* are employed to designate respectively zero, positive, and negative sequence.

With phase *a* as reference phase, the choice of the phase to be regarded as the reference phase is arbitrary, and making use of the operator $\alpha = e^{j\frac{2\pi}{3}}$, the following relations can be written:

For zero-sequence phasors,

$$\mathbf{V}_{b0} = \mathbf{V}_{a0}; \quad \mathbf{V}_{c0} = \mathbf{V}_{a0} \qquad 2.2\text{-a}$$

For positive sequence phasors,

$$\mathbf{V}_{bp} = \alpha^2 \mathbf{V}_{ap}; \quad \mathbf{V}_{cp} = \alpha \mathbf{V}_{ap} \qquad 2.2\text{-b}$$

For negative-sequence phasors,

$$\mathbf{V}_{bn} = \alpha \mathbf{V}_{an}; \quad \mathbf{V}_{cn} = \alpha^2 \mathbf{V}_{an} \qquad 2.2\text{-c}$$

Substituting these relations in Equation 2.1,

$$\mathbf{V}_a = \mathbf{V}_{a0} + \mathbf{V}_{ap} + \mathbf{V}_{an} \quad 2.3$$

$$\mathbf{V}_b = \mathbf{V}_{a0} + \alpha^2\mathbf{V}_{ap} + \alpha\mathbf{V}_{an} \quad 2.4$$

$$\mathbf{V}_c = \mathbf{V}_{a0} + \alpha\mathbf{V}_{ap} + \alpha^2\mathbf{V}_{an} \quad 2.5$$

Comparing Equations 2.3-2.5 with 2.1, the constant coefficients required to express a set of three phasors in terms of their symmetrical components are 1, 1, 1 for the zero-sequence components, 1, α^2, α for the positive sequence components, and 1, α, α^2 for the negative sequence components.

The three symmetrical component phasors $\mathbf{V}_{a0}, \mathbf{V}_{ap}, \mathbf{V}_{an}$ can be expressed in terms of the original phasors $\mathbf{V}_a$, $\mathbf{V}_b$ and $\mathbf{V}_c$ by a solution of the simultaneous linear phasor Equations 2.3, 2.4, and 2.5.

Adding Equations 2.3 and 2.4, and remembering that $(1+\alpha+\alpha^2)=0$, therefore

$$\mathbf{V}_{a0} = \frac{1}{3}(\mathbf{V}_a + \mathbf{V}_b + \mathbf{V}_c) \quad 2.6$$

Multiplying Equations 2.3, 2.4, and 2.5 by 1, α, α^2 respectively, and adding, therefore

$$\mathbf{V}_{ap} = \frac{1}{3}(\mathbf{V}_a + \alpha\mathbf{V}_b + \alpha^2\mathbf{V}_c) \quad 2.7$$

And multiplying Equations 2.3, 2.4, and 2.5 by 1, α^2, α respectively and adding, therefore

$$\mathbf{V}_{an} = \frac{1}{3}(\mathbf{V}_a + \alpha^2\mathbf{V}_b + \alpha\mathbf{V}_c) \quad 2.8$$

So the zero-sequence, the positive sequence, and the negative sequence values can be found from 2.2a, 2.2b, and 2.2c by using Equations 2.6, 2.7, and 2.8. By using Equations 2.6, 2.7 and 2.8 a *transformation matrix* can be introduced as

$$\begin{bmatrix} \mathbf{V}_{a0} \\ \mathbf{V}_{ap} \\ \mathbf{V}_{an} \end{bmatrix} = \frac{1}{3}\mathbf{A}\begin{bmatrix} \mathbf{V}_{a} \\ \mathbf{V}_{b} \\ \mathbf{V}_{c} \end{bmatrix} \tag{2.9}$$

where $\mathbf{A}$, the transformation matrix, is

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & \alpha & \alpha^2 \\ 1 & \alpha^2 & \alpha \end{bmatrix} \tag{2.10}$$

and $\mathbf{A}$ has the property that

$$\mathbf{A}^{-1} = \frac{1}{3}\mathbf{A}^{*} \tag{2.11}$$

$\mathbf{A}^{*}$ is complex conjugate of matrix $\mathbf{A}$.

2.2.3 Instantaneous and Spiral Symmetrical Components Methods

The symmetrical component method proposed by Fortescue uses the phasor representation of variables, and thus its analysis is limited to the steady state. In order to extent the method to transient analysis of three phase machines under asymmetrical operation Bekku [54] proposed the instantaneous symmetrical method. In the instantaneous symmetrical method positive and negative sequence components are conjugate of each other and are not independent [55, 65, 68]. This method is not compatible with Fortescue's symmetrical components, and method did not succeed in analyzing transient phenomena of AC machines.

To overcome this problem, the spiral vector method proposed by Yamamura [55] can be used for the transient analysis of AC machines. The spiral vector symmetrical components are identical to Fortescue's symmetrical components, and two methods are compatible. The symmetrical components are independent of each other. In the spiral vector symmetrical components method, the transformation matrix is the same as Fortescue's. More information is given in Appendix A for instantaneous and spiral vector symmetrical components as well as spiral vector.

2.3 The Power Decomposition Technique Computing Electrical Components in Unbalanced and Nonsinusiodal Conditions

A different method to derive the positive and negative sequence components of arbitrary three-phase signals in the time domain has been proposed by Huang [1] by describing their relevant powers in two-dimension ordinates. The main contribution of the method, namely *The Power Decomposition Technique* (PDT), is to explore power distribution of the positive and negative sequences in two-dimension based on a concept of energy conservation theorems in order to complete the decomposition procedure in two-dimensions.

2.3.1 Description of Power Decomposition Technique Method

An arbitrary three phase signal can be divided into positive, negative and zero sequence components using time phasors described in the conventional theory of symmetrical component [31]. The Power Decomposition Technique (PDT) method provides an alternate and direct way to get these results. In this algorithm, these symmetrical components can be obtained directly in the time-domain by separating positive and negative sequence powers with quantities expressed in a two-dimensional coordinate system.

2.3.2 Mathematical Description

Two sets of three-phase input signals, current and voltage signals are used in this analysis procedure. Assuming an arbitrary three-phase current signals, namely $i_a(t)$, $i_b(t)$, and $i_c(t)$, reference voltage signals $u_a(t)$, $u_b(t)$, and $u_c(t)$ are adopted which are balanced sinusoidal waveforms with a positive sequence rotating frame. The three-phase input currents *abc* can be transformed from three-phase to two-phase quantities by

$$\begin{bmatrix} i_\alpha(t) \\ i_\beta(t) \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_a(t) \\ i_b(t) \\ i_c(t) \end{bmatrix} \qquad 2.12$$

where the α and β axes are orthogonal coordinates. For the reference voltages

$$\begin{bmatrix} u_\alpha(t) \\ u_\beta(t) \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} u_a(t) \\ u_b(t) \\ u_c(t) \end{bmatrix} \qquad 2.13$$

The reference voltages u_α and u_β in 2-dimensions may be taken as

$$u_\alpha(t) = U\cos(\omega t) \quad \text{and} \quad u_\beta = U\sin(\omega t) \qquad 2.14$$

The relevant powers in two-dimensions can be expressed by the *Instantaneous Reactive Power theory* [70] which gives

$$\begin{bmatrix} p_\alpha(t) \\ p_\beta(t) \end{bmatrix} = \begin{bmatrix} u_\alpha(t) & 0 \\ 0 & u_\beta(t) \end{bmatrix} \begin{bmatrix} i_\alpha(t) \\ i_\beta(t) \end{bmatrix} \qquad 2.15$$

$$\begin{bmatrix} q_\alpha(t) \\ q_\beta(t) \end{bmatrix} = \begin{bmatrix} -u_\beta(t) & 0 \\ 0 & u_\alpha(t) \end{bmatrix} \begin{bmatrix} i_\alpha(t) \\ i_\beta(t) \end{bmatrix} \qquad 2.16$$

In Equation 2.15 p_α and p_β represent instantaneous *real powers* associated with the α and β axes currents respectively. Obviously, the relationship between the power in two-phase and three-phase system can be satisfied by

$$p(t) = p_\alpha(t) + p_\beta(t) = p_a(t) + p_b(t) + p_c(t) \qquad 2.17$$

where *p(t)* is total real instantaneous power flowing from the supply to the load in the three-phase circuit and has units of watts.

However, for Equation 2.16 the instantaneous powers q_α and q_β are not real powers, because the voltages and currents multiplied together are taken from different α, β

components. They are referred to as *imaginary powers*.

The average value of these powers given in above equations can be derived using the following equations

$$\begin{aligned}
\overline{p_\alpha(t)} &= P_\alpha = \frac{1}{T_s}\int_0^{T_s} p_\alpha(t)dt \\
\overline{p_\beta(t)} &= P_\beta = \frac{1}{T_s}\int_0^{T_s} p_\beta(t)dt \\
\overline{q_\alpha(t)} &= Q_\alpha = \frac{1}{T_s}\int_0^{T_s} q_\alpha(t)dt \\
\overline{q_\beta(t)} &= Q_\beta = \frac{1}{T_s}\int_0^{T_s} q_\beta(t)dt
\end{aligned} \qquad 2.18$$

where T_s is the fundamental period; *P* and *Q* are called active and reactive powers respectively. P_α, P_β, Q_α and Q_β are constant in steady state. If Equation 2.18 integration is over exact number of cycles, harmonics will be eliminated. Note that Equations 2.15, 2.16 and 2.18 are similar to complex DFT.

These powers P_α, P_β, Q_α and Q_β can be separated into positive sequence powers related to the positive sequence voltages and currents, and negative sequence powers related to the negative sequence voltages and currents in the appropriate axes. Then the individual powers related to positive and negative sequence currents can be obtained from

$$\begin{aligned}
P^i_{\alpha p} &= P_\alpha + P_\beta = P^i_{\beta p} \\
Q^i_{\alpha p} &= Q_\alpha + Q_\beta = Q^i_{\beta p}
\end{aligned} \qquad 2.19$$

$$\begin{aligned}
P^i_{\alpha n} &= P_\alpha - P_\beta = -P^i_{\beta n} \\
Q^i_{\alpha n} &= Q_\alpha - Q_\beta = -Q^i_{\beta n}
\end{aligned} \qquad 2.20$$

where *p* and *n* denote positive and negative sequence powers respectively.

The positive sequences currents $i_{\alpha p}$ and $i_{\beta p}$ are obtained as

$$\begin{bmatrix} i_{\alpha p} \\ i_{\beta p} \end{bmatrix} = \frac{1}{u_\alpha^{\ 2} + u_\beta^{\ 2}} \begin{bmatrix} P^i_{\alpha p}.u_\alpha - Q^i_{\alpha p}.u_\beta \\ P^i_{\beta p}.u_\beta + Q^i_{\beta p}.u_\alpha \end{bmatrix} \qquad 2.21$$

and the negative sequence currents $i_{\alpha n}$ and $i_{\beta n}$ are obtained as

$$\begin{bmatrix} i_{\alpha n} \\ i_{\beta n} \end{bmatrix} = \frac{1}{u_\alpha^{\ 2} + u_\beta^{\ 2}} \begin{bmatrix} P^i_{\alpha n}.u_\alpha - Q^i_{\alpha n}.u_\beta \\ P^i_{\beta n}.u_\beta + Q^i_{\beta n}.u_\alpha \end{bmatrix} \qquad 2.22$$

where $u_\alpha^{\ 2} + u_\beta^{\ 2} = U^2$, and U is the peak reference voltage and for convenience may be taken as unity (U = *1)*. Equations 2.21 and 2.22 generate instantaneous $i_{\alpha p}$, $i_{\beta p}$, $i_{\alpha n}$, and $i_{\beta n}$ currents from the constant P_α, P_β, Q_α and Q_β values and instantaneous sinusoidal u_α and u_β.

Positive and negative sequence currents are then given by phasors:

$$\mathbf{I}_p = P^i_{\alpha p} + jQ^i_{\alpha p} \qquad 2.23$$

$$\mathbf{I}_n = P^i_{\alpha n} + jQ^i_{\alpha n} \qquad 2.24$$

The same procedure can be used to obtain negative and positive sequence voltages, with voltage measurements $v_a(t)$, $v_b(t)$, and $v_c(t)$ replacing the current in Equation 2.12.

2.4 Application of Symmetrical Components to the Induction Motor

An important application of the theory of symmetrical components is the analysis of induction motor operation under unbalanced conditions. Operation under different conditions, and induction motor equivalent circuits, will be discussed in this subsection. Before that, sequence impedances will be defined.

2.4.1 Sequence Impedances

It has been shown [49, 55] that in symmetrical linear networks the components of voltage and current of the different sequences do not react upon each other. When the voltage of a given sequence is applied to such a system, a definite current of the same sequence flows. The sequence impedances are simply the ratio of the sequence voltage to the sequence current. Special names have been given to these impedances, namely: the impedance to positive-sequence currents, the impedance to negative-sequence currents, and the impedance to zero-sequence currents. These are contracted to the simple expressions, *positive sequence impedance, negative sequence impedance,* and *zero sequence impedance*.

The impedances of symmetrical static networks are the same for the positive- and the negative-sequences but may be different for the zero-sequence. For rotating machines the impedances are in general different for the three sequences.

2.4.2 Operation of Symmetrical Induction Motor under Unbalanced Condition

Figure 2.2 shows a model for the induction motor that will be used. '*Balanced*' three phase windings are assumed for both stator and rotor. That is, all three phase windings designated as *as*, *bs*, and *cs* for stator windings are assumed to have the same number of turns, and the *bs* and *cs* windings are symmetrically displaced from the *as* winding by $\pm 120^o$. The rotor windings are similarly arranged. The subscript '*s*' and '*r*' are for stator and rotor respectively.

Let the self-inductance of the stator and rotor windings be respectively denoted by L_s and L_r. These can be divided into two parts:

$$L_s = L_{ls} + M \quad \text{and} \quad L_r = L_{lr} + M \qquad 2.25$$

Here L_{ls} and L_{lr} are the stator and rotor leakage inductances, and M is the maximum value of the mutual inductances between the stator and rotor phase windings. As shown in

Figure 2.2, if the spatial phase difference between phase *as* and *ar* is θ_r, then the mutual inductance between the two phases is

$$M\cos\theta_r \tag{2.26}$$

which assumes sinusoidal distribution of the magnetic field in the air gap.

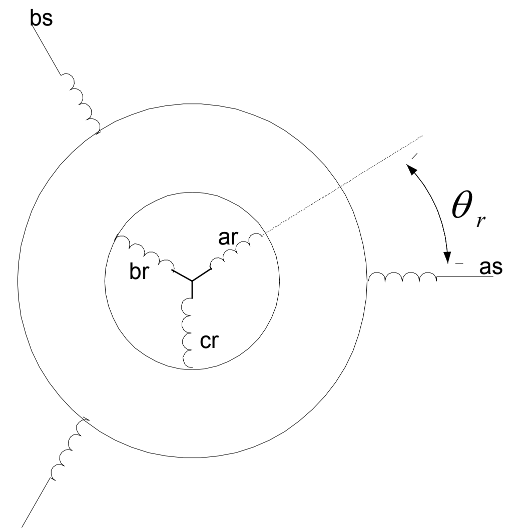


Figure 2.2: Induction motor model

The instantaneous voltage equations describing the stator and rotor circuits can be written conveniently in matrix form as

$$\begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} = r_s \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} + p \begin{bmatrix} \lambda_{as} \\ \lambda_{bs} \\ \lambda_{cs} \end{bmatrix} \tag{2.27}$$

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = r_r \begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix} + p \begin{bmatrix} \lambda_{ar} \\ \lambda_{br} \\ \lambda_{cr} \end{bmatrix} \tag{2.28}$$

where *p* represents the operator *d/dt*. The flux linkage λ_{as} can be divided into two parts

$$\lambda_{as} = L_{ls} i_{as} + \lambda_{gas} \tag{2.29}$$

The first term is the stator leakage flux linkage and λ_{gas} is the flux linkage of the revolving field in the air gap, also called the main flux. Similar definitions apply for the other stator and rotor phases. Equations 2.25 and 2.26 can be rewritten as

$$\begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} = (r_s + L_{ls} p) \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} + p \begin{bmatrix} \lambda_{gas} \\ \lambda_{gbs} \\ \lambda_{gcs} \end{bmatrix} \tag{2.30}$$

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = (r_r + L_{lr} p) \begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix} + p \begin{bmatrix} \lambda_{gar} \\ \lambda_{gbr} \\ \lambda_{gcr} \end{bmatrix} \tag{2.31}$$

The magnetic flux linkages for stator are given as

$$\begin{aligned} \begin{bmatrix} \lambda_{gas} \\ \lambda_{gbs} \\ \lambda_{gcs} \end{bmatrix} &= M \begin{bmatrix} 1 & \cos\left(\frac{2\pi}{3}\right) & \cos\left(-\frac{2\pi}{3}\right) \\ \cos\left(-\frac{2\pi}{3}\right) & 1 & \cos\left(\frac{2\pi}{3}\right) \\ \cos\left(\frac{2\pi}{3}\right) & \cos\left(-\frac{2\pi}{3}\right) & 1 \end{bmatrix} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} \\ &+ M \begin{bmatrix} \cos\theta_r & \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) \\ \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos\theta_r & \cos\left(\theta_r + \frac{2\pi}{3}\right) \\ \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos\theta_r \end{bmatrix} \begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix} \end{aligned} \tag{2.32}$$

The symmetrical components of the stator and rotor currents are

$$\begin{bmatrix} i_{s0} \\ i_{sp} \\ i_{sn} \end{bmatrix} = \frac{1}{3} \mathbf{A} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} i_{r0} \\ i_{rp} \\ i_{rn} \end{bmatrix} = \frac{1}{3} \mathbf{A} \begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix} \tag{2.33}$$

where $\mathbf{A}$ is given in Equation 2.10. Transforming the flux linkages into symmetrical components by using the transformation matrix $\mathbf{A}$ and $\mathbf{A}^{-1} = \frac{1}{3}\mathbf{A}^*$, we have

$$\begin{bmatrix} \lambda_{s0} \\ \lambda_{sp} \\ \lambda_{sn} \end{bmatrix} = \frac{1}{3}\mathbf{A}\begin{bmatrix} \lambda_{gas} \\ \lambda_{gbs} \\ \lambda_{gcs} \end{bmatrix} = \frac{1}{3}\mathbf{A}M\begin{bmatrix} 1 & \cos\left(\frac{2\pi}{3}\right) & \cos\left(-\frac{2\pi}{3}\right) \\ \cos\left(-\frac{2\pi}{3}\right) & 1 & \cos\left(\frac{2\pi}{3}\right) \\ \cos\left(\frac{2\pi}{3}\right) & \cos\left(-\frac{2\pi}{3}\right) & 1 \end{bmatrix}\mathbf{A}^*\begin{bmatrix} i_{s0} \\ i_{sp} \\ i_{sn} \end{bmatrix}$$
$$+\frac{1}{3}\mathbf{A}M\begin{bmatrix} \cos\theta_r & \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) \\ \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos\theta_r & \cos\left(\theta_r + \frac{2\pi}{3}\right) \\ \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos\theta_r \end{bmatrix}\mathbf{A}^*\begin{bmatrix} i_{r0} \\ i_{rp} \\ i_{rn} \end{bmatrix} \qquad 2.34$$

Performing the matrix multiplication, Equation 2.34 becomes

$$\begin{bmatrix} \lambda_{s0} \\ \lambda_{sp} \\ \lambda_{sn} \end{bmatrix} = \frac{3}{2}M\begin{bmatrix} 0 \\ i_{sp} + i_{rp}e^{j\theta_r} \\ i_{sn} + i_{rn}e^{-j\theta_r} \end{bmatrix} \qquad 2.35$$

The rotor flux linkages coming from the revolving gap flux are

$$\begin{bmatrix} \lambda_{gar} \\ \lambda_{gbr} \\ \lambda_{gcr} \end{bmatrix} = M\begin{bmatrix} 1 & \cos\left(\frac{2\pi}{3}\right) & \cos\left(-\frac{2\pi}{3}\right) \\ \cos\left(-\frac{2\pi}{3}\right) & 1 & \cos\left(\frac{2\pi}{3}\right) \\ \cos\left(\frac{2\pi}{3}\right) & \cos\left(-\frac{2\pi}{3}\right) & 1 \end{bmatrix}\begin{bmatrix} i_{ar} \\ i_{br} \\ i_{cr} \end{bmatrix}$$
$$+M\begin{bmatrix} \cos(-\theta_r) & \cos\left(-\theta_r + \frac{2\pi}{3}\right) & \cos\left(-\theta_r - \frac{2\pi}{3}\right) \\ \cos\left(-\theta_r - \frac{2\pi}{3}\right) & \cos(-\theta_r) & \cos\left(-\theta_r + \frac{2\pi}{3}\right) \\ \cos\left(-\theta_r + \frac{2\pi}{3}\right) & \cos\left(-\theta_r - \frac{2\pi}{3}\right) & \cos(-\theta_r) \end{bmatrix}\begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} \qquad 2.36$$

By applying the same transformation, flux linkages will be

$$\begin{bmatrix} \lambda_{r0} \\ \lambda_{rp} \\ \lambda_{rn} \end{bmatrix} = \frac{1}{3}\mathbf{A}\begin{bmatrix} \lambda_{gar} \\ \lambda_{gbr} \\ \lambda_{gcr} \end{bmatrix} = \frac{3}{2}M\begin{bmatrix} 0 \\ i_{rp} + i_{sp}e^{-j\theta_r} \\ i_{rn} + i_{sn}e^{j\theta_r} \end{bmatrix} \qquad 2.37$$

Equations 2.35 and 2.37 indicate that there are no zero-sequence component flux linkages, even though there are zero-sequence component currents.

Multiplying Equations 2.30 and 2.31 by $\frac{1}{3}\mathbf{A}$ from the left side, we get the following two equations:

$$\begin{bmatrix} v_{s0} \\ v_{sp} \\ v_{sn} \end{bmatrix} = (r_s + L_{ls}p)\begin{bmatrix} i_{s0} \\ i_{sp} \\ i_{sn} \end{bmatrix} + p\begin{bmatrix} \lambda_{s0} \\ \lambda_{sp} \\ \lambda_{sn} \end{bmatrix} \qquad 2.38$$

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = (r_r + L_{lr}p)\begin{bmatrix} i_{r0} \\ i_{rp} \\ i_{rn} \end{bmatrix} + p\begin{bmatrix} \lambda_{r0} \\ \lambda_{rp} \\ \lambda_{rn} \end{bmatrix} \qquad 2.39$$

Inserting Equations 2.35 and 2.37 into the Equations 2.38 and 2.39 respectively, we get following sequence components.

For zero-sequence components:

$$v_{s0} = (r_s + L_{ls}p)i_{s0} \qquad 2.40$$

$$0 = (r_r + L_{lr}p)i_{r0} \qquad 2.41$$

For positive sequence components:

$$v_{sp} = (r_s + L_{ls}p)i_{sp} + \frac{3}{2}Mp(i_{sp} + e^{j\theta_r}i_{rp}) \qquad 2.42$$

$$0 = (r_r + L_{lr}p)i_{rp} + \frac{3}{2}Mp(i_{rp} + e^{-j\theta_r}i_{sp}) \qquad 2.43$$

For negative sequence components:

$$v_{sn} = (r_s + L_{ls} p) i_{sn} + \frac{3}{2} Mp(i_{sn} + e^{-j\theta_r} i_{rn}) \qquad 2.44$$

$$0 = (r_r + L_{lr} p) i_{rn} + \frac{3}{2} Mp(i_{rn} + e^{j\theta_r} i_{sn}) \qquad 2.45$$

Note that for three wire three phase systems there is no zero sequence component. The positive sequence component equations can be simplified as follows:

In Equation 2.42 i_{rp} is positive sequence rotor current expressed in the rotor reference frame, but $i_{rp} e^{j\theta_r}$ is the positive sequence rotor current expressed in the stator reference frame,

$$i'_{rp} = i_{rp} e^{j\theta_r} \qquad 2.46$$

where θ_r is the rotor angle.

Then Equation 2.42 becomes

$$v_{sp} = r_s i_{sp} + (L_{ls} + \frac{3}{2} M) p i_{sp} + \frac{3}{2} M p i'_{rp} \qquad 2.47$$

By using i'_{rp} and $i_{rp} = e^{-j\theta_r} i'_{rp}$ Equation 2.43 becomes

$$0 = r_r i'_{rp} + (L_{ls} + \frac{3}{2} M)(p - j\omega_m) i'_{rp} + \frac{3}{2} M (p - j\omega_m) i_{sp} \qquad 2.48$$

Since $pe^{-j\theta_r} = -j\omega_m e^{-j\theta_r}$, where $\omega_m = \frac{d\theta_r}{dt}$, the rotor speed. These equations are transient state equations for positive sequence components.

For *the steady state*, p becomes $j\omega$ and Equations 2.47 and 2.48 gives the following equation for the positive sequence (see Appendix A.2)

$$\begin{bmatrix} \dot{V}_{sp} \\ 0 \end{bmatrix} = \begin{bmatrix} r_s + j(x_s + x_m) & jx_m \\ jx_m & \frac{r_r}{s_p} + j(x_r + x_m) \end{bmatrix} \begin{bmatrix} \dot{I}_{sp} \\ \dot{I}'_{rp} \end{bmatrix} \qquad 2.49$$

where $x_s = \omega L_{ls}$, $x_r = \omega L_{lr}$, $x_m = \frac{3}{2}\omega M$ and $s_p = \frac{\omega - \omega_m}{\omega}$ (s_p is positive sequence slip) and $\dot{V}_{sp} = V_m e^{j(\omega t + \varphi)}$. Figure 2.3 shows the corresponding equivalent circuit. I_{sp} and I_{rp} are stator and rotor positive sequence currents respectively.

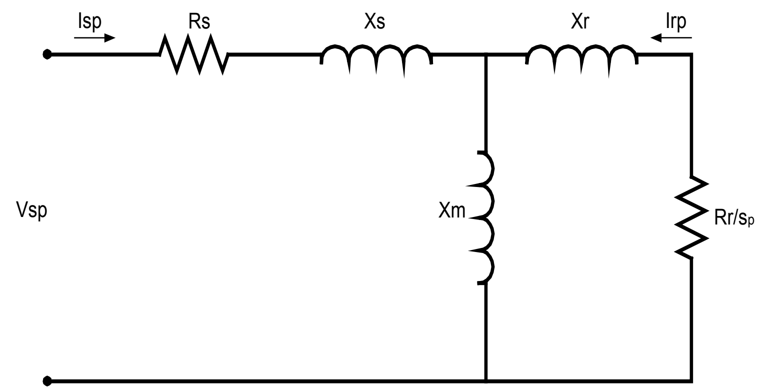


Figure 2.3: Steady state equivalent circuit of the induction motor for positive sequence components

Negative sequence components are treated in the same way as positive sequence components and i_{rn} is replaced by

$$i'_{rn} = e^{-j\theta_r} i_{rn} \tag{2.50}$$

The steady state equations for negative sequence becomes

$$\begin{bmatrix} \dot{V}_{sn} \\ 0 \end{bmatrix} = \begin{bmatrix} r_s + j(x_s + x_m) & jx_m \\ jx_m & \frac{r_r}{s_n} + j(x_r + x_m) \end{bmatrix} \begin{bmatrix} \dot{I}_{sn} \\ \dot{I}'_{rn} \end{bmatrix} \tag{2.51}$$

where $s_n = \frac{\omega + \omega_m}{\omega}$ (s_n is negative sequence slip). Figure 2.4 shows the corresponding equivalent circuit for negative sequence components.

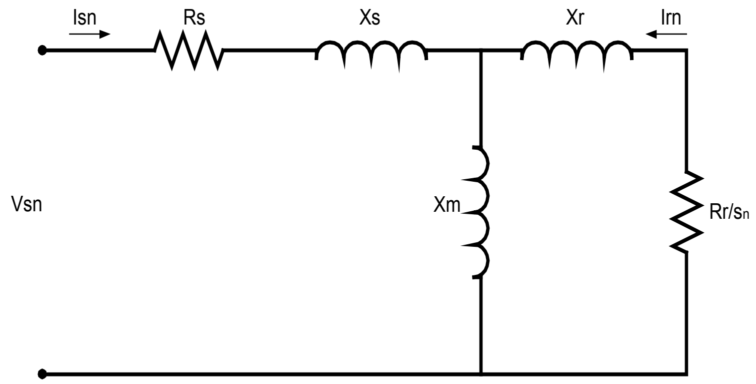


Figure 2.4: Steady state equivalent circuit of the induction motor for negative sequence components

The application of negative sequence voltage is in effect the same as interchanging the leads to two motor terminals. A field is set up which rotates at synchronous speed in the direction opposite to that set up by the positive sequence voltages. It follows then that characteristics for the negative sequence may be obtained by merely selecting the appropriate value of slip to use in conjunction with the positive sequence equivalent circuit. The equation relating the slips is

$$s_n = 2 - s_p \qquad 2.52$$

For very small values of s_p, s_n is very nearly equal to 2; and for the rotor locked, both s_p, and s_n are equal to unity.

As remarked previously, positive-sequence currents in the stator of a symmetrical machine produce a rotating field, which rotates in the same direction as the rotor. This field naturally produces only voltages of the same sequence at the terminals. The slip frequency currents in the rotor of induction machines produce only synchronously rotating fields in the same direction as the rotation of the rotor so that only positive-sequence voltages are produced on the terminals of the stator due to these currents. It is apparent then that under normal operating conditions with positive sequence voltages applied to the stator of rotating machines only positive-sequence currents are produced.

If negative-sequence voltages only are applied to the stator of induction machines, a synchronously rotating field is produced which rotates in a direction opposite to the rotation of the rotor. This field induces currents in the rotor, which in turn produces a synchronously rotating field in a direction opposite to that of the rotor. Thus all the currents and voltages in the stator would be of negative-sequence.

Because of the 120-degree space displacement of the windings, the zero-sequence currents, which are in phase with each other in the three phases, produce no flux in the air gap. Hence, these currents can produce only zero-sequence voltage drops (Equation 2.40).

When both positive and negative sequence voltages are applied simultaneously, as long as iron saturation is ignored and speed may be assumed constant due to a sufficiently large inertia, as it has been shown in above analysis, the current of the two sequences do not react upon each other. Thus they may be computed separately by the method discussed.

The sequence impedances of a symmetrical motor under unbalanced supply conditions are simply the ratios of the sequence voltages to the sequence currents.

i.e. $\mathbf{Z}_p = \dfrac{\mathbf{V}_{sp}}{\mathbf{I}_{sp}}$, $\mathbf{Z}_n = \dfrac{\mathbf{V}_{sn}}{\mathbf{I}_{sn}}$, and $\mathbf{Z}_0 = \dfrac{\mathbf{V}_{s0}}{\mathbf{I}_{s0}}$.

The results of this analysis can be used to separate supply negative sequence current from motor negative sequence current, which will be discussed in Chapter 3.

2.4.3 Operation of Asymmetrical Induction Motor under Unbalanced Condition

When both motor and applied voltages are unsymmetrical, complicated relations exist between positive and negative sequences. The voltages and currents of one sequence have influence upon those of another sequence. To demonstrate this, the same motor model will be used, except that the phase-*as*, stator phase resistance r_{as} will be assumed to differ from the other phases ($r_{as} \neq r_{bs} = r_{cs}$).

Let phase *as* have resistance $r_{as} = r_s + r_{ext}$, *bs* and *cs* have resistance r_s. The voltage equation describing the stator circuit in Equations 2.27 and 2.30 will be

$$\begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} = \begin{bmatrix} r_{as} & 0 & 0 \\ 0 & r_s & 0 \\ 0 & 0 & r_s \end{bmatrix} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} + p \begin{bmatrix} \lambda_{as} \\ \lambda_{bs} \\ \lambda_{cs} \end{bmatrix} \qquad 2.53$$

$$\begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} = \begin{bmatrix} r_{as} + L_{ls} p & 0 & 0 \\ 0 & r_s + L_{ls} p & 0 \\ 0 & 0 & r_s + L_{ls} p \end{bmatrix} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} + p \begin{bmatrix} \lambda_{gas} \\ \lambda_{gbs} \\ \lambda_{gcs} \end{bmatrix} \qquad 2.54$$

The magnetic flux linkages for stator and rotor will be the same as in Equations 2.35 and 2.37.

Multiplying Equation 2.54 by $\frac{1}{3}\mathbf{A}$ from the left side, stator voltages will be

$$\begin{bmatrix} v_{s0} \\ v_{sp} \\ v_{sn} \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 3(r_s + L_{ls} p) + r_{ext} & r_{ext} & r_{ext} \\ r_{ext} & 3(r_s + L_{ls} p) + r_{ext} & r_{ext} \\ r_{ext} & r_{ext} & 3(r_s + L_{ls} p) + r_{ext} \end{bmatrix} \begin{bmatrix} i_{s0} \\ i_{sp} \\ i_{sn} \end{bmatrix} + p \begin{bmatrix} \lambda_{s0} \\ \lambda_{sp} \\ \lambda_{sn} \end{bmatrix} \qquad 2.55$$

It can be shown that when r_{ext} is zero, Equation 2.55 will be the same as Equation 2.38.

By following the same procedure as before, the steady state equations for positive and negative sequences become

$$\begin{bmatrix} \dot{V}_{sp} \\ 0 \end{bmatrix} = \begin{bmatrix} r_s + \frac{r_{ext}}{3} + j(x_s + x_m) & \frac{r_{ext}}{3} & jx_m \\ jx_m & 0 & \frac{r_r}{s_p} + j(x_r + x_m) \end{bmatrix} \begin{bmatrix} \dot{I}_{sp} \\ \dot{I}_{sn} \\ \dot{I}'_{rp} \end{bmatrix} \qquad 2.56$$

$$\begin{bmatrix} \dot{V}_{sn} \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{r_{ext}}{3} & r_s + \frac{r_{ext}}{3} + j(x_s + x_m) & jx_m \\ 0 & jx_m & \frac{r_r}{s_n} + j(x_r + x_m) \end{bmatrix} \begin{bmatrix} \dot{I}_{sp} \\ \dot{I}_{sn} \\ \dot{I}'_{rn} \end{bmatrix} \qquad 2.57$$

Instead of Equations 2.49 and 2.51, Equations 2.56 and 2.57 show that for asymmetrical motor positive and negative sequences are no longer independent of each other, since there is an interaction between voltages and currents of different phase sequences. The voltage of one phase sequence is related not only to its own phase sequence current but also to the opposite phase sequence current as well as the imbalance of the motor.

2.5 Conclusion

Since the presentation of Fortescue's 1918 paper, work with symmetrical components has been directed principally toward the application of the method to the solution of problems commonly encountered in commercial systems, the most important of these being the determination of system currents and voltages under unbalanced fault conditions. This work has been much facilitated by the concepts of sequence networks and the use of equivalent circuits connecting these networks to represent fault conditions.

Knowledge of the method of symmetrical components is essential for an adequate understanding of application problems involving the magnitude and phase relation of voltages and currents under unbalanced conditions.

An important application of the theory of symmetrical components is the analysis of induction motor operation under unbalanced conditions. Chapter 3 discusses the application of symmetrical components to stator fault detection.

Chapter 3

The Algorithm for Stator Fault Diagnosis

3.1 Introduction

This chapter is concerned with the algorithm developed in this research for stator fault diagnosis. The symmetrical components theory of Fortescue [31] is a valuable tool to analyze and solve problems involving imbalance faults. The theoretical background for the symmetrical components and their application to induction motors was presented in the previous chapter.

Induction motor stator winding faults can be inferred from observation of the unbalance in motor phase currents. Two problems make this difficult:

- the effects of supply voltage unbalance
- the non-ideal behavior of the motor with respect to variations in load, voltage, unbalance, and temperature

Methods of eliminating these problems are described.

Unbalance in the three-phase voltage supply, as explained in Chapter 2, causes unbalanced (negative sequence) currents in a healthy motor. To distinguish whether negative sequence currents arise from unbalance in the supply or from a fault in the motor, it is necessary to measure the supply voltage unbalance and separate out the effect of supply unbalance on the current. Motor terminal voltage and current must both be sampled. An efficient and precise method of calculation is the *Power Decomposition Technique* (PDT). The analysis is independent of effects from supply harmonics provided that signals are sampled at a frequency at least twice that of significant harmonics. A software technique to eliminate the effects of non-simultaneous data sampling is also described.

Elimination of the effects of motor iron saturation, winding unbalance, rotor static eccentricity, and other non-linear effects involves characterization of these behaviours with respect to variations in load, voltage, unbalance, and temperature. To eliminate these effects semi-empirical formulae are introduced. To obtain parameters for these formulae a *Recursive Least Squares Identification* technique is used. The effect of large thermal changes in the resistive component of the motor negative sequence impedance $\mathbf{Z}_n$ may be greatly reduced by using a calculation technique which requires knowledge of only the negative sequence reactance X_n, which is essentially independent of temperature and slip variation.

3.2 The Stator Fault Detection Principle

Unbalance in three-phase systems may be measured in terms of positive and negative sequence components in the supply voltage and motor current. One of the approaches to detecting stator faults is to measure the negative sequence current in the unbalance motor phase current. However, the negative sequence current can arise from both supply unbalance and deterioration in the motor. It has been shown that in symmetrical systems the sequence components of voltage and current of the different systems do not react upon each other. The sequence impedances are simply the ratio of the sequence voltage to the sequence current. The negative sequence impedance of motor can be used to separate the negative sequence current $\mathbf{I}_n$ into a negative sequence current $\mathbf{I}_{sn}$ due to supply negative sequence voltage $\mathbf{V}_{sn}$ and a negative sequence current $\mathbf{I}_{mn}$ due to stator impedance unbalance.

Several limitations of stator fault diagnosis by monitoring negative sequence current have to be considered very carefully: Due to manufacture limitation initially all motors have some degree of unbalance. As it has been shown in Section 2.4.3, motor positive and negative sequences can be interdependent. This will introduce a coupling between positive and negative sequence equivalent circuits. The voltage of one phase sequence will be affected not only by its own phase current but also by the opposite phase sequence current as well as the unbalanced degree of the motor. Changes in load will modify negative sequence current because of this inherent asymmetry in the motor, varying iron saturation, and skin effect. Supply unbalance orientation (i.e. whether the maximum voltage is applied to the *as*, *bs*, or *cs* phase, or in combination) and variation from below to above nominal supply voltage ($-\mathbf{V}_{sn}$ to $+\mathbf{V}_{sn}$) causes variation in negative sequence impedance.

Non-identical gains of the current and voltage sensors in the measurement unit will also have some effect on measurement. These non-ideal and non-linear motor characteristic effects have to be removed from the measured negative sequence current before making any decision.

In real world, these effects can be eliminated by considering theoretical and experimental results together. The motor can be characterized by using semi-empirical formulae for load, voltage, and supply unbalance variation to get residual negative sequence currents for a healthy motor under different operating conditions.

3.3 The PDT Method for Measuring Negative Sequences

To measure the negative sequence components by the PDT method, 3-phase instantaneous current signals i_a, i_b, and i_c, and 3-phase line or phase voltages v_a, v_b, and v_c are both sampled simultaneously. A balanced sinusoidal positive sequence set of reference voltages u_a, u_b, and u_c is also used.

3.3.1 Measurement of Sequence Currents

The 3-phase sampled input currents are first transformed from 3- to 2-phase quantities by

$$\begin{bmatrix} i_\alpha(k) \\ i_\beta(k) \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} i_a(k) \\ i_b(k) \\ i_c(k) \end{bmatrix} \tag{3.1}$$

where the α and β suffices refer to 2-axis orthogonal coordinates fixed relative to the stator, and $k = 0,1,....,N_s - 1$. N_s is number of samples taken per supply cycle. For the reference voltages

$$\begin{bmatrix} u_\alpha(k) \\ u_\beta(k) \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} u_a(k) \\ u_b(k) \\ u_c(k) \end{bmatrix} \tag{3.2}$$

The reference voltages u_α and u_β in 2-dimensions may be taken from a look-up table as

$$u_\alpha(k) = \cos(2\pi \frac{k}{N_s} + \phi_{ref}) \text{ and } u_\beta(k) = \sin(2\pi \frac{k}{N_s} + \phi_{ref}) \tag{3.3}$$

ϕ_{ref} is taken to be zero. More information will be given in Section 3.4 concerning setting ϕ_{ref} to zero.

Instantaneous real power and imaginary powers *p(k)* and *q(k)* are calculated using

$$\begin{bmatrix} p_\alpha(k) \\ p_\beta(k) \end{bmatrix} = \begin{bmatrix} u_\alpha(k) & 0 \\ 0 & u_\beta(k) \end{bmatrix}\begin{bmatrix} i_\alpha(k) \\ i_\beta(k) \end{bmatrix} \tag{3.4}$$

$$\begin{bmatrix} q_\alpha(k) \\ q_\beta(k) \end{bmatrix} = \begin{bmatrix} -u_\beta(k) & 0 \\ 0 & u_\alpha(k) \end{bmatrix}\begin{bmatrix} i_\alpha(k) \\ i_\beta(k) \end{bmatrix} \tag{3.5}$$

The average value of these powers is then calculated over an integral number of supply cycles using the equations

$$\overline{p_\alpha(k)} = P_\alpha = \frac{1}{N_s}\sum_{k=0}^{Ns-1} p_\alpha(k)$$
$$\overline{p_\beta(k)} = P_\beta = \frac{1}{N_s}\sum_{k=0}^{Ns-1} p_\beta(k)$$
$$\overline{q_\alpha(k)} = Q_\alpha = \frac{1}{N_s}\sum_{k=0}^{Ns-1} q_\alpha(k)$$
$$\overline{q_\beta(k)} = Q_\beta = \frac{1}{N_s}\sum_{k=0}^{Ns-1} q_\beta(k) \qquad 3.6$$

P and Q are called active power and reactive power respectively.

These powers $P_\alpha, P_\beta, Q_\alpha, Q_\beta$ can be divided into positive sequence power related to the positive sequence currents, and negative sequence power related to negative sequence currents on the appropriate axes.

For the positive sequence power in 2 phases

$$P^i_{\alpha p} = P_\alpha + P_\beta = P^i_{\beta p}$$
$$Q^i_{\alpha p} = Q_\alpha + Q_\beta = Q^i_{\beta p} \qquad 3.7$$

whilst for the negative sequence power

$$P^i_{\alpha n} = P_\alpha - P_\beta = -P^i_{\beta n}$$
$$Q^i_{\alpha n} = Q_\alpha - Q_\beta = -Q^i_{\beta n} \qquad 3.8$$

Subscripts 'p' and 'n' denote positive and negative sequence respectively. By using result of Equations 3.7 and 3.8 the positive and negative sequence phasor components can be written in complex notation $(x + jy)$ as follows

$$\mathbf{I}_n = P^i_{\alpha n} + jQ^i_{\alpha n} = I_{nx} + jI_{ny}$$
$$\mathbf{I}_p = P^i_{\alpha p} + jQ^i_{\alpha p} = I_{px} + jI_{py} \qquad 3.9$$

3.3.2 Measurement of Sequence Voltages

Simultaneously with the current analysis exactly the same analysis is performed with supply voltage samples. The measured 3-phase voltage signals v_a, v_b, v_c, replace the current signals i_a, i_b, i_c, in order to calculate the positive and negative supply voltage sequence components**.** Voltage *P* and *Q* components $P^v_{\alpha p}$, $Q^v_{\alpha p}$**,** $P^v_{\alpha n}$ **,** $Q^v_{\alpha n}$ are formed like as in Equations 3.7 and 3.8. The positive and negative sequence voltage phasor components can be written as follows

$$\begin{aligned}\mathbf{V}_{sn} &= P^v_{\alpha n} + jQ^v_{\alpha n} = V_{snx} + jV_{sny} \\ \mathbf{V}_{sp} &= P^v_{\alpha p} + jQ^v_{\alpha p} = V_{spx} + jV_{spy}\end{aligned} \qquad 3.10$$

The *v* superfix denotes *P, Q* values from the processing of the *voltages*.

3.3.3 Noise Reduction

In the physical world, unwanted random disturbances contaminate measurements. These disturbances are due to both the nature of the variable being measured and the fallibility of measuring devices. Those unwanted fluctuations in measured values can be minimized by using time-domain averaging which is called *exponential averaging*. The x- and y-component of the sequence components can be averaged separately as

$$I_{nx\_av} = I_{nx\_av} + \left(I_{nx} - I_{nx\_av}\right)\frac{T_s}{T_{av}} \qquad 3.11$$

$$I_{ny\_av} = I_{ny\_av} + \left(I_{ny} - I_{ny\_av}\right)\frac{T_s}{T_{av}} \qquad 3.12$$

T_s is the period of the supply, and T_{av} is the averaging time constant. The same averaging can be carried out for other sequence components $\mathbf{I}_p$, $\mathbf{V}_{sn}$, and $\mathbf{V}_{sp}$ as well. This is equivalent to putting the fluctuating measurements of I_{nx} (and I_{ny}) made every supply period T_s through a first order low pass filter of time constant T_{av}. Typically, T_{av} has a value in the range 0.2-to-0.4 seconds.

3.4 Signal Sampling

The pattern of signal sampling should be locked to the supply frequency, with sampling equally spaced over exactly an integral number of cycles. The sampling frequency should be at least twice the frequency of any significant harmonic frequencies in the supply voltage or current waveforms, to avoid aliassing effects.

Reference voltages u_α and u_β must have the same frequency as the supply, which is by Equation 3.3. It is advantageous to phase lock the reference voltage to the positive sequence component $\mathbf{V}_{sp}$ of the supply voltage, to keep the measured imaginary component V_{spy} of $\mathbf{V}_{sp}$ equal to zero. This is equivalent to setting $\phi_{ref} = 0$.

(a) ***Phase locked sampling***

A phase locked loop circuit can be used to lock the reference signal to the positive sequence voltage signal. The measured value of this signal may also be used to adjust the reference frequency ω to maintain phase locked conditions and ensure that ω remains locked to the supply frequency.

(b) ***Non phase locked sampling***

If cycles are sampled only from time to time, on a non-continuous, asynchronous basis, without a phase locked loop, so that phase lock is not maintained continuously, an alternative approach is possible. As it was explained in section 3.3.1, ϕ_{ref} is initially taken zero for reference signal in Equation 3.3. The phase angle ϕ_p of $\mathbf{V}_{sp}$ can be used to lock the reference signals to the positive sequence supply voltage. ϕ_p is given by

$$\phi_p = \text{atan}\left(\frac{V_{spy}}{V_{spx}}\right) \qquad 3.13$$

Then to take $\mathbf{V}_{sp}$ as reference phasor, Equations 3.9 and 3.10 are multiplied by

$\exp(-j\phi_p)$ to correct for the unlocked phase difference.

3.4.1 Supply Frequency Variation and Harmonic Effects

Supply frequency variation and sampling frequencies which are not integer multiples of the supply frequency (when f_{samp} / f_s result is not integer) have effect on the PDT, because of the requirement of sampling over a integral number of complete cycles. These effects can be minimized by applying a Hanning window to the sampled data. With Hanning windowing applied to the sampled data, there is full freedom for selecting the sampling frequency.

The PDT method is naturally free from effects from supply harmonics. However, fluctuation in current at the shaft rotation frequency f_r and its harmonics can effect the results. To eliminate these effects, instead of sampling over one cycle, sampling can be done over a number of cycles equal to the number of pole pairs, for calculation of sequence components.

3.4.2 Software Correction for Non-Simultaneous Sampling

Non-simultaneous signal sampling of a three-phase system introduces a phase difference between sampled voltages (or currents) which depends on the A/D conversion time and the switching (multiplexer) time between channels. When currents and voltages are being measured to determine their positive and negative sequence components, any sampling delay time t_{delay} will introduce errors in the measurement of sequence components. A delay angle of δ ($\delta = 2\pi f_s t_{delay}$) between samples (in terms of fundamental frequency phase angles), gives a negative sequence amplitude error of $\delta/\sqrt{3}$ times the positive sequence amplitude. This is 18mA for a 10A positive sequence current, with 10 μs sample delays. Target sensitivity for negative sequence current is about 10mA in a full load current of 10A, so that this error is unacceptable.

Because simultaneous sample and hold circuits are expensive, non-simultaneous, sequential sampling is preferable. Fortunately, the error can be eliminated by a software correction, which is derived in Appendix B. If $\bar{\mathbf{I}}_p$ and $\bar{\mathbf{I}}_n$ are uncompensated measured values from non-simultaneous sampling, then the corrected true sequence values $\mathbf{I}_p$ and $\mathbf{I}_n$ can be derived as follows

$$\mathbf{I}_n = \bar{\mathbf{I}}_n e^{-j\delta} - \frac{\delta}{\sqrt{3}} \bar{\mathbf{I}}_p e^{-j(\frac{\pi}{3}+2\delta)} \tag{3.14}$$

$$\mathbf{I}_p = \bar{\mathbf{I}}_p e^{-j\delta} + \frac{\delta}{\sqrt{3}} \bar{\mathbf{I}}_n e^{j(\frac{\pi}{3}-2\delta)} \tag{3.15}$$

This virtually eliminates errors from non-simultaneous sampling. The same procedure can be applied to positive and negative sequence voltages. Figures 3.1 and 3.2 show the error in negative sequence current and voltage at different operating point with different sampling delay time t_{delay} without correction. While t_{delay} is taken constant, the positive sequence current and voltage *rms* values are assumed varying between 4-to-10 A and 235-to-245 V respectively. As can be seen from the figures, errors are not constant but varying with positive sequence current (or voltage) level.

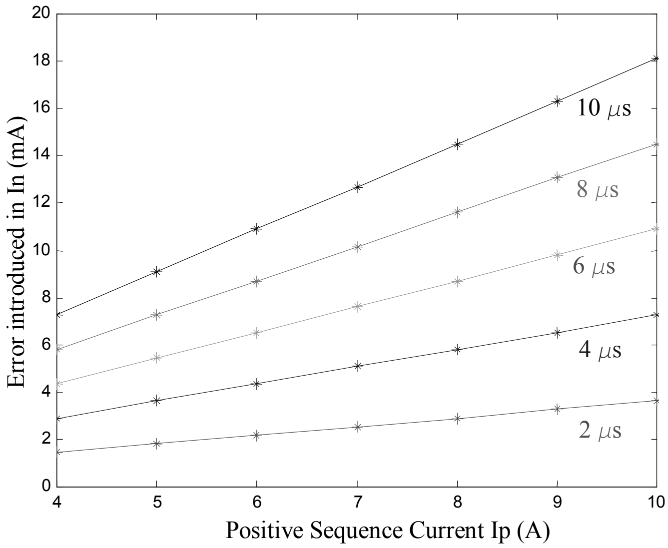


Figure 3.1: Error introduced in I_n for different t_{delay} at different I_p without compensation.

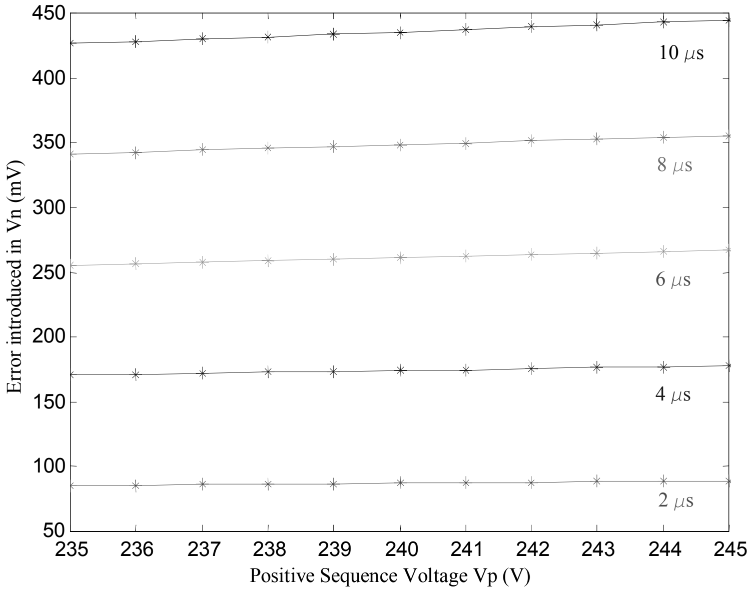


Figure 3.2: Error introduced in V_{sn} for different t_{delay} at different V_{sp} without compensation.

3.5 Analysis of Stator Condition

Values for positive sequence supply voltage $\mathbf{V}_{sp}$, positive sequence current $\mathbf{I}_p$, negative sequence supply voltage $\mathbf{V}_{sn}$ and current $\mathbf{I}_n$ are derived from sampled data by the PDT method above. In symmetrical systems the component of voltage and current of the different systems do not react upon each other. The sequence impedances are simply the ratio of the sequence voltages to the sequence currents. The negative sequence impedance $\mathbf{Z}_{hn}$ of the healthy motor can be defined from measured negative sequences when the motor is young. The negative sequence impedance $\mathbf{Z}_{hn}$ of the motor will be

$$\mathbf{Z}_{hn} = \frac{\mathbf{V}_{sn}}{\mathbf{I}_n} = \frac{V_{snx} + jV_{sny}}{I_{nx} + jI_{ny}} = R_{hn} + jX_{hn} \qquad 3.16$$

Subsequently at later stages in the motor history, in the presence of a stator fault, the observed negative sequence current $\mathbf{I}_n$ is considered as the sum of a negative sequence component $\mathbf{I}_{sn}$, caused by supply voltage unbalance $\mathbf{V}_{sn}$, plus a component $\mathbf{I}_{mn}$ arising from the motor stator winding fault i.e.

$$\mathbf{I}_n = \mathbf{I}_{sn} + \mathbf{I}_{mn} \qquad 3.17$$

Hence, the stator fault is associated with a motor fault current $\mathbf{I}_{mn}$ given by

$$\mathbf{I}_{mn} = \mathbf{I}_n - \mathbf{I}_{sn} = I_{mnx} + jI_{mny} \qquad 3.18$$

The magnitude and phase of this current component $\mathbf{I}_{mn}$ indicates the unbalance state of the stator due to winding deterioration. $\mathbf{I}_{sn}$ is calculated from the previously measured healthy impedance $\mathbf{Z}_{hn}$ as

$$\mathbf{I}_{sn} = \frac{\mathbf{V}_{sn}}{\mathbf{Z}_{hn}} = I_{snx} + jI_{sny} \qquad 3.19$$

This is expected supply negative sequence current produced by negative sequence voltage $\mathbf{V}_{sn}$.

3.6 Minimisation of Thermal Effects

Stator and rotor resistances are strongly dependent on temperature. Since $(r_r + jX_r) << jX_m$, jX_m may usually be omitted. From negative sequence equivalent circuit of induction motors in Chapter 2, we can deduce that negative sequence impedance can be written as

$$\mathbf{Z}_n = r_s + \frac{r_r}{2 - s_p} + j(X_s + X_r) = R_n + jX_n \qquad 3.20$$

Under operating condition, changes in motor temperature will directly alter the resistive component R_{hn} of the motor negative sequence impedance $\mathbf{Z}_{hn}$, and would modify the value of the fault current $\mathbf{I}_{mn}$. However, the fault detection process can be calculated from the negative sequence current in a way that is not affected by temperature. To lessen thermal effects on calculated value, $\mathbf{I}_{sn}$ can be derived using only the reactive part X_{hn} of $\mathbf{Z}_{hn}$.

The Figure 3.3 shows the phase relation between the negative sequence voltage $\mathbf{V}_{sn}$ and the expected negative sequence current $\mathbf{I}_{sn}$ for a healthy motor, with current lagging behind the voltage by phase angle θ_n. $\mathbf{V}_{sn}$ is seen to be given by

$$\mathbf{V}_{sn} = \mathbf{I}_{sn} R_{hn} + j\mathbf{I}_{sn} X_{hn} \qquad 3.21$$

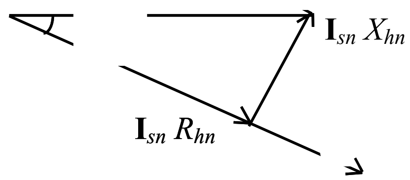


Figure 3.3: Vector diagram for negative sequence components

From Figure 3.3,

$$I_{sn} X_{hn} = V_{sn} \sin(\theta_n) \qquad 3.22$$

so that the expected negative sequence current I_{sn} caused by supply unbalance can be calculated from V_{sn} and the healthy negative sequence reactance X_{hn} as

$$I_{sn} = \frac{V_{sn} \sin\theta_n}{X_{hn}} \qquad 3.23$$

where $\theta_n = \text{atan}\left(\frac{X_n}{R_n}\right) = \text{atan}\left(\frac{V_{sny}}{V_{snx}}\right) - \text{atan}\left(\frac{I_{ny}}{I_{nx}}\right)$

The measured value is used for the phase angle θ_n of the actual measured current I_n. For a healthy motor, I_{sn} and I_n are identical, and the method does not require knowledge of R_n which varies with temperature. With a faulty stator, I_n may include a component I_{mn}

due to the stator fault. Hence we may estimate the fault current as

$$I_{mn} = I_n - I_{sn} = I_n - \frac{V_{sn}\ \sin\theta_n}{X_{hn}} \qquad 3.24$$

Calculated this way, an apparent fault current is not generated by thermal changes in R_n. X_n is determined by the mutual coupling of motor windings, and is a stable geometric quantity with little temperature variation. Comparing Equation 3.24 with 3.18, it is seen that I_{mn} in Equation 3.24 is the magnitude of the motor fault current. Phase information which, has been lost by not knowing the thermal effect on the value of R_n, is not need.

In a medium sized motor the negative sequence impedance can vary with slip between 2% to 5% due to load changes [56]. This variation is quite high and is directly related to resistive part of negative sequence impedance (Equation 3.20). By using the temperature independent method based on X_{hn} this effect is also minimized.

3.7 Non-linear Motor Effects

Unfortunately, if the fault current I_{mn} for a healthy motor is measured under different conditions of mains voltage and load, and with zero mains unbalance ($V_{sn} = 0$), not only is the measured negative sequence current I_n not zero, but it varies in a complex way with load and mains voltage. This indicates that even a healthy stator possesses unbalance dependent on operating conditions, and the variation in I_n may appear as an apparent fault current. The reasons are most probably due to variation in the degree of saturation of the iron in the motor, and to residual asymmetry in construction, such as the grain orientation of the steel, which may not have cylindrical symmetry around the periphery of the motor. Because of this residual asymmetry in the motor, sequence components will no longer be independent.

To eliminate load and voltage effects, we must determine how I_{mn} varies with load and voltage. This is called *characterization* of the motor.

A good fit with the observed variation in I_{mn} can be found by defining two functions I_{mnl} and I_{mnv} which express the dependence of I_{mn} on load and voltage, and making them functions of the positive sequence currents I_{px} and I_{py} respectively. I_{px} is in phase with the supply voltage, and approximates to the active load or torque current. I_{py} on the other hand, is in quadrature with supply voltage, and therefore represents magnetization current into the motor, and largely controls effects associated with iron saturation, which is directly influenced by supply voltage. Quadratic functions of I_{px} and I_{py} are found to give a good description of the healthy motor negative sequence currents caused by load and voltage variation (under conditions of zero supply unbalance) i.e.

$$I_{mnl} = \beta_0 + \beta_1 I_{px} + \beta_2 I_{px}^2 \qquad 3.25$$

$$I_{mnv} = \gamma_0 + \gamma_1 I_{py} + \gamma_2 I_{py}^2 \qquad 3.26$$

When combined as

$$I_{mnlv} = I_{mnl} + I_{mnv} \qquad 3.27$$

they describe how the negative sequence current of a healthy motor depends on load and voltage under conditions of zero supply unbalance. β_1, β_2, $(\beta_0 + \gamma_0)$, γ_1, and γ_2 are empirical constants which may be determined by fitting the observed behavior of I_{mn} as load and voltage are varied.

The stator fault current may then be expressed, in a form that remains independent of load and voltage, by

$$I_{mn}^{\bullet} = I_{mn} - (I_{mnl} + I_{mnv}) = I_{mn} - I_{mnlv} \qquad 3.28$$

The bracketed term in Equation 3.28 expresses the variation of I_{mn} with load and voltage for a healthy motor, so that $I_{mn}^{\bullet}$ represents any genuine fault current that has developed.

In order to find these parameters, measurements are taken whilst load and voltage are varied. To fit I_{mnl} and I_{mnv} simultaneously voltage and load may be varied together. From Equations 3.25, 3.26, 3.27,

$$I_{mnl} + I_{mnv} = (\beta_0 + \gamma_0) + \beta_1 I_{px} + \beta_2 I_{px}^2 + \gamma_1 I_{py} + \gamma_2 I_{py}^2 \quad 3.29$$

This may be written in terms of five α parameters as,

$$I_{mnlv} = \alpha_0 + \alpha_1 I_{px} + \alpha_2 I_{px}^2 + \alpha_3 I_{py} + \alpha_4 I_{py}^2 \quad 3.30$$

The stator fault current is then given by

$$I_{mn}^* = I_n - I_{sn} - I_{mnlv} \quad 3.31$$

where I_n is the total negative sequence current, and I_{sn} is given by Equation 3.23. Trends in I_{mn}^* can be used to show variation in the condition of the stator. Other terms like $I_{px} \cdot I_{py}$ are not included because of their negligible benefit.

3.8 Elimination of Dependence on Supply Voltage Unbalance

Experiments, in which supply unbalance is created, by unbalancing the 3 phase voltages, show that the negative sequence reactance X_{hn} varies with the degree of unbalance and load. Saturation in rotor leakage inductance is very strong in small/medium motors with closed rotor slots [83, 84, 85]. Healey et al [84] have shown that rotor leakage inductance can decrease by as much as a factor of 6 between minimum and maximum current density in the rotor. X_{hn} is summation of stator and rotor leakage reactances (Equation 3.20) which is therefore likely to vary with load (current) changes. It also varies with the orientation ϕ_n of the unbalance (i.e. on whether the voltage change is applied to the *a*-, *b*-, or *c*-phase or in combination). The orientation ϕ_n is measured in the PDT method as

$$\phi_n = \operatorname{atan}\left(\frac{V_{sny}}{V_{snx}}\right) \quad 3.32$$

To allow for this, the reactance X_{hn} in Equation 3.23 may be written as

$$X_{hn}^{-1} = \gamma_0 + \gamma_1 V_{sn} + \gamma_2 \sin 2\phi_n + \gamma_3 \cos 2\phi_n + \gamma_4 I_{px} + \gamma_5 I_{py}^2 \qquad 3.33$$

The justification for this form is as follows

- *1/γ₀* is the basic negative sequence reactance value X_{hn}
- the V_{sn} term allows for some slight dependence on V_{sn}
- the *sin* and *cos* terms allow for a geometrical static eccentricity or non-centering of the rotor in the airgap causing X_{hn} to show an elliptical variation with ϕ_n
- the I_{py} term represents the variation of X_{hn} with magnetization current due to main flux saturation
- the I_{px} term represents the variation of the X_{hn} with load change (rotor leakage inductance saturation due to closed rotor slots)

Equation 3.23 then becomes, in terms of generalized parameters γ

$$I_{sn} = \frac{V_{sn} \sin\theta_n}{X_{hn}} = V_{sn} \sin\theta_n [\gamma_0 + \gamma_1 V_{sn} + \gamma_2 \sin 2\phi_n + \gamma_3 \cos 2\phi_n + \gamma_4 I_{px} + \gamma_5 I_{py}^2] \quad 3.34$$

Substituting for I_{sn} in Equation 3.31 yields a final formula for the stator fault current I_{mn}^*:

$$\begin{aligned} I_{mn}^* = I_n - V_{sn} \sin\theta_n [\gamma_0 + \gamma_1 V_{sn} + \gamma_2 \sin 2\phi_n + \gamma_3 \cos 2\phi_n + \gamma_4 I_{px} + \gamma_5 I_{py}] \\ - (\alpha_0 + \alpha_1 I_{px} + \alpha_2 I_{px}^2 + \alpha_3 I_{py} + \alpha_4 I_{py}^2) \end{aligned} \qquad 3.35$$

The parameters γ can be obtained in the same way as parameters α, by varying the supply unbalance V_{sn} and orientation ϕ_n, at no-load and full-load when I_{mn}^* for the healthy motor remains zero.

3.9 Motor Characterization

Calculation of the fault current I^*_{mn} after characterization of the non-linear behavior of the motor by measurement of the parameters in Equation 3.35 allows a high sensitivity to be achieved for stator fault detection. The characterization must be made for variations in supply voltage, supply unbalance, and load. It is best carried out in two stages. Firstly, the motor is characterized for supply unbalance, described by the γ parameters. Secondly, the motor is characterized for load and voltage variation, described by the α parameters. It is important to perform the unbalance characterization first, to obtain the basic value for X_{hn} and its variation according to Equation 3.33.

3.9.1 Characterization for Supply Unbalance

Characterization for supply unbalance is carried out as follows. Using three single-phase autotransformers, the voltage of one phase is varied between 4% below to 2% above nominal to cause supply unbalance. The unbalance percentage is changed linearly over time whilst sampling and fitting is carried out on-line (or voltage and current data are saved into files). When the voltage variation of one phase is complete, the transformers are set to start variation of the next phase. Approximately 700 data points are enough for each phase. If the voltage is stepped automatically (e.g. by using a Variac driven by a stepper motor) only about 500 data points per phase are required. The variac may be turned slowly and continuously to effect the changes in unbalance whilst the characterization is carried out. The data may be collected over 2 minutes.

3.9.2 Characterization for Voltage and Load

Characterization is then performed for load and voltage variation, with a fully balanced supply (if a fully balanced supply is not possible, any supply unbalance can be subtracted off by supply unbalance characterization). Load and voltage may be varied simultaneously, or successively.

The characterization for load is carried out at rated voltage, and load is increased linearly from no-load to full-load, and approximately 2000 sampled voltage and current data points are required to fit on-line or save into a file for later processing. Then the (balanced) supply voltage is varied linearly, by means of a three phase Variac, from 10% below to 5% above rated voltage, and further data is collected. The two process takes about 2 minutes.

3.10 On-line Motor Characterization-Recursive Least Square Method

The Recursive Least Squares method can be used to find the characterization parameters, which describe the non-linear behavior of the motor with respect to variation of voltage, load, and supply unbalance. In recursive (also called on-line) identification methods, parameter estimates are computed recursively in real time. This means that an up to date estimated model of the process is available at all times.

The Recursive Least Squares Method requires only a modest amount of information to be stored, which does not depend on the number of data points used in fitting process. Details of method are given in Appendix C. The method is equivalent to obtaining a least square fit to the measured negative sequence current,

$$\begin{aligned} I_n = V_{sn} \sin\theta_n [\gamma_0 + \gamma_1 V_{sn} + \gamma_2 \sin 2\phi_n + \gamma_3 \cos 2\phi_n + \gamma_4 I_{px} + \gamma_5 I_{py}^2] \\ + (\alpha_0 + \alpha_1 I_{px} + \alpha_2 I_{px}^2 + \alpha_3 I_{py} + \alpha_4 I_{py}^2) \end{aligned}, \qquad 3.36$$

as V_{sn}, ϕ_n, I_{px}, and I_{py} are varied. Perfect fit will yield $I_{mn}^* = 0$. The actual fit achieved leaves $I_{mn}^* = residual\ current$ (Figure 6.8 (c) and 6.11 (b)).

3.11 Conclusion

The aim of the thesis is to use the Power Decomposition Technique to detect stator inter-turn short circuit faults by monitoring negative sequence current. The method is one of simplest but most powerful technique for stator fault detection. In the present thesis, the algorithm has been extended for all operating conditions, and the sensitivity of the method increased.

Supply unbalance is the major problem of using negative sequence current as a stator fault indicator. The supply unbalance effect can be eliminated by using healthy negative sequence impedance. By using the healthy negative sequence impedance, the negative sequence current can be separated into two parts: one due to supply unbalance, other due to motor degradation. But this research shows that the healthy negative sequence impedance is not a constant. This is mainly because of motor inherent asymmetry, main flux saturation, rotor leakage inductance saturation due to closed rotor slots, and geometrical static eccentricity or non-centering of the rotor in the air gap. This is a mainly different variation then one with slip changes. A semi-empirical formula had been proposed for negative sequence reactance to account for these dependencies.

Temperature effects and variation in the negative sequence resistance with slip change are minimized by using only negative sequence reactance to find supply negative sequence current.

A compensation method had been introduced for non-simultaneous sampling. With non-simultaneous sampling, the negative sequence components will be depending on positive sequence components and vice versa. This will introduce more complexity in the fault detection algorithm. By using the method introduced, effect of non-simultaneous signal sampling can be eliminated.

An on-line system identification technique, Recursive Least Squares Method, has been used to fit the semi-empirical formulae proposed for motor characterization.

Implementation of the stator fault diagnosis algorithm using different motors under different conditions will be given in Chapter 6.

Chapter 4

Modelling and Simulation of Induction Motors for Stator Fault Diagnosis

4.1 Introduction

This chapter presents an orthogonal axis models for three-phase induction motors having asymmetrical windings on the stator. Two models are presented here, the first one assumes that each motor phase has a different number of turns; and by assuming phase-*as* has two series windings, the second model uses the results of first model to transfer phase *as* to *qd* axis to apply short circuit in *qd* axis. Equivalent *q-d-0* equations for a stationary reference frame are derived, and new parameters have been defined from well known general motor parameters. The models developed have been simulated in SIMULINK and used to test analysis of the condition of the motor by the stator fault analysis algorithm.

The asymmetrical induction machine has been a subject of considerable interest. Brown and Butler [58] have utilised symmetrical component theory to establish a general method of analysis for operation of polyphase induction motors having asymmetrical primary connection. Jha and Murthy [57] have utilised rotating field concepts to develop a generalised theory of induction machine having asymmetrical windings on both stator and rotor.

The generalised theory of electrical machines incorporating orthogonal axis or *qd0* axis theory is generally accepted as the preferred approach to almost all types of transient and steady state phenomena [60]. The analysis of machines is greatly facilitated by the standard transformation to *qd0* axis. The same transformation process can be applied to machines in which there are phase unbalances [61].

Hence, it is useful to extend this approach to also incorporate problems encountered with asymmetrical induction motors. The aim of this chapter is to present a useful and easy method to simulate inter-turn winding faults for diagnostic purposes. The inter-turn fault can be simulated by disconnection of one or more turns making up of a stator phase winding [15, 59]. In this chapter, firstly an induction motor model that has unequal numbers of stator turns will be developed. Then using this model, a second model will be developed to simulate stator inter-turn short circuits. SIMULINK models and results of simulation will be given.

4.2 Modelling Of Induction Motors with Different Numbers of Stator Turn

The model for a symmetrical three-phase induction motor is well known [62, 65, 67]. To derive all equations for asymmetrical stator winding and rotor, the following assumptions have been made:

- each stator phase of motor has a different number of turns, but uniform spatial displacement is assumed
- saturation is not present

With the appropriate subscript *as, bs, cs, ar, br,* and *cr,* the voltage equations of the magnetically coupled stator and rotor circuits can be written as follows:

Stator voltage equations

$$
\begin{aligned}
v_{as} &= r_{as} i_{as} + \frac{d\lambda_{as}}{dt} \\
v_{bs} &= r_{bs} i_{bs} + \frac{d\lambda_{bs}}{dt} \\
v_{cs} &= r_{cs} i_{cs} + \frac{d\lambda_{cs}}{dt}
\end{aligned}
\qquad 4.1
$$

and rotor voltage equations

$$
\begin{aligned}
v_{ar} &= r_{ar} i_{ar} + \frac{d\lambda_{ar}}{dt} \\
v_{br} &= r_{br} i_{br} + \frac{d\lambda_{br}}{dt} \\
v_{cr} &= r_{cr} i_{cr} + \frac{d\lambda_{cr}}{dt}
\end{aligned}
\qquad 4.2
$$

By applying the stationary reference frame transformation to these equations yields the corresponding *qd0* equations. The transformation matrix $\mathbf{T}_{qd0}(\theta)$ and its inverse $\mathbf{T}_{qd0}(\theta)^{-1}$ are

$$
\mathbf{T}_{qd0}(\theta) = \frac{2}{3}\begin{bmatrix} \cos\theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin\theta & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \qquad 4.3a
$$

$$
\mathbf{T}_{qd0}(\theta)^{-1} = \begin{bmatrix} \cos\theta & \sin\theta & 1 \\ \cos\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) & 1 \end{bmatrix} \qquad 4.3b
$$

In stationary reference frame $\omega = 0$ and assuming q and *as* axis are aligned $\theta = 0$.

In matrix notation, the stator winding *abc* voltage equations can be expressed as

$$\mathbf{v}^s_{abc} = \mathbf{r}^s_{abc}\mathbf{i}^s_{abc} + p\lambda^s_{abc} \qquad 4.4$$

where $p = d/dt$.

Applying the transformation $\mathbf{T}_{qd0}(0)$, to the voltage, flux linkage and current, Equation 4.4 becomes

$$\mathbf{T}_{qd0}(0)\mathbf{v}^s_{abc} = \mathbf{T}_{qd0}(0)\mathbf{r}^s_{abc}\mathbf{i}^s_{abc} + \mathbf{T}_{qd0}(0)p\lambda^s_{abc}$$

$$\mathbf{v}^s_{qd0} = \mathbf{T}_{qd0}(0)\mathbf{r}^s_{abc}\mathbf{T}_{qd0}(0)^{-1}\mathbf{i}^s_{qd0} + \mathbf{T}_{qd0}(0)p\mathbf{T}_{qd0}(0)^{-1}\lambda^s_{qd0}$$

$$\mathbf{v}^s_{qd0} = \mathbf{r}^s_{qd0}\mathbf{i}^s_{qd0} + p\lambda^s_{qd0} \qquad 4.5$$

where $\mathbf{r}^s_{qd0} = \begin{bmatrix} r^s_{11} & r^s_{12} & r^s_{13} \\ r^s_{21} & r^s_{22} & r^s_{23} \\ r^s_{31} & r^s_{32} & r^s_{32} \end{bmatrix}$ and matrix elements are given in Appendix D.

Likewise the rotor quantities can be transformed into the same *qd* frame. In the same manner as we have done with the stator voltage equation, the following *qd0* voltage equation for the rotor windings can be obtained

$$\mathbf{v}^r_{abc} = \mathbf{r}^r_{abc}\mathbf{i}^r_{abc} + p\lambda^r_{abc}$$

$$\mathbf{v}^r_{qd0} = \mathbf{T}_{qd0}(-\theta_r)\mathbf{r}^r_{abc}\mathbf{T}_{qd0}(-\theta_r)^{-1}\mathbf{i}^r_{qd0} + \mathbf{T}_{qd0}(-\theta_r)p\mathbf{T}_{qd0}(-\theta_r)^{-1}\lambda^r_{qd0}$$

$$\mathbf{v}^r_{qd0} = \mathbf{r}^r_{qd0}\mathbf{i}^r_{qd0} - \omega_r \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\lambda^r_{qd0} + p\lambda^r_{qd0} \qquad 4.6$$

Assuming $r_{ar} = r_{br} = r_{cr} = r_r$, $\mathbf{r}^r_{qd0} = \begin{bmatrix} r_r & 0 & 0 \\ 0 & r_r & 0 \\ 0 & 0 & r_r \end{bmatrix}$ and θ_r is the rotor angle.

In matrix notation, the flux linkages of the stator and rotor windings, in terms of the winding inductances and the current may be written as

$$\begin{bmatrix} \lambda_{abc}^{s} \\ \lambda_{abc}^{r} \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{abc}^{ss} & \mathbf{L}_{abc}^{sr} \\ \mathbf{L}_{abc}^{rs} & \mathbf{L}_{abc}^{rr} \end{bmatrix} \begin{bmatrix} \mathbf{i}_{abc}^{s} \\ \mathbf{i}_{abc}^{r} \end{bmatrix} \quad 4.7$$

Where stator and rotor inductances are

$$\mathbf{L}_{abc}^{ss} = \begin{bmatrix} L_{asas} & L_{asbs} & L_{ascs} \\ L_{bsas} & L_{bsbs} & L_{bscs} \\ L_{csas} & L_{csbs} & L_{\mathrm{csc}\,s} \end{bmatrix} \qquad \mathbf{L}_{abc}^{rr} = \begin{bmatrix} L_{arar} & L_{arbr} & L_{arcr} \\ L_{brar} & L_{brbr} & L_{brcr} \\ L_{crar} & L_{crbr} & L_{crcr} \end{bmatrix} \quad 4.8$$

Because of symmetry, stator mutual inductances are $L_{asbs} = L_{bsas}$, $L_{ascs} = L_{csas}$ and $L_{bscs} = L_{csbs}$. Similarly rotor self and mutual inductances are $L_{arar} = L_{brbr} = L_{crcr}$ and $L_{arbr} = L_{arcr} = L_{brar} = L_{brcr} = L_{crar} = L_{crbr}$ respectively.

Those of the stator-to-rotor mutual inductances are dependent on the rotor angle (orientated with respect to stator), therefore

$$\mathbf{L}_{abc}^{sr} = \begin{bmatrix} L_{asar}\cos\theta_r & L_{asbr}\cos\left(\theta_r + \frac{2\pi}{3}\right) & L_{ascr}\cos\left(\theta_r - \frac{2\pi}{3}\right) \\ L_{bsar}\cos\left(\theta_r - \frac{2\pi}{3}\right) & L_{bsbr}\cos\theta_r & L_{bscr}\cos\left(\theta_r + \frac{2\pi}{3}\right) \\ L_{csar}\cos\left(\theta_r + \frac{2\pi}{3}\right) & L_{csbr}\cos\left(\theta_r - \frac{2\pi}{3}\right) & L_{\mathrm{csc}\,r}\cos\theta_r \end{bmatrix} \quad 4.9$$

and $\mathbf{L}_{abc}^{rs} = \left[\mathbf{L}_{abc}^{sr}\right]'$ where (') means transpose of matrix. The coefficients L_{asar}, L_{asbr}, L_{ascr}, L_{bsar}, L_{bsbr}, L_{bscr}, L_{csar}, L_{csbr}, and $L_{\mathrm{csc}\,r}$ are peak values of stator-to-rotor mutual inductances. Because of rotor symmetry $L_{asar} = L_{asbr} = L_{ascr}$, $L_{bsar} = L_{bsbr} = L_{\mathrm{csc}\,r}$, and $L_{csar} = L_{csbr} = L_{\mathrm{csc}\,r}$.

The stator *qd0* resistances are obtained by applying transformation $\mathbf{T}_{qd0}(0)\mathbf{r}_{abc}^{s}\mathbf{T}_{qd0}(0)^{-1}$ and result is given in Appendix D.

The stator *qd0* flux linkages are obtained by applying transformation matrix $\mathbf{T}_{qd0}(0)$ to the stator *abc* flux linkages in Equation 4.7, that is

$$\lambda_{qd0}^{s} = \mathbf{T}_{qd0}(0)\left(\mathbf{L}_{abc}^{ss}\mathbf{i}_{abc}^{s} + \mathbf{L}_{abc}^{sr}\mathbf{i}_{abc}^{r}\right) \qquad 4.10$$

Using the appropriate inverse transformations to replace the *abc* stator and rotor current by their corresponding *qd0* current, Equation 4.10 becomes

$$\lambda_{qd0}^{s} = \mathbf{T}_{qd0}(0)\mathbf{L}_{abc}^{ss}\mathbf{T}_{qd0}(0)^{-1}\mathbf{i}_{qd0}^{s} + \mathbf{T}_{qd0}(0)\mathbf{L}_{abc}^{sr}\mathbf{T}_{qd0}(-\theta_r)^{-1}\mathbf{i}_{qd0}^{r} \qquad 4.11$$

The transformations of $\mathbf{L}_{abc}^{ss}$ to $\mathbf{L}_{qd0}^{ss}$ and $\mathbf{L}_{abc}^{sr}$ to $\mathbf{L}_{qd0}^{sr}$ are given by

$$\mathbf{L}_{qd0}^{ss} = \begin{bmatrix} L_{11}^{ss} & L_{12}^{ss} & L_{13}^{ss} \\ L_{21}^{ss} & L_{22}^{ss} & L_{23}^{ss} \\ L_{31}^{ss} & L_{32}^{ss} & L_{33}^{ss} \end{bmatrix} \quad \text{and} \quad \mathbf{L}_{qd0}^{sr} = \begin{bmatrix} L_{11}^{sr} & L_{12}^{sr} & 0 \\ L_{21}^{sr} & L_{22}^{sr} & 0 \\ L_{31}^{sr} & L_{32}^{sr} & 0 \end{bmatrix} \qquad 4.12$$

and flux linkage can be simplified as

$$\lambda_{qd0}^{s} = \mathbf{L}_{qd0}^{ss}\mathbf{i}_{qd0}^{s} + \mathbf{L}_{qd0}^{sr}\mathbf{i}_{qd0}^{r} \qquad 4.13$$

Similarly, the *qd0* rotor flux linkages are given by

$$\lambda_{qd0}^{r} = \mathbf{T}_{qd0}(-\theta_r)\mathbf{L}_{abc}^{rs}\mathbf{T}_{qd0}(0)^{-1}\mathbf{i}_{qd0}^{s} + \mathbf{T}_{qd0}(-\theta_r)\mathbf{L}_{abc}^{rr}\mathbf{T}_{qd0}(-\theta_r)^{-1}\mathbf{i}_{qd0}^{r} \qquad 4.14$$

The resulting transformations can be simplified as

$$\mathbf{L}_{qd0}^{rr} = \begin{bmatrix} L_{11}^{rr} & 0 & 0 \\ 0 & L_{22}^{rr} & 0 \\ 0 & 0 & L_{33}^{rr} \end{bmatrix} \quad \mathbf{L}_{qd0}^{rs} = \begin{bmatrix} L_{11}^{sr} & L_{21}^{sr} & \frac{1}{2}L_{31}^{sr} \\ L_{12}^{sr} & L_{22}^{sr} & -\frac{1}{2}L_{32}^{sr} \\ 0 & 0 & 0 \end{bmatrix} \qquad 4.15$$

and flux linkage is

$$\lambda_{qd0}^{r} = \mathbf{L}_{qd0}^{rs}\mathbf{i}_{qd0}^{s} + \mathbf{L}_{qd0}^{rr}\mathbf{i}_{qd0}^{r} \qquad 4.16$$

Matrix elements are given in Appendix D.

Normally an induction machine is connected to a 3-phase supply by a three wire connection (i.e. neutral current does not flow) and for the induction motor with squirrel cage v_q^r , v_d^r and v_0^r are identically zero. So that for the squirrel cage induction machine and three-wire connection, the stator and rotor flux linkages in Equations 4.14 and 4.16 may be expressed compactly as follows

$$\begin{bmatrix} \lambda_q^s \\ \lambda_d^s \\ \lambda_q^r \\ \lambda_d^r \end{bmatrix} = \begin{bmatrix} L_{11}^{ss} & L_{12}^{ss} & L_{11}^{sr} & L_{12}^{sr} \\ L_{21}^{ss} & L_{22}^{ss} & L_{21}^{sr} & L_{22}^{sr} \\ L_{11}^{sr} & L_{12}^{sr} & L_{11}^{rr} & 0 \\ L_{21}^{sr} & L_{22}^{sr} & 0 & L_{22}^{rr} \end{bmatrix} \begin{bmatrix} i_q^s \\ i_d^s \\ i_q^r \\ i_d^r \end{bmatrix} \quad 4.17$$

4.2.2 Determination of Inductances

The mutual inductances between two windings (*x* and *y*) can be defined as [62]

$$L_{xy} = \mu_0 N_x N_y \left(\frac{rl}{g}\right)\frac{\pi}{4}\cos\alpha = N_x N_y A_g \cos\alpha \quad 4.18$$

N_x and N_y are effective number of turns of *x* and *y* windings respectively, α is angle displacement between the magnetic axes of the two windings, and A_g may be defined as machine constant dependent on airgap length *g*, the axial length *l* of the machine that flux is evenly distributed, radius *r* and permeability of free space μ_0.

Before we define inductances for an asymmetrical motor, we firstly define them for a symmetrical motor by using well known magnetic and leakage inductances.

The self-inductance of stator phase *as* can be obtained by simply setting $\alpha = 0$ and N_x and N_y equal to N_s

$$L_{am} = N_s^{\ 2} A_g \quad 4.19$$

The subscript *m* is used to denote the fact that this inductance is a magnetizing inductance. In general it is necessary to add a relatively small, but important, leakage term to L_{am}, to account for leakage flux. The leakage inductance is proportional to square of number of turns ($\propto N_s^{\,2}$) [63], and can be defined as follows

$$L_{ls} \cong N_s^{\,2} A_l \qquad 4.20$$

where A_l is a machine constant for leakage inductance.

Hence the total self-inductance of phase *as* can be expressed as

$$L_{as} = L_{ls} + L_{am} \qquad 4.21$$

and for a symmetric machine $L_{am} = L_{bm} = L_{cm}$.

This makes the self-inductances also equal. So it is therefore useful to define stator-magnetizing inductance

$$L_{ms} = N_s^{\,2} A_g \qquad 4.22$$

which is the magnetising inductance of one stator phase winding by a single-phase stator current. Generally, instead of stator magnetising inductance, magnetising inductance (which is take into account the effects of the other two stator phase windings) is given and is equal to

$$L_m = \frac{3}{2} L_{ms} \qquad 4.23$$

Inductances for asymmetrical machine can be defined by using these of symmetrical machine parameter as follows:

Machine constants can be defined in terms of magnetising inductance L_m and L_{ls} as

$$A_g = \frac{L_{ms}}{N_s^{\,2}} = \frac{2}{3}\frac{L_m}{N_s^{\,2}} \quad \text{and} \quad A_l = \frac{L_{ls}}{N_s^{\,2}} \qquad 4.24$$

where N_s is the reference turn number when motor is symmetrical.

Before we move on to the definition of asymmetrical machine inductances let assume stator phase *as*, *bs* and *cs* have number of winding turns given by N_a, N_b, and N_c respectively, and rotor phase *ar*, *br* and *cr* have winding turns given by $N_{ar} = N_{br} = N_{cr} = N_r$.

4.2.2.1 Stator Self-Inductances

By using these known parameters the self-inductances for phase *as*, *bs* and *cs* can be calculated by setting $\alpha = 0$ and N_x and N_y to N_a, N_b and N_c for phase *as*, *bs* and *cs* respectively. Therefore

$$L_{asas} = L_{las} + L_{mas} = N_a^{\ 2} A_l + N_a^{\ 2} A_g = \frac{N_a^{\ 2}}{N_s^{\ 2}} \left(L_{ls} + \frac{2}{3} L_m \right) = N_a^{\ 2} L_{mls} \qquad 4.25$$

$$L_{bsbs} = L_{lbs} + L_{mbs} = N_b^{\ 2} A_l + N_b^{\ 2} A_g = \frac{N_b^{\ 2}}{N_s^{\ 2}} \left(L_{ls} + \frac{2}{3} L_m \right) = N_b^{\ 2} L_{mls} \qquad 4.26$$

$$L_{cscs} = L_{lcs} + L_{mcs} = N_c^{\ 2} A_l + N_c^{\ 2} A_g = \frac{N_c^{\ 2}}{N_s^{\ 2}} \left(L_{ls} + \frac{2}{3} L_m \right) = N_c^{\ 2} L_{mls} \qquad 4.27$$

where $L_{mls} = \frac{1}{N_s^{\ 2}} \left(L_{ls} + \frac{2}{3} L_m \right)$.

4.2.2.2 Stator Mutual Inductances

The stator mutual inductances between phase *as* and *bs*, *bs* and *cs*, and *cs* and *as* can be derived by setting $\alpha = \frac{2\pi}{3}$ in Equation 4.18. The results are

$$L_{asbs} = L_{masbs} = -\frac{1}{2} N_a N_b A_g = \left(-\frac{1}{2} N_a N_b \right) \left(\frac{2}{3} \frac{L_m}{N_s^{\ 2}} \right) = -\frac{1}{3} \frac{N_a N_b}{N_s^{\ 2}} L_m = N_a N_b L_{mss} \qquad 4.28$$

$$L_{ascs} = L_{mascs} = -\frac{1}{2} N_a N_c A_g = \left(-\frac{1}{2} N_a N_c \right) \left(\frac{2}{3} \frac{L_m}{N_s^{\ 2}} \right) = -\frac{1}{3} \frac{N_a N_c}{N_s^{\ 2}} L_m = N_a N_c L_{mss} \qquad 4.29$$

$$L_{bsas} = L_{asbs} = N_a N_b L_{mss} \quad 4.30$$

$$L_{bscs} = L_{mbscs} = -\frac{1}{2} N_b N_c A_g = \left(-\frac{1}{2} N_b N_c\right)\left(\frac{2}{3}\frac{L_m}{N_s^{\,2}}\right) = -\frac{1}{3}\frac{N_b N_c}{N_s^{\,2}} L_m = N_b N_c L_{mss} \quad 4.31$$

$$L_{csas} = L_{ascs} = N_a N_c L_{mss} \quad 4.32$$

$$L_{csbs} = L_{bscs} = N_b N_c L_{mss} \quad 4.33$$

where $L_{mss} = -\frac{1}{3}\frac{L_m}{N_s^{\,2}}$

4.2.2.3 Rotor Self and Mutual Inductances

The rotor self and mutual inductances can be found by a similar way. Because the rotor is assumed symmetric, the total self-inductance of rotor phase *ar, br* and *cr* are equal. Therefore

$$L_{arar} = L_{brbr} = L_{crcr} = L_{lr} + N_r^{\,2} A_g = L_{lr} + \frac{2}{3}\frac{N_r^{\,2}}{N_s^{\,2}} L_m = L_{lr} + L_{mar} \quad 4.34$$

And for the same reason rotor mutual inductances are also equal to each other

$$L_{arbr} = L_{arcr} = L_{brar} = L_{brcr} = L_{crar} = L_{crbr} = -\frac{1}{2} N_r^{\,2} A_g = -\left(\frac{1}{2}\right)\left(\frac{2}{3}\frac{N_r^{\,2}}{N_s^{\,2}} L_m\right) = -\frac{1}{2} L_{mar} \quad 4.35$$

where $L_{mar} = \frac{2}{3}\frac{N_r^{\,2}}{N_s^{\,2}} L_m$

4.2.2.4 Stator-to-Rotor Mutual Inductances

Previously defined stator-to-rotor mutual inductances can be defined in term of new parameters. Because of rotor symmetry (turn numbers for each rotor phase are equal) mutual inductances will be $L_{asar} = L_{asbr} = L_{ascr}$, $L_{bsar} = L_{bsbr} = L_{bscr}$, and $L_{csar} = L_{csbr} = L_{cscr}$.

Referring to Figure 4.1, we can see that the rotor phase *ar* is displaced from the stator phase *as* by the electrical angle θ_r, where θ_r is a variable. The corresponding mutual inductances can be obtained by setting

i. $\alpha = \theta_r$ for the mutual inductances between *as* and *ar* , *bs* and *br*, and *cs* and *cr*

ii. $\alpha = \theta_r + \frac{2\pi}{3}$ for the mutual inductances between *as* and *br*, *bs* and *cr*, and *cs* and *ar*

iii. $\alpha = \theta_r - \frac{2\pi}{3}$ for the mutual inductances between *as* and *cr*, *bs* and a*r*, and *cs* and *br*

respectively.

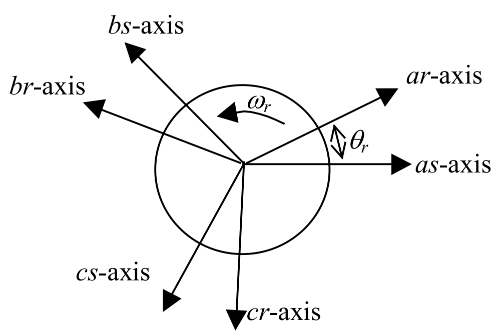


Figure 4.1: Induction machine winding displacement.

Cosine terms are already appear in Equation 4.9, so peak mutual inductances will be

$$L_{asar} = L_{asbr} = L_{ascr} = N_a N_r A_g = \frac{2}{3}\frac{N_a N_r}{N_s^2} L_m = N_a L_{msr} \qquad 4.36$$

$$L_{bsar} = L_{bsbr} = L_{bscr} = N_b N_r A_g = \frac{2}{3}\frac{N_b N_r}{N_s^2} L_m = N_b L_{msr} \qquad 4.37$$

$$L_{csar} = L_{csbr} = L_{cscr} = N_c N_r A_g = \frac{2}{3}\frac{N_c N_r}{N_s^2} L_m = N_c L_{msr} \qquad 4.38$$

where $L_{msr} = \frac{2}{3}\frac{N_r}{N_s^2} L_m$.

4.2.3 The Electromagnetic Torque

The electrical power into the terminals of the motor can be segregated into three parts: the power dissipated in the stator and rotor resistances, magnetic energy storage in the inductances, and the power being converted from electrical to mechanical form. The last one is the electromechanical output power P_{em}, and it is equal to

$$P_{em} = \frac{3}{2}\omega_r(\lambda_d^s i_q^s - \lambda_q^s i_d^s) \qquad 4.39$$

The electrical angular velocity of machine, ω_r, is related to the actual mechanical angular speed of the pole pairs. That is

$$\omega_{rm} = \frac{2\omega_r}{P} \qquad 4.40$$

where P is number of poles and ω_{rm} is the mechanical speed in radian per second. Since the rotational mechanical power is defined as the product of speed times torque, it is apparent that the electromagnetic torque impressed on the shaft of the machine can be expressed as

$$T_{em} = \frac{3}{2}\frac{P}{2}(\lambda_d^s i_q^s - \lambda_q^s i_d^s) \qquad 4.41$$

The equation of motion of the rotor is obtained by equating the inertia torque to the accelerating torque, which is

$$\frac{2J}{P}\frac{d\omega_r}{dt} = T_{em} + T_{mech} - T_{damp} \qquad 4.42$$

T_{mech} is the externally applied mechanical torque in the direction of the rotor speed, T_{damp} is the damping torque in the opposite direction of the rotor speed, J is inertia.

4.2.4 Simulation of the Asymmetrical Induction Motors

In this subsection, SIMULINK models for the asymmetrical induction motor model that has been developed will be given.

4.2.4.1 Supply Voltage Model

The *qd*-applied voltage in a reference frame fixed to the stator can be obtained from the stator phase voltage v_{as}, v_{bs}, and v_{cs} by applying $\mathbf{T}_{qd0}(0)$ transformation. The *qd* voltages will be

$$v_q^s = \frac{2}{3}\left[v_{as} - \frac{1}{2}(v_{bs} + v_{cs})\right] = \frac{2}{3}\left[v_{ag} - \frac{1}{2}(v_{bg} + v_{cg})\right] \qquad 4.43$$

$$v_d^s = \frac{1}{\sqrt{3}}(-v_{bs} + v_{cs}) = \frac{1}{\sqrt{3}}(-v_{bg} + v_{cg}) \qquad 4.44$$

where v_{ag}, v_{bg}, and v_{cg} are supply phase voltages.

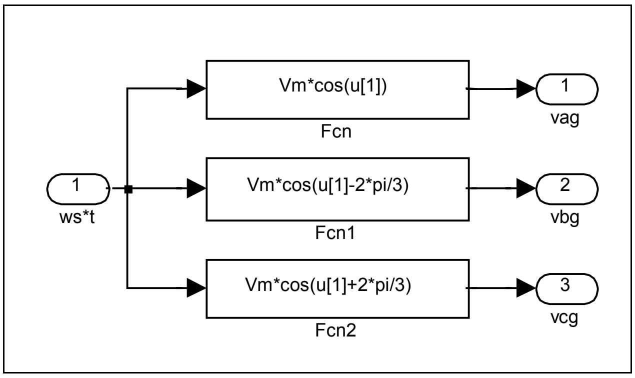


Figure 4. 2: Supply voltage model

4.2.4.2 Stator and Rotor Model

The model equation of the induction motor in the stationary reference frame may be rearranged into more compact form in term of flux linkages and reactances. These are given by:

$$\psi = \omega_s \lambda \qquad X = \omega_s L$$

where ω_s is synchronous angular velocity. Flux linkages may be obtained from Equations 4.5 and 4.6 for a 3-wire system to give

$$\psi_q^s = \omega_s \int (v_q^s - r_{11}^s i_q^s - r_{12}^s i_d^s) dt \qquad 4.45$$

$$\psi_d^s = \omega_s \int (v_d^s - r_{21}^s i_q^s - r_{22}^s i_d^s) dt \qquad 4.46$$

$$\psi_q^r = \int (\omega_r \psi_d^r - \omega_s r_r^r i_q^r) dt \qquad 4.47$$

$$\psi_d^r = -\int (\omega_r \psi_q^r + \omega_s r_r^r i_d^r) dt \qquad 4.48$$

The current can be found by inverting Equation 4.17

$$\begin{bmatrix} i_q^s \\ i_d^s \\ i_q^r \\ i_d^r \end{bmatrix} = \begin{bmatrix} X_{11}^{ss} & X_{12}^{ss} & X_{11}^{sr} & X_{12}^{sr} \\ X_{21}^{ss} & X_{22}^{ss} & X_{21}^{sr} & X_{22}^{sr} \\ X_{11}^{sr} & X_{12}^{sr} & X_{11}^{rr} & 0 \\ X_{21}^{sr} & X_{22}^{sr} & 0 & X_{22}^{rr} \end{bmatrix}^{-1} \begin{bmatrix} \psi_q^s \\ \psi_d^s \\ \psi_q^r \\ \psi_d^r \end{bmatrix} \qquad 4.49$$

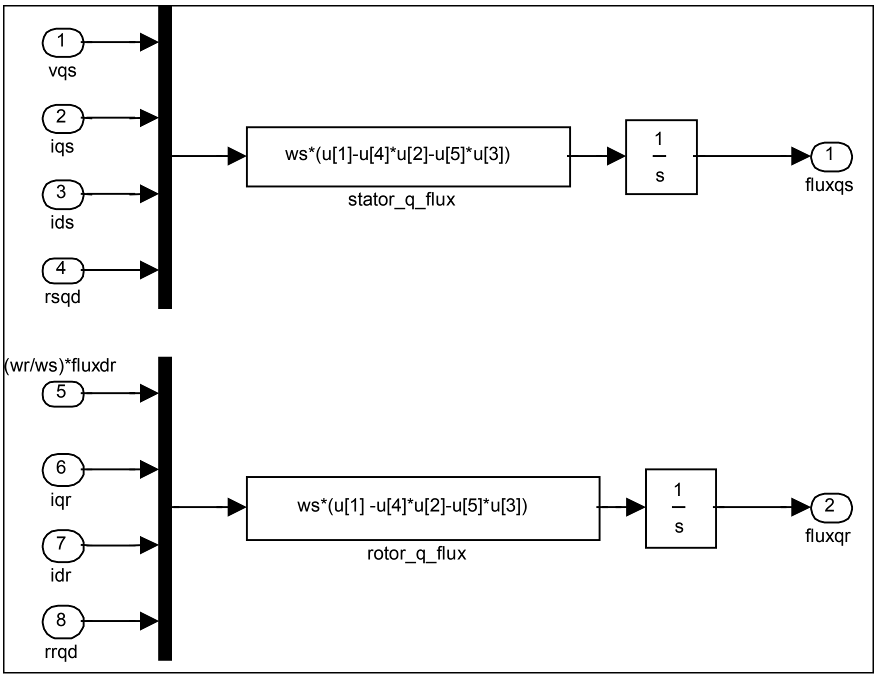


Figure 4. 3: Stator and rotor q-axis model

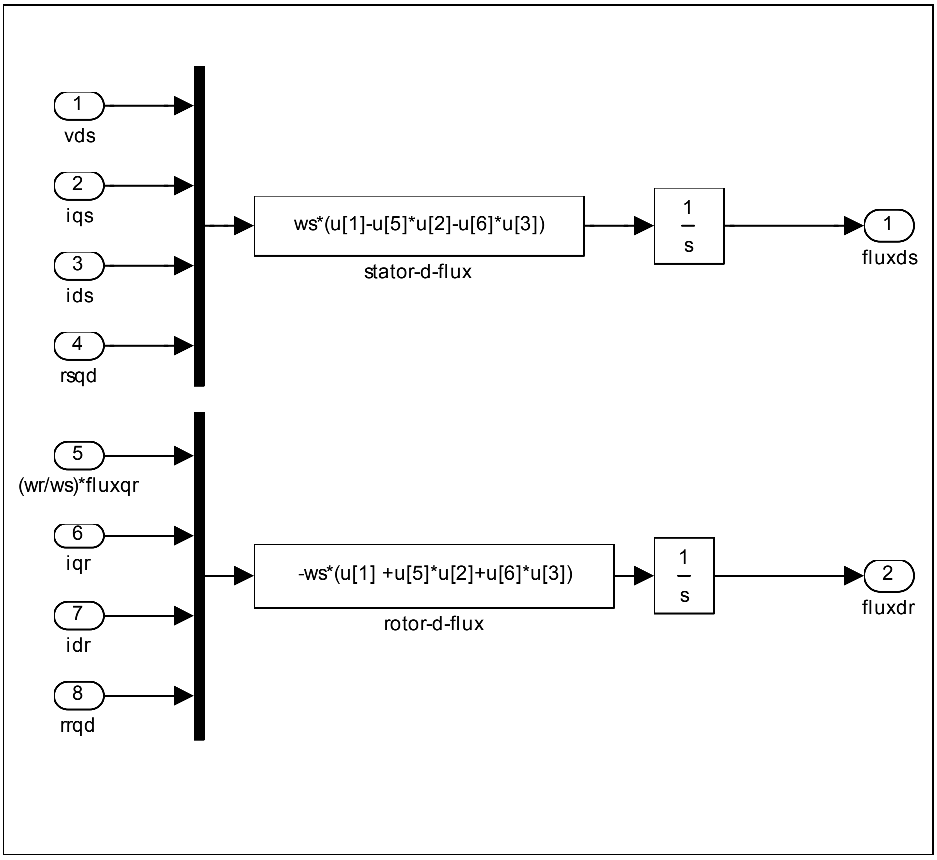


Figure 4. 4: Stator and rotor d-axis model

4.2.4.3 Load Model

The speed of machine can be obtained from torque equation as

$$\omega_r(t) = \frac{P}{2J}\int (T_{em} + T_{mech} - T_{damp})dt \qquad 4.50$$

$\theta_r(t)$, rotor angle, can be obtained by integrating the motor speed

$$\theta_r(t) = \int \omega_r(t)dt + \theta_r(0) \qquad 4.51$$

where $\theta_r(0)$ is initial rotor angle with respect to the stator reference phase.

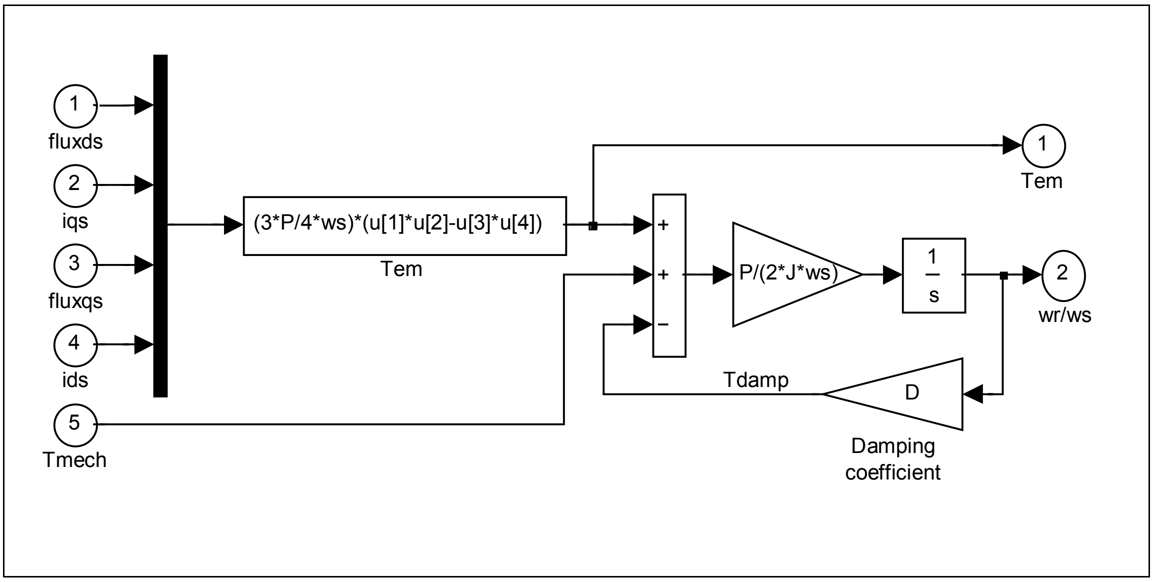


Figure 4.5: Load model

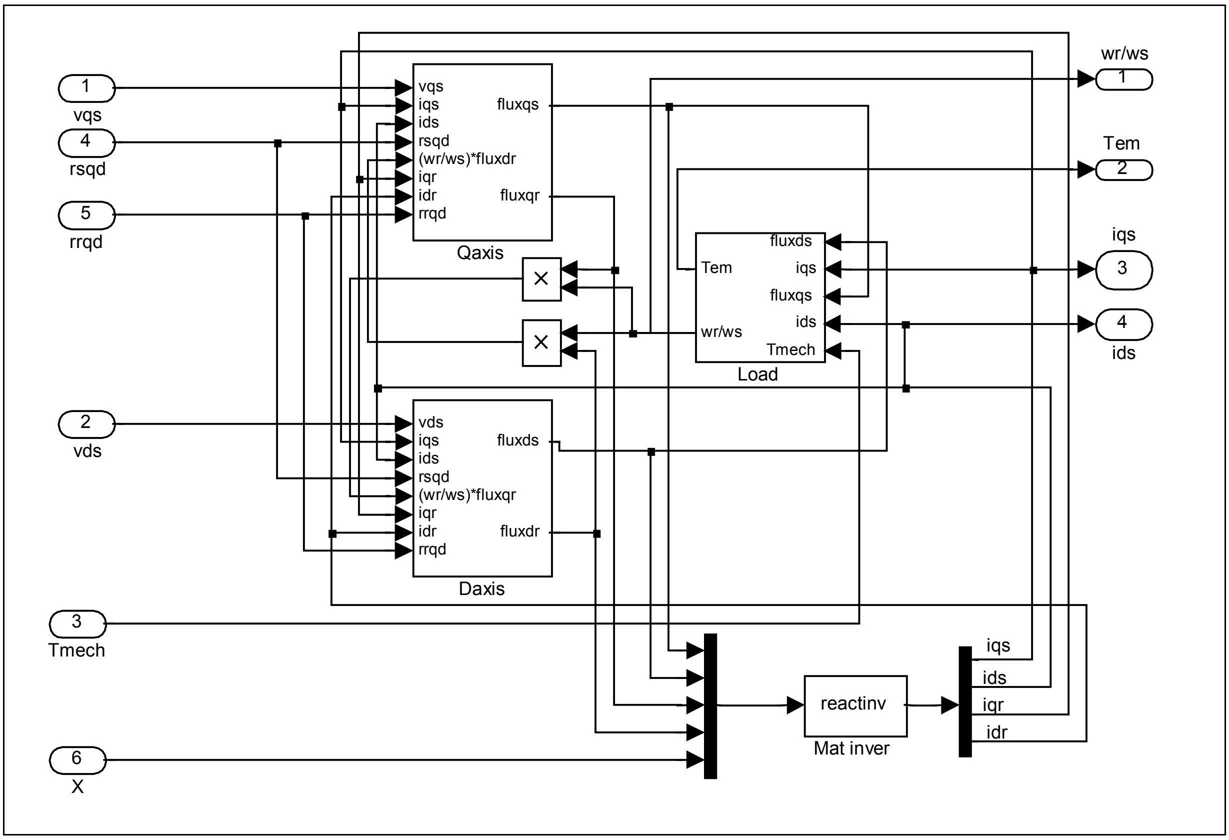


Figure 4.6: Motor model (Input (6) feeds in the reactance values of Equation 4.49).

4.2.4.4 Fault Diagnosis Model

The fault diagnosis model contains the Signal Sampling Model, PDT Model and Fault Analysis Model. The signal sampling model samples voltage and current signals and feeds this sampled data to the PDT model for decomposing signals into their negative and positive sequence components. The fault analysis model gives condition of motor by looking at the motor negative sequence current. Figure 4.7 shows overall simulation model which '*General'* object contained motor model, PDT model, and fault diagnosis models.

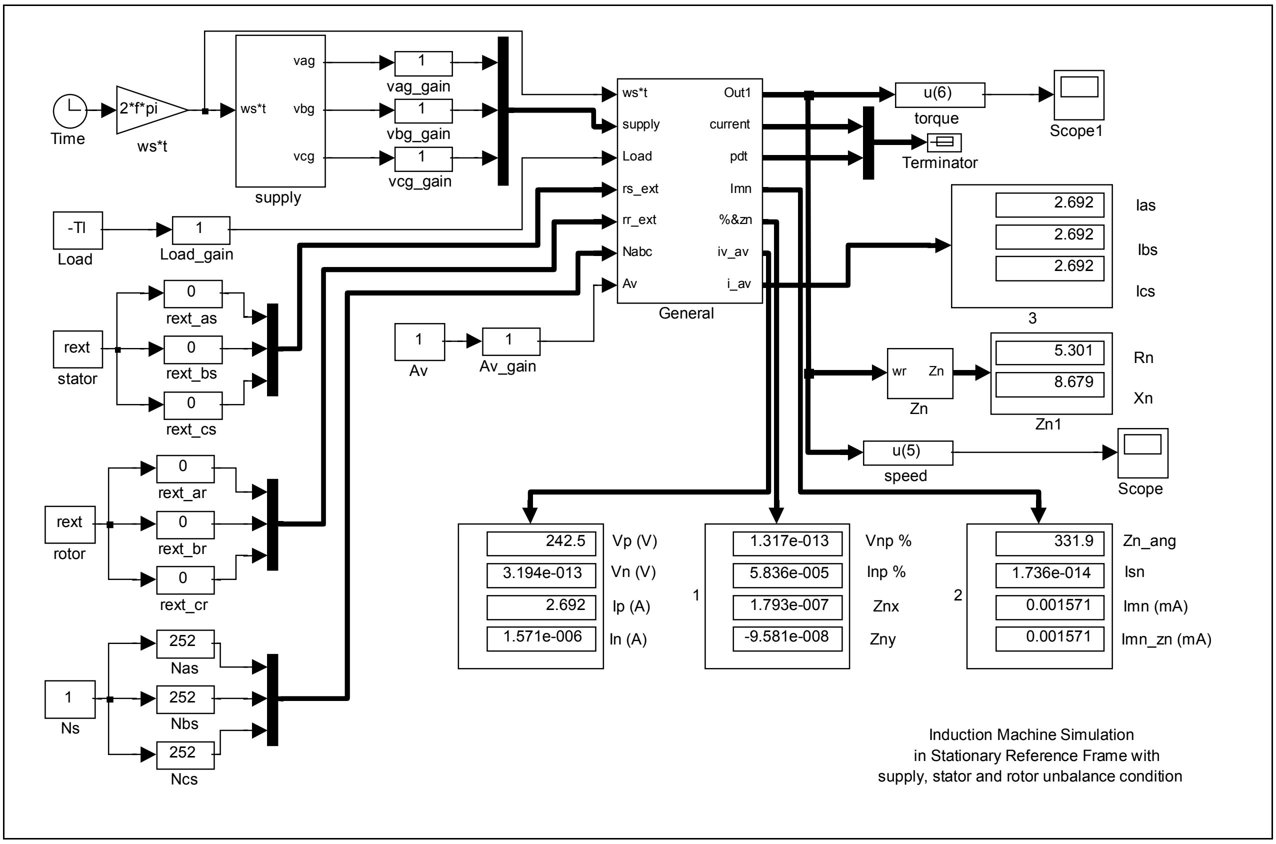


Figure 4.7: Overall induction motor simulation model

By using these equations and resistances and inductances that are defined in Appendix D, a motor with asymmetrically windings can be simulated. The compact model for the simulation of asymmetrical motor is shown in Figure 4.8.

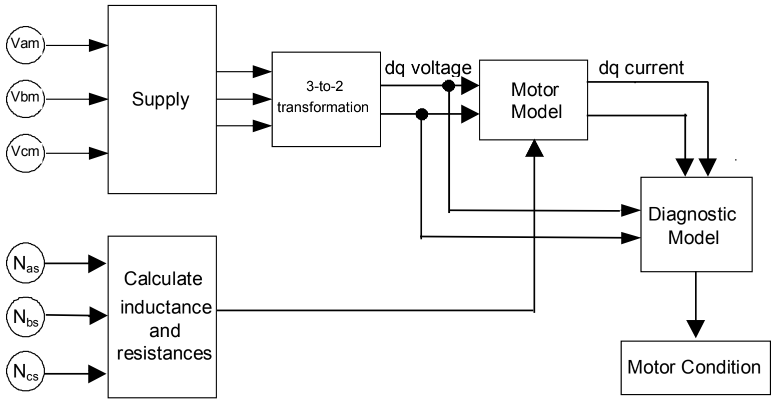


Figure 4.8: Simulation of asymmetrical induction motor model

4.3 Modelling of Induction Motors with Inter-Turn Stator Short Circuit

In order to develop an induction motor model with a stator inter-turn short circuit, it has been assumed that phase-*as* has two series windings; unshorted number of turns N_{us} and shorted number of turn(s) N_{sh} where $N_{as} = N_{us} + N_{sh} = N_s$, the overall number of turns N_s. The phase-*bs*, and –*cs* have $N_{bs} = N_{cs} = N_s$. By assuming the unequal number of stator turns, the first model can be used to transfer motor equation from *abc* axis to *qd0* axis. By doing this, inter-turn short circuit can be introduced in *qd0* axis. The degree of fault can be changed by number of shorted turns and current limiting resistance across the short circuit windings. The same assumptions that were made for the first model are valid for this model as well.

In the following subsections modification of equations for the new model will be discussed.

4.3.1 Stator Self Inductances

The self-inductances for phase *as, bs,* and *cs* in Equation 4.25, 4.26 and 4.27 will be

$$
\begin{aligned}
L_{asas} &= \frac{N_{us}^{2}}{N_{s}^{2}}\left(L_{ls}+\frac{2}{3}L_{m}\right)+2N_{us}N_{sh}\left(\frac{2}{3}\frac{L_{m}}{N_{s}^{2}}\right)+\frac{N_{sh}^{2}}{N_{s}^{2}}\left(L_{ls}+\frac{2}{3}L_{m}\right)\\
&=\left(N_{us}^{2}L_{mls}+N_{us}N_{sh}L_{msh}\right)+\left(N_{us}N_{sh}L_{msh}+N_{us}^{2}L_{mls}\right)\\
&=\left(L'_{asas}+L_{assh}\right)+\left(L_{assh}+L_{shsh}\right)
\end{aligned}
\qquad 4.52
$$

where $L_{msh} = \frac{2}{3}\frac{L_m}{N_s^2}$ and subscript *sh* is used for shorted winding(s).

$$L_{bsbs} = L_{\csc s} = \frac{N_s^{2}}{N_s^{2}}\left(L_{ls}+\frac{2}{3}L_m\right) = N_s^{2}\,L_{mls} \qquad 4.53$$

As it can be seen from Equation 4.52 phase *as* self-inductance contains unshorted part self-inductance L'_{asas}, mutual inductance between unshorted and shorted parts L_{assh}, and shorted part self inductance L_{shsh}.

4.3.2 Stator Mutual Inductances

The stator mutual inductances will be

$$L_{asbs} = -\frac{1}{3}\frac{N_a N_s}{N_s^{2}}L_m = -\frac{1}{3}\frac{N_{us}}{N_s}L_m - \frac{1}{3}\frac{N_{sh}}{N_s}L_m = L'_{asbs} + L_{shbs} \qquad 4.54$$

$$L_{ascs} = L_{bsas} = L_{csas} = L'_{asbs} + L_{shbs}$$

$$L_{bscs} = L_{csbs} = -\frac{1}{3}L_m \qquad 4.55$$

4.3.3 Stator-to-Rotor Mutual Inductances

The stator-to-rotor mutual inductances in Equations 4.36, 4.37, and 4.38 will be

$$L_{asar} = L_{asbr} = L_{ascr} = \frac{2}{3}\frac{N_a N_r}{N_s^{\ 2}} L_m = \frac{2}{3}\frac{N_{us} N_r}{N_s^{\ 2}} L_m + \frac{2}{3}\frac{N_{sh} N_r}{N_s^{\ 2}} L_m \\ = L'_{asar} + L_{shar} \qquad 4.56$$

$$L_{bsar} = L_{bsbr} = L_{bscr} = L_{csar} = L_{csbr} = L_{\mathrm{csc}\,r} = \frac{2}{3}\frac{N_s N_r}{N_s^{\ 2}} L_m \qquad 4.57$$

4.3.4 Inductances in *dq0* Axis

As in section 4.2, assuming that stator has different number of turns, *dq0* axis inductances can be obtained by using the new defined self and mutual inductances. The transformation of $\mathbf{L}_{abc}^{ss}$ to $\mathbf{L}_{qd0}^{ss}$ and $\mathbf{L}_{abc}^{sr}$ to $\mathbf{L}_{qd0}^{sr}$ are

$$\mathbf{L}_{qd0}^{ss} = \begin{bmatrix} L_{11}^{ss} & 0 & L_{13}^{ss} \\ 0 & L_{22}^{ss} & 0 \\ L_{31}^{ss} & 0 & L_{33}^{ss} \end{bmatrix} \text{ and } \mathbf{L}_{qd0}^{sr} = \begin{bmatrix} L_{11}^{sr} & 0 & 0 \\ 0 & L_{22}^{sr} & 0 \\ L_{31}^{sr} & 0 & 0 \end{bmatrix} \qquad 4.58$$

where

$$L_{11}^{ss} = \left(L_q^{s} + L_q^{ssh}\right) + \left(L_q^{sh} + L_q^{ssh}\right) \text{ and } L_{22}^{ss} = L_d^{s} \qquad 4.59$$

$$L_{11}^{sr} = L_q^{sr} + L_q^{shr} \quad \text{and} \quad L_{22}^{sr} = L_d^{sr} \qquad 4.60$$

Similarly, result of transformation of rotor self and mutual inductances can be simplified as

$$\mathbf{L}_{qd0}^{rr} = \begin{bmatrix} L_{11}^{rr} & 0 & 0 \\ 0 & L_{22}^{rr} & 0 \\ 0 & 0 & L_{33}^{rr} \end{bmatrix} \quad \mathbf{L}_{qd0}^{rs} = \begin{bmatrix} L_{11}^{sr} & 0 & \frac{1}{2}L_{31}^{sr} \\ 0 & L_{22}^{sr} & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad 4.61$$

where $L_{11}^{rr} = L_q^{r}$ and $L_{22}^{rr} = L_d^{r}$.

Matrix elements are given in Appendix D.

The stator and rotor flux linkages for new model of the squirrel cage induction motors will be

$$\begin{bmatrix} \lambda_q^{sh} \\ \lambda_q^s \\ \lambda_d^s \\ \lambda_q^r \\ \lambda_d^r \end{bmatrix} = \begin{bmatrix} L_q^{sh} & L_q^{ssh} & 0 & L_q^{shr} & 0 \\ L_q^{ssh} & L_q^s & 0 & L_q^{sr} & 0 \\ 0 & 0 & L_d^s & 0 & L_d^{sr} \\ L_q^{shr} & L_q^{sr} & 0 & L_q^r & 0 \\ 0 & 0 & L_d^{sr} & 0 & L_d^r \end{bmatrix} \begin{bmatrix} i_q^{sh} \\ i_q^s \\ i_d^s \\ i_q^r \\ i_d^r \end{bmatrix} \qquad 4.62$$

The stator phase resistances are

$$r_{as} = \frac{N_a}{N_s} r_s = \frac{N_{us}}{N_s} r_s + \frac{N_{sh}}{N_s} r_s = r'_{as} + r_{sh} \qquad 4.63$$

$$r_{bs} = r_{cs} = r_s \qquad 4.64$$

where r_{sh} shorted winding(s) resistance.

Stator *qd* resistance for new model will be

$$\begin{bmatrix} r_q^{sh} \\ r_q^s \\ r_d^s \end{bmatrix} = \begin{bmatrix} \frac{2}{3} r_{sh} & 0 & 0 \\ 0 & r_{11}^s & r_{12}^s \\ 0 & r_{21}^s & r_{22}^s \end{bmatrix} \begin{bmatrix} i_q^{sh} \\ i_q^s \\ i_d^s \end{bmatrix} \qquad 4.65$$

The matrix elements are given in Appendix D. The new model equation of the induction motor in the stationary reference frame may be rearranged in term of flux linkages per second and reactances:

Flux linkages may be obtained from Equations 4.4 and 4.5 for a 3-wire system to give

$$\psi_q^{sh} = \omega_s \int (v_q^{sh} - r_{sh} i_q^{sh}) dt \qquad 4.66$$

$$\psi_q^s = \omega_s \int (v_q^s - v_q^{sh} - r_{11}^s i_q^s - r_{12}^s i_d^s) dt \qquad 4.67$$

$$\psi_d^s = \omega_s \int \left(v_d^s - r_{21}^s i_q^s - r_{22}^s i_d^s\right) dt \qquad 4.68$$

$$\psi_q^r = \int (\omega_r \psi_d^r - \omega_s r_r^r i_q^r) dt \qquad 4.69$$

$$\psi_d^r = -\int (\omega_r \psi_q^r + \omega_s r_r^r i_d^r) dt \qquad 4.70$$

Note that v_q^{sh} will be zero if external current limiting resistance is zero. The current can be found by inverting Equation 4.62

$$\begin{bmatrix} i_q^{sh} \\ i_q^s \\ i_d^s \\ i_q^r \\ i_d^r \end{bmatrix} = \begin{bmatrix} X_q^{sh} & X_q^{ssh} & 0 & X_q^{shr} & 0 \\ X_q^{ssh} & X_q^s & 0 & X_q^{sr} & 0 \\ 0 & 0 & X_d^s & 0 & X_d^{sr} \\ X_q^{shr} & X_q^{sr} & 0 & X_q^r & 0 \\ 0 & 0 & X_d^{sr} & 0 & X_d^r \end{bmatrix}^{-1} \begin{bmatrix} \psi_q^{sh} \\ \psi_q^s \\ \psi_d^s \\ \psi_q^r \\ \psi_d^r \end{bmatrix} \qquad 4.71$$

The following figures show the SIMULINK models of a short circuited induction motor.

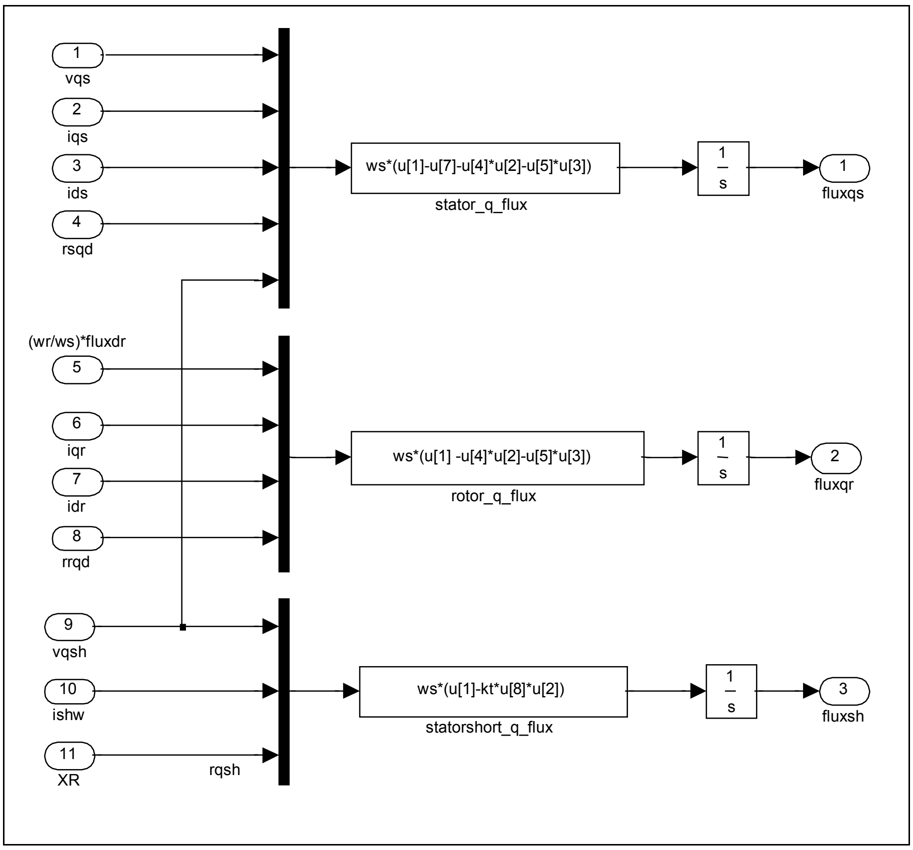


Figure 4.9: q-axis stator and rotor model

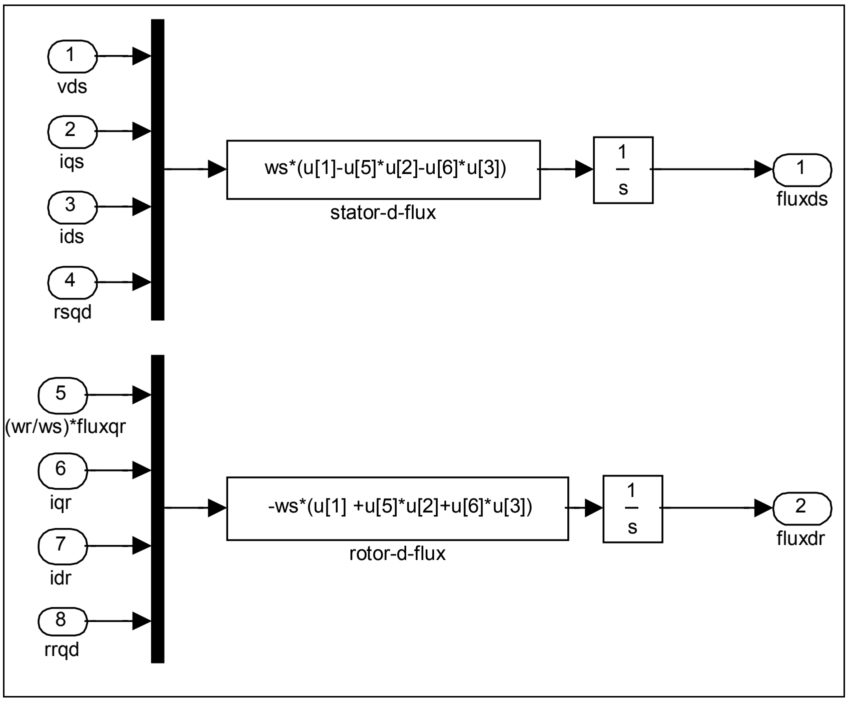


Figure 4.10: d-axis stator and rotor model

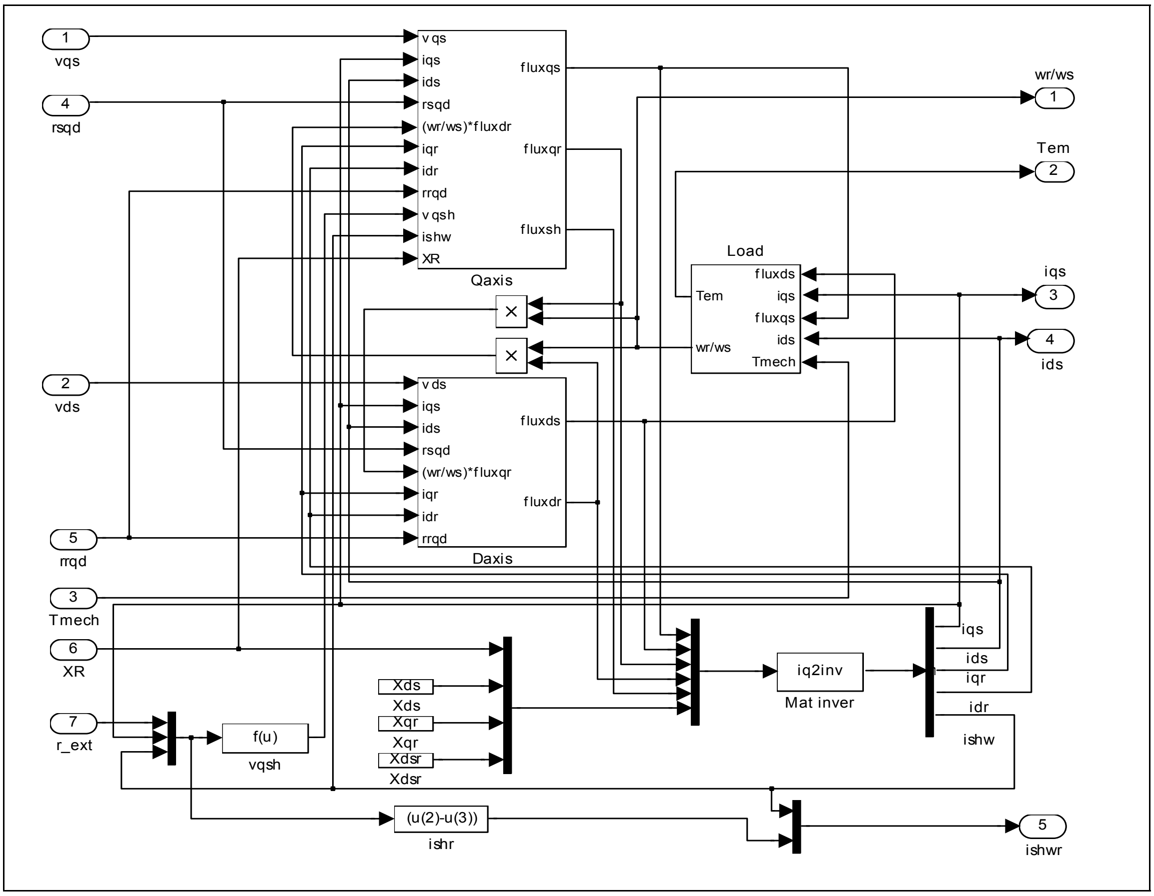


Figure 4.11: Complete motor model

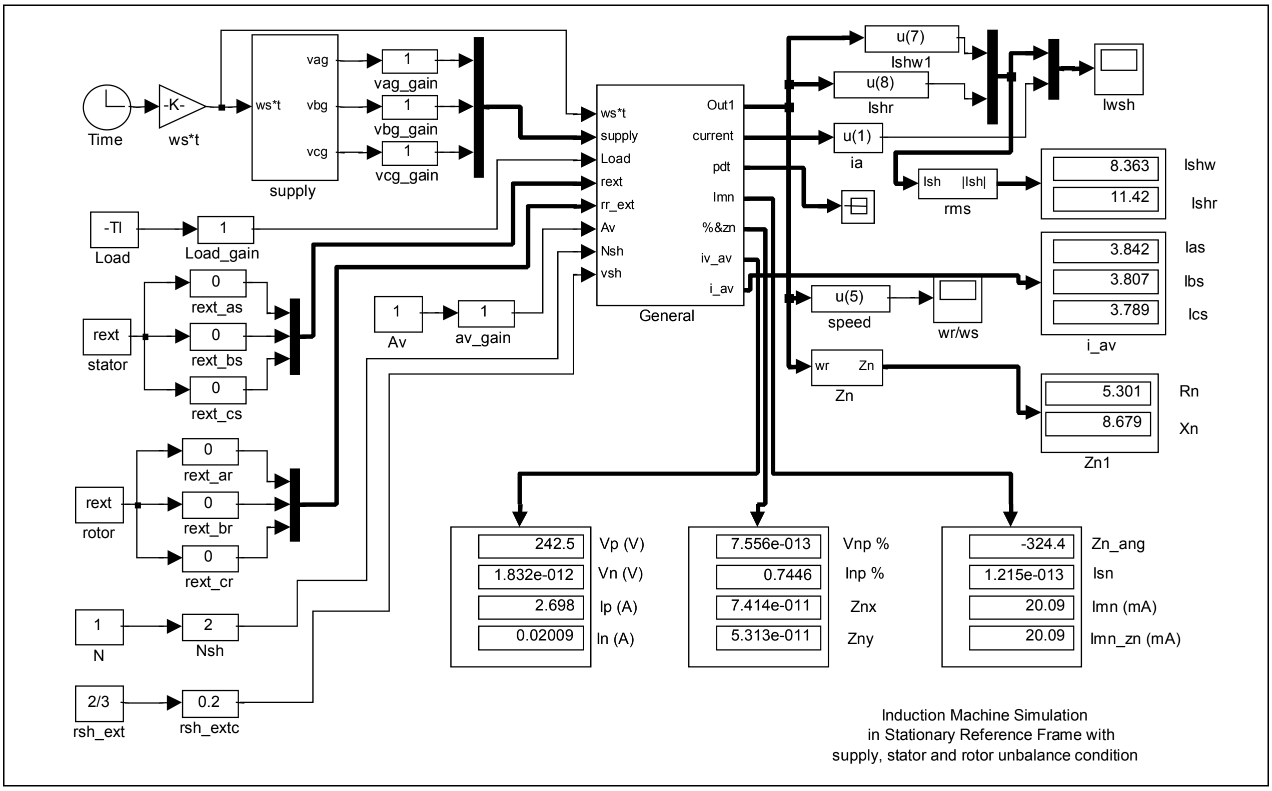


Figure 4.12: Overall induction motor simulation model

4.4 Treatment of Rotor Asymmetry

Although rotor asymmetries have not been considered, they can also be simulated easily. The rotor fault can be simulated by putting extra resistance in one or more rotor phases. For example cage damage can be implemented as an extra resistance in the rotor phase [59]. At fault condition phase resistances may be rewritten as $r_{ar} = r_r + r_{ar\_ext}$, $r_{br} = r_r + r_{br\_ext}$, and $r_{cr} = r_r + r_{cr\_ext}$ for phase *ar, br, cr* respectively and $\mathbf{r}^r_{qd0}$ will be

$$\mathbf{r}^r_{qd0} = \begin{bmatrix} r^r_{11} & r^r_{12} & r^r_{13} \\ r^r_{21} & r^r_{22} & r^r_{23} \\ r^r_{31} & r^r_{32} & r^r_{32} \end{bmatrix}$$ and matrix elements are given in appendix D.

Flux linkages have to be rearranged for rotor in Equations 4.47, 4.48, 4.69, and 4.70. The new flux linkages for rotor will be

$$\psi_q^r = \int (\omega_r \psi_d^r - \omega_s r_{11}^r i_q^r - \omega_s r_{12}^r i_d^r) dt \qquad 4.72$$

$$\psi_d^r = -\int (\omega_r \psi_q^r + \omega_s r_{21}^r i_q^r + \omega_s r_{22}^r i_d^r) dt \qquad 4.73$$

4.5 Simulation Results

The following simulation results are for a 2 hp motor whose parameters are given in Appendix D. The results are taken from stand still to full speed.

Figure 4.13, 4.12, and 4.15 are results for acceleration from stand still to full speed at full load under normal running condition. At starting, motor currents are not symmetrical because of the starting transient. This will introduce very high negative sequence current until the motor reaches full speed.

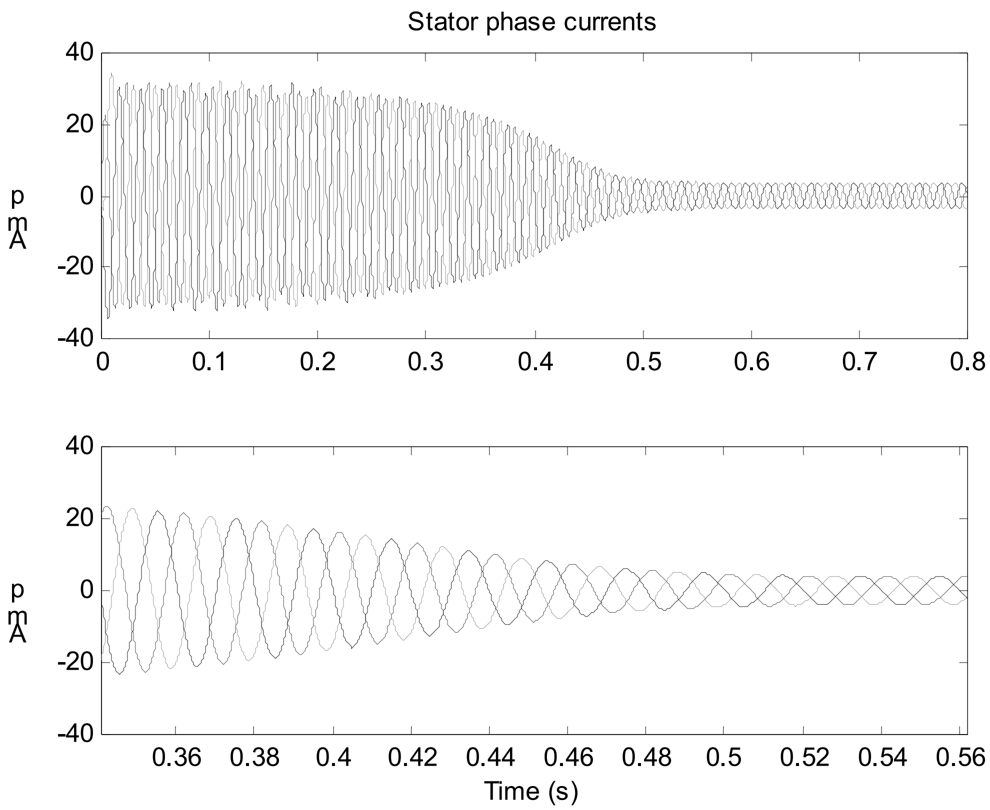


Figure 4.13: Stator phase currents for normal running of ideal motor.

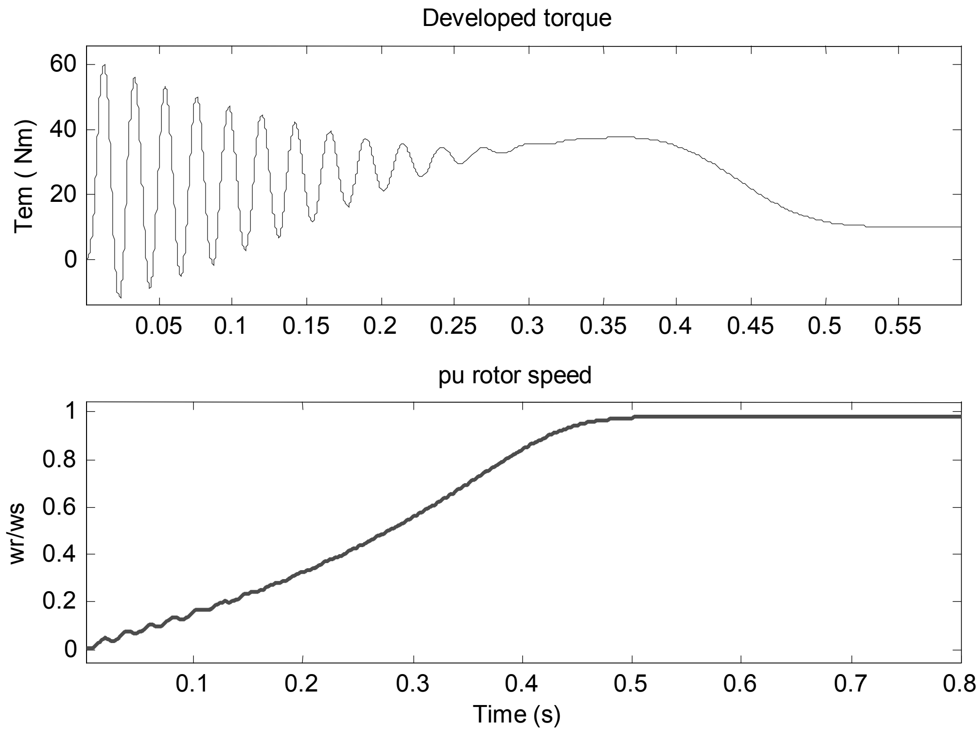


Figure 4.14: Torque and speed variation

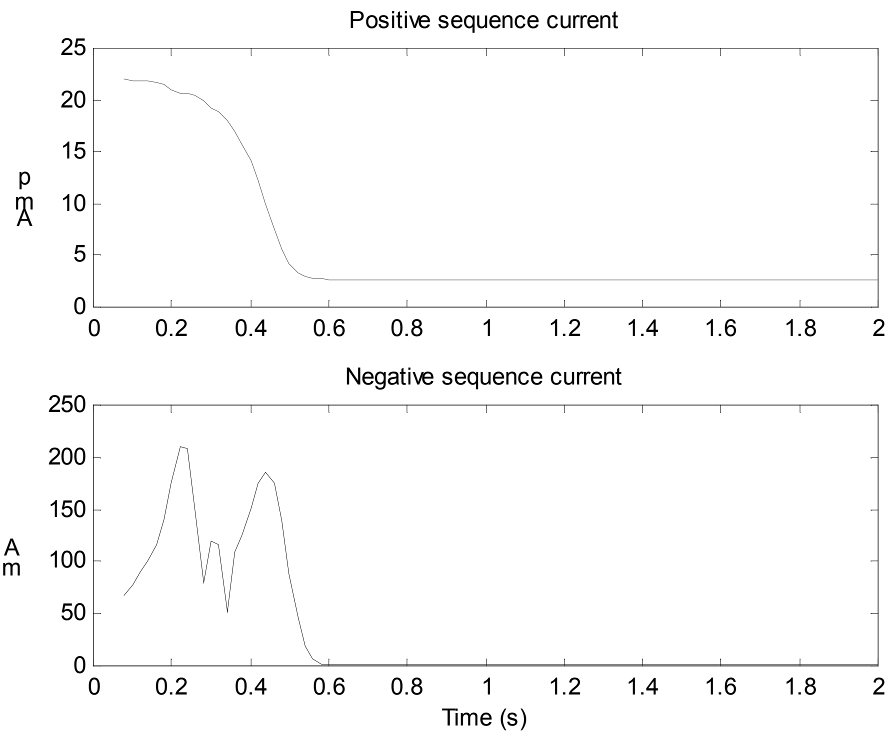


Figure 4.15: Positive and negative sequence currents (rms)

Figures 4.16 and 4.17 show the simulation results from stand still to full speed at full load with five turns shorted. The supply negative sequence current is around 400mA. Because of stator asymmetry, the torque graph in Figure 4.16 shows pulsation at twice supply frequency even at steady state. This is because of negative sequence current, which is introducing a braking torque in the motor. The short circuit current of 60 amps in the shorted turns is nearly three times the normal locked rotor current of 23 amps.

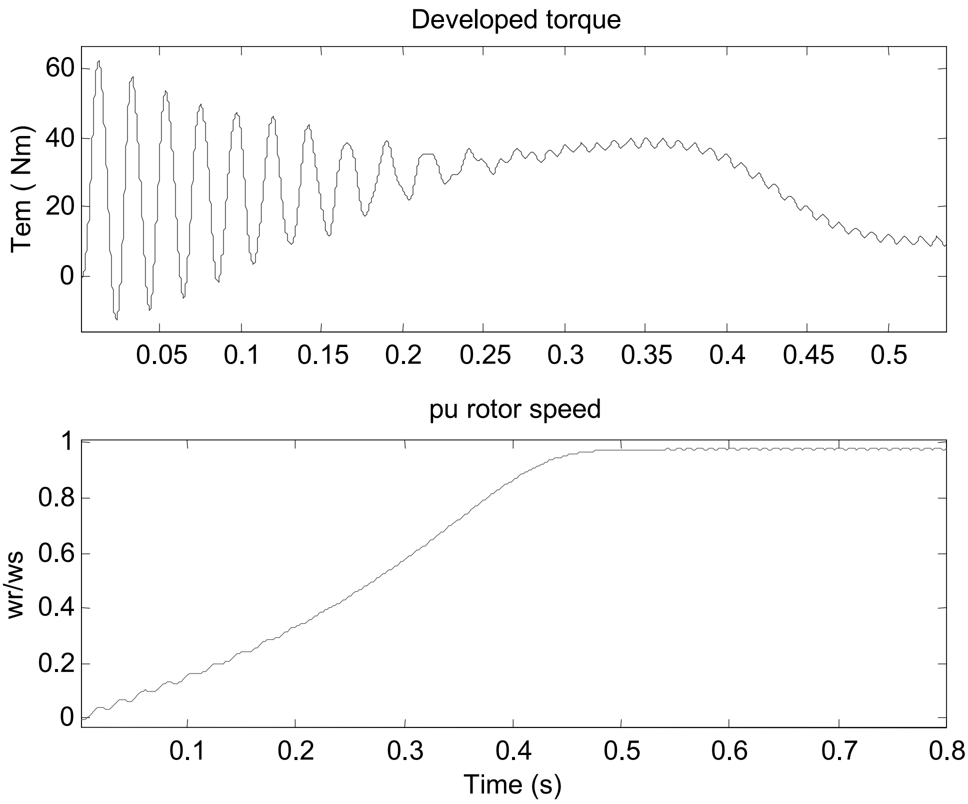


Figure 4.16: Torque and speed variation with 5 turns shorted

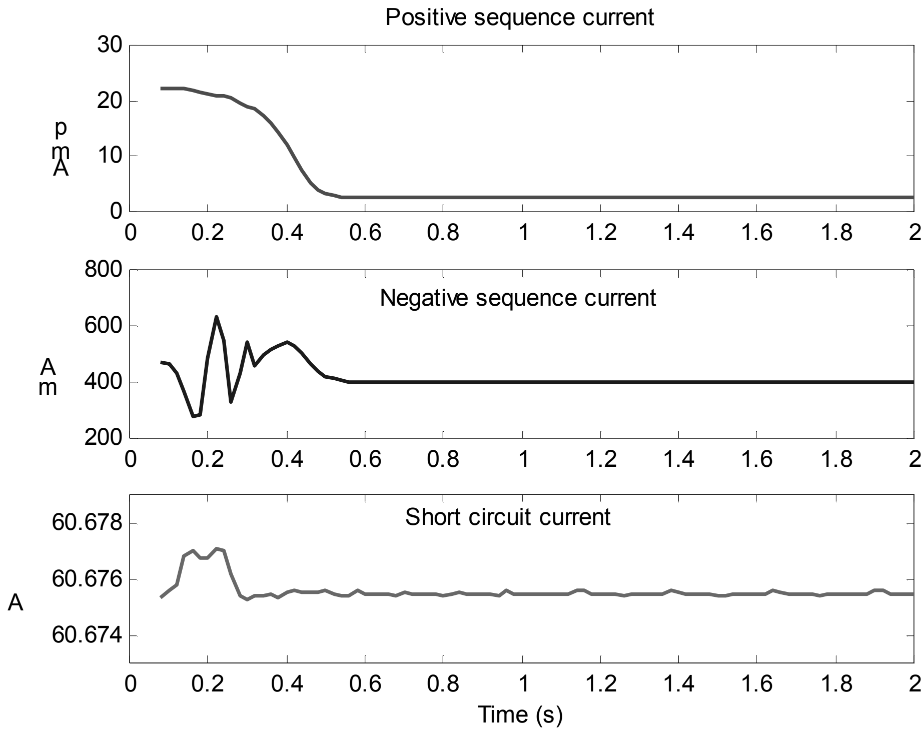


Figure 4.17: Positive sequence, negative sequence and short circuit currents for 5 turns shorted.

Figure 4.18 and 4.19 are for same condition with 1.5 ohm resistance to limit the short circuit current to avoid destruction of the motor. This time short circuit current is reduced to nearly 3 A and supply negative sequence current to 27 mA. The torque still has pulsation but is very small as seen in the Figure 4.18.

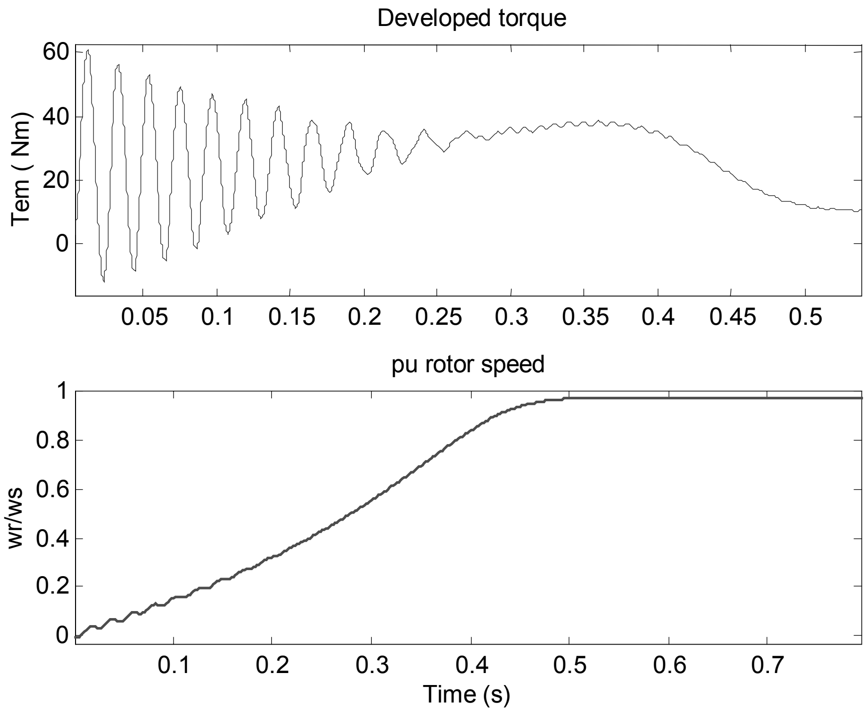


Figure 4.18: Torque and speed variation with 5 turns shorted with 1.5 ohm resistance

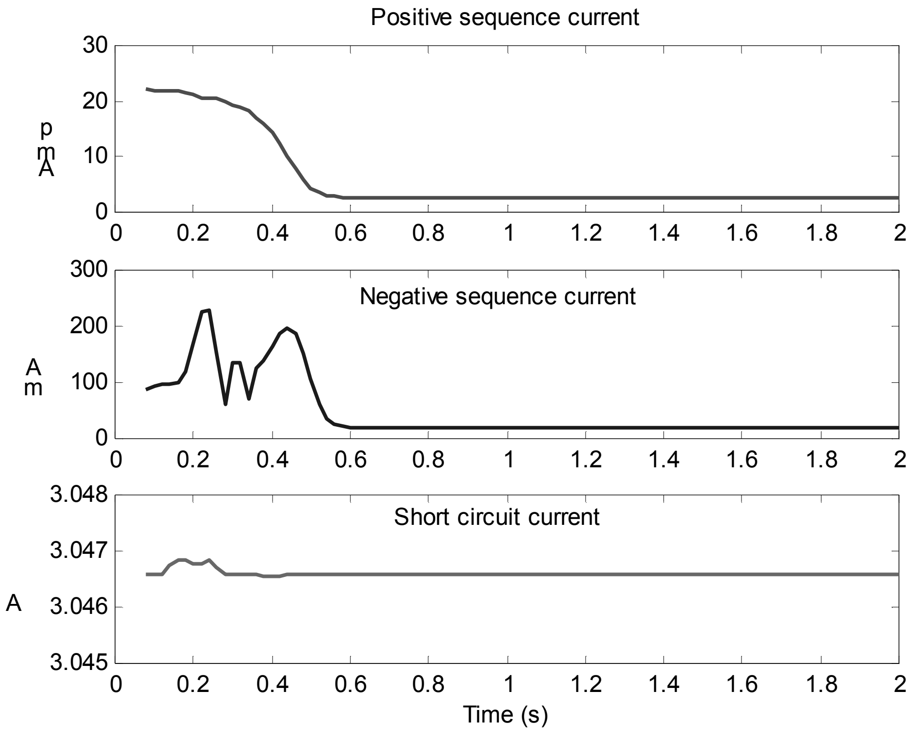


Figure 4.19: Positive sequence, negative sequence and short circuit currents for 5 turns shorted with 1.5 ohm resistance

4.5.1 Shorted Turn Number-Negative Sequence Current Relation

Results here are obtained from simulation by changing number of shorted turns for direct short-circuiting. The negative sequence current I_n is proportional to shorted turn number n

$$I_n \propto n \quad \text{or } I_n \cong k\,n \tag{4.74}$$

where k is constant. I_{shw} is current in the shorted winding and I_{r_ext} is the current in the external resistance (Figure 6.15) which is set to zero ohms. Total number of turns per phase is 252.

n	I_n (A)	I_{shw} (A)	I_{r_ext} (A)	I_p (A)	I_{as} (A)	I_{bs} (A)	I_{cs} (A)
0	0	0	0	2.693	2.693	2.693	2.693
1	.054	56.09	58.44	2.692	2.745	2.678	2.653
2	.109	56.18	58.60	2.691	2.799	2.663	2.614
3	.166	56.27	58.75	2.691	2.852	2.646	2.578
4	.222	56.37	58.89	2.689	2.908	2.633	2.542
5	.279	56.44	59.03	2.690	2.962	2.619	2.507
6	.337	56.54	59.17	2.690	3.022	2.609	2.474
7	.395	56.62	59.32	2.691	3.080	2.599	2.441
8	.453	56.70	59.47	2.693	3.139	2.591	2.409
9	.513	56.77	59.61	2.695	3.199	2.585	2.380
10	.572	56.84	59.75	2.698	3.261	2.580	2.350
12	.693	56.99	60.03	2.704	3.386	2.578	2.296
14	.817	57.13	60.29	2.713	3.517	2.583	2.246
16	.941	57.27	60.54	2.724	3.651	2.598	2.202
18	1.068	57.39	60.81	2.738	3.789	2.623	2.162
20	1.198	57.51	61.07	2.753	3.932	2.658	2.128
24	1.462	57.71	61.57	2.794	4.231	2.762	2.074
28	1.736	57.91	62.04	2.845	4.548	2.911	2.041
32	2.017	58.05	62.52	2.911	4.883	3.107	2.027
36	2.309	58.18	62.96	2.991	5.240	3.348	2.029
40	2.608	58.21	63.40	3.085	5.616	3.633	2.046

Table 4. 1: Currents versus shorted turn numbers (r_ext = 0)

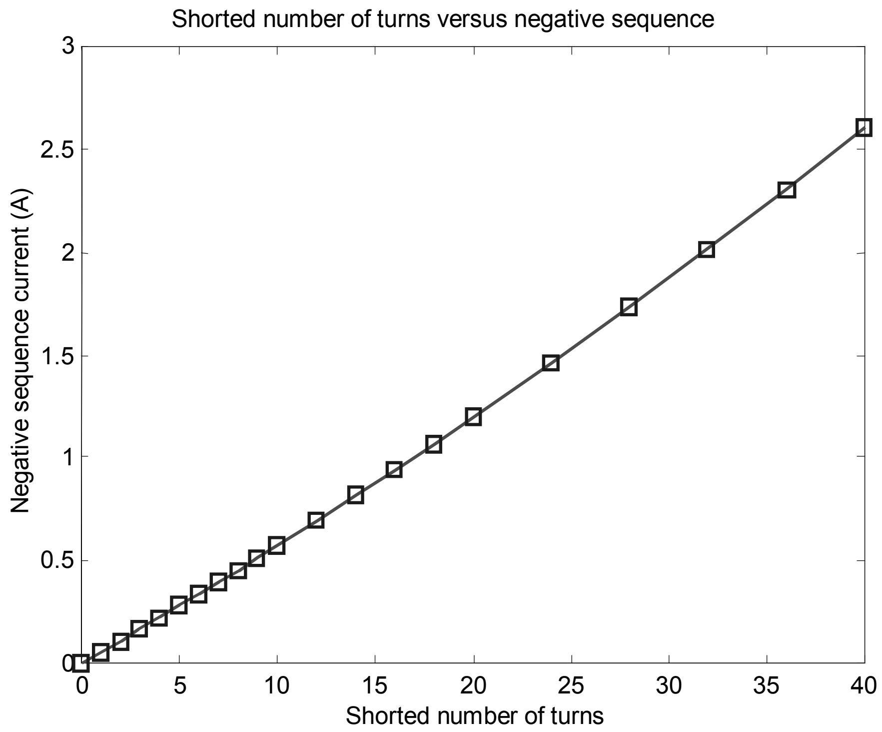


Figure 4. 20: Negative sequence current versus shorted number of turns for direct short circuit (*r_ext* = 0).

Following result are obtained from simulation by using 0.2 Ohm parallel resistance across shorted turns to limit current.

n	I_n (A)	I_{shw} (A)	I_{r_ext} (A)	I_p (A)	I_{as} (A)	I_{bs} (A)	I_{cs} (A)
1	.006	2.45	4.33	2.697	2.703	2.698	2.689
4	.063	11.93	14.23	2.701	2.761	2.688	2.653
8	.190	20.55	23.03	2.702	2.887	2.660	2.570
12	.356	26.44	29.09	2.704	3.049	2.630	2.468
16	.543	30.74	33.60	2.709	3.239	2.610	2.363
20	.750	34.04	37.13	2.717	3.452	2.609	2.262
24	.972	36.65	39.97	2.732	3.686	2.633	2.172
28	1.209	38.79	42.37	2.754	3.938	2.690	2.095
32	1.457	40.57	44.41	2.785	4.210	2.785	2.035
36	1.717	42.07	46.19	2.825	4.502	2.920	1.991
40	1.987	43.37	47.77	2.877	4.814	3.100	1.966

Table 4. 2: Currents versus shorted turn numbers (*r_ext* = 0.2 Ω)

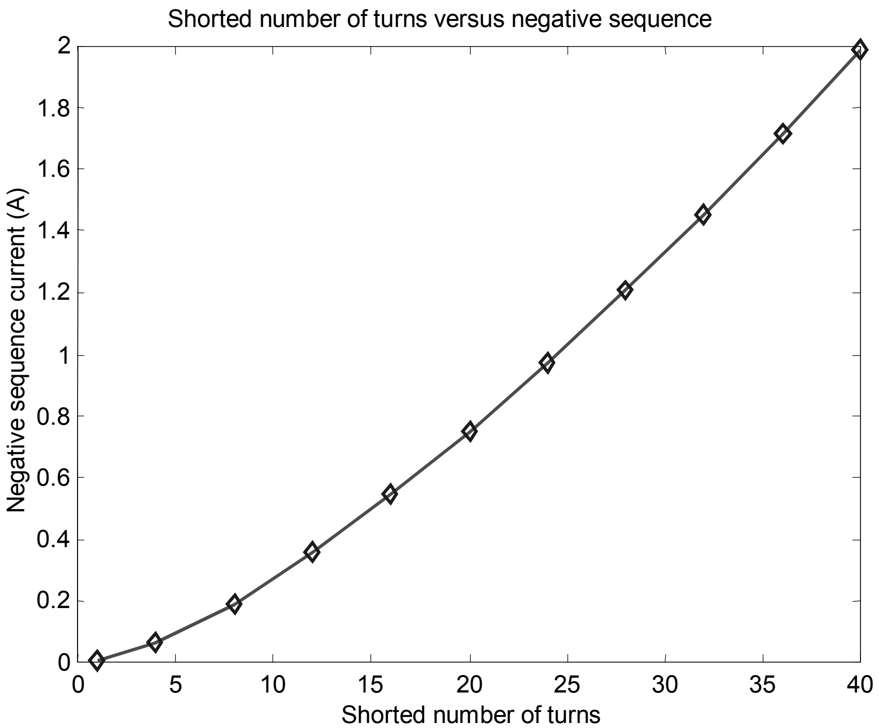


Figure 4. 21: Negative sequence current versus shorted turn. Short circuit current limited by 0.2 Ohm resistance.

4.5.2 Temperature effect on one phase

By assuming that only one phase of motor has been heated (by applying supply unbalance), resistance part of that phase will increase with temperature. The following results are obtained by putting an extra resistance in series with stator phase *as*. Maximum resistance r_{ext} is 0.63 ohm which is equivalent to a 37.5 ^{0}C increase in phase temperature.

I_n is linearly proportional to r_{ext}.

$$I_n \propto r_{ext} \tag{4.75}$$

r_{ext}	.02	.04	.06	.08	.1	.12	.14	.16	.18	.2	.3	.4	.63
I_n mA	2.3	4.7	7.2	9.7	12.2	14.7	17.2	19.7	22.2	24.6	37	49.5	77

Table 4. 3: Negative sequence current versus extra resistance

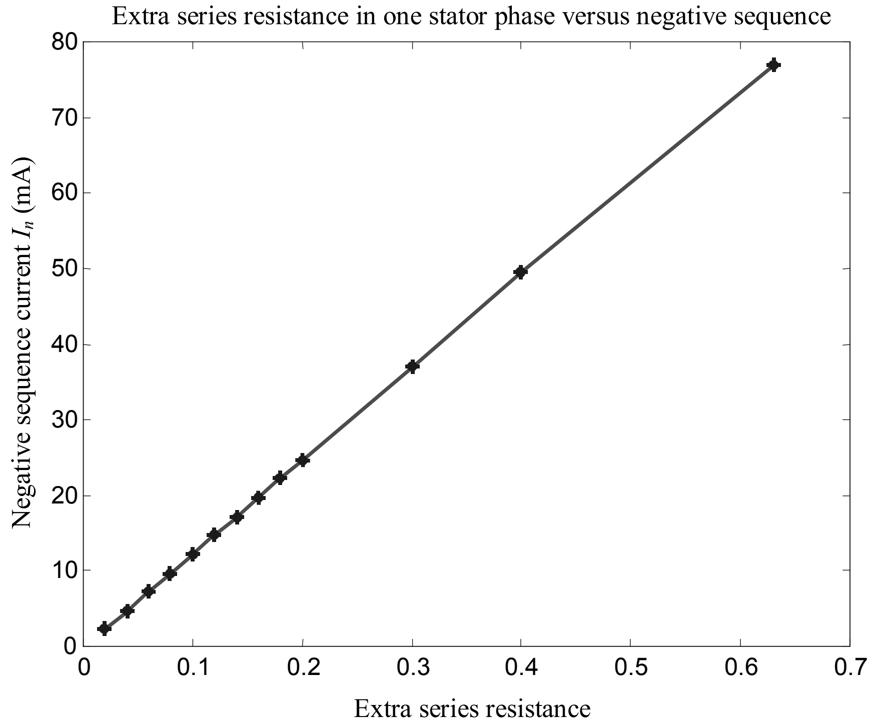


Figure 4. 22: Negative sequence current versus extra series resistance

The relation between r_{ext} and negative sequence current can be defined from the analysis of Chapter 2, Section 2.4.3 as

$$I_n = \frac{r_{ext}}{3}\frac{I_{sp}}{Z_n + r_{ext}/3} \qquad 4.76$$

where $Z_n = \left| r_s + \frac{r_r}{2-s_p} + j(X_s + X_r) \right|$, and if $Z_n >> \frac{r_{ext}}{3}$ the above equation can be simplified to

$$I_n = \frac{r_{ext}}{3}\frac{I_{sp}}{Z_n} \qquad 4.77$$

under balanced supply conditions.

The relation between negative sequence and shorted number of turn n is discussed in Chapter 6 in terms of short circuit winding current. Simulation results show that short circuit winding current I_{shw} is equal to

$$I_{shw} = \frac{3}{2}\frac{V_{sp}}{Z_s} \qquad 4.78$$

where V_{sp} is applied positive sequence voltage and $Z_s = |r_s + jX_s|$. By putting Equation 4.78 into 6.9 negative sequence current can be obtained in term of shorted turn number as

$$I_n = 0.25\frac{n}{N}I_{shw} = \frac{3}{8}\frac{n}{N}\frac{V_{sp}}{Z_s} \qquad 4.79$$

relation between r_{ext} and n can be defined from Equation 4.75 and 4.77

$$\frac{r_{ext}}{n} = \frac{9}{8}\frac{1}{N}\frac{Z_n}{Z_s}\frac{V_{sp}}{I_{sp}} \qquad 4.80$$

By using simulated motor parameters that are given in Appendix D and rated voltage 240 and full-load current 2.7 A, Equation 4.80 can be simplified to

$$r_{ext} = 0.69n \text{ ohms} \qquad 4.81$$

From simulation result $r_{ext} = 0.63n$ is obtained which is quite near to estimated relation.

Hence a thermal increase in resistance (due to current unbalance) of one phase by 0.69 ohm relative to the other windings will give the same effect as a one turn short circuit. For the motor used, this is a 17% increase, which is equivalent to a 42.5 ^{0}C increase in temperature in one winding over the other windings.

4.6 Conclusion

Two models of analysing the asymmetrical induction motor have been developed. The developed models are based on the general machine parameters so that there is no need to know detailed motor geometry or physical layout of the windings. The new parameters with asymmetry have been presented. By using the presented method, inter-turn faults can be easily simulated. Simulation results have been fed to the fault detection algorithm to monitor the motor condition.

A stator fault can also be simulated by putting extra resistance in series with one stator winding of the machine. The models presented can handle this method of introducing unbalance into the motor.

Relation between shorted number of turn and negative sequence current is presented. Results show that negative sequence current and shorted number of turns are linearly dependent. The results obtained for short circuit under limited short circuit current (by external resistance) reveal that sensitivity will be depend on shorted number of turn and short circuit current. Temperature effect on one phase (unbalance heating) has been analysed, and shown not to cause effects greater than the effect of on single turn.

Chapter 5

Hardware and Software Design

5.1 Introduction

The stator fault detection algorithm has been discussed in Chapter 3. In this chapter, hardware and software design for implementation of the algorithm will be discussed. Implementation of the algorithm has been carried out using two different hardware configurations. The first system used a Digital Signal Processor (DSP) circuit board and PC. In this voltage and current signals were sampled and pre-processed in DSP then transferred via serial communication to the PC for final processing and display. In a second configuration, a data collection Plug-in card and PC were used. The signals were sampled by the Plug-in card and loaded into PC memory for processing.

The hardware and software for each configuration will be discussed in separate subsections. Signal conditioning, which is mainly the same for both configurations, will be discussed first.

5.2 Signal Conditioning

To implement the stator fault detection algorithm, three phase voltages and three phase (line) currents are measured. The signal conditioning unit, which converts input signal to a level suitable for the A/D, the same for both DSP-PC and Plug-in-card PC configurations. For DSP-PC, signals are shifted to 0-5V level for input to the A/D converter.

5.2.1 Voltage Measurement

The voltage measurement circuit measures motor phase voltages. The three phase voltages are fed to three differential amplifiers through high resistances to limit the current flow from the supply. The voltage signals are similarly adjusted to 0-5V for DSP and $\pm 5\,\mathrm{V}$ for Plug-in card. The differential amplifiers are non-inverting amplifiers which can reduce common mode interference. The voltage signals are then isolated by followers which draw very little current from the voltage measurement circuit and give significant output for the A/D converter.

The output voltage of unit for one phase is

$$v_{a\_out} = -\frac{R_2}{R_1} v_a + \frac{R_1 + R_2}{R_1} \frac{3R_3}{3R_3 + R_1} v_n \qquad 5.1$$

for $R_2 = 3R_3$

$$v_{a\_out} = -\frac{R_2}{R_1}\left(v_a - v_n\right) \qquad 5.2$$

where $v_n = \frac{v_a + v_b + v_c}{3}$ is any zero sequence component present in the supply. For a balance supply $v_a = v_b = v_c$, and v_n is zero.

As it can be seen from this analysis under the supply unbalance condition measurement voltages are not phase voltages any more, but phase voltages minus zero sequence voltage. In Chapter 3 to find positive and negative sequence voltages, the sampled voltages first have been transferred from 3-to-2 phases, this transformation also eliminate zero sequence voltage.

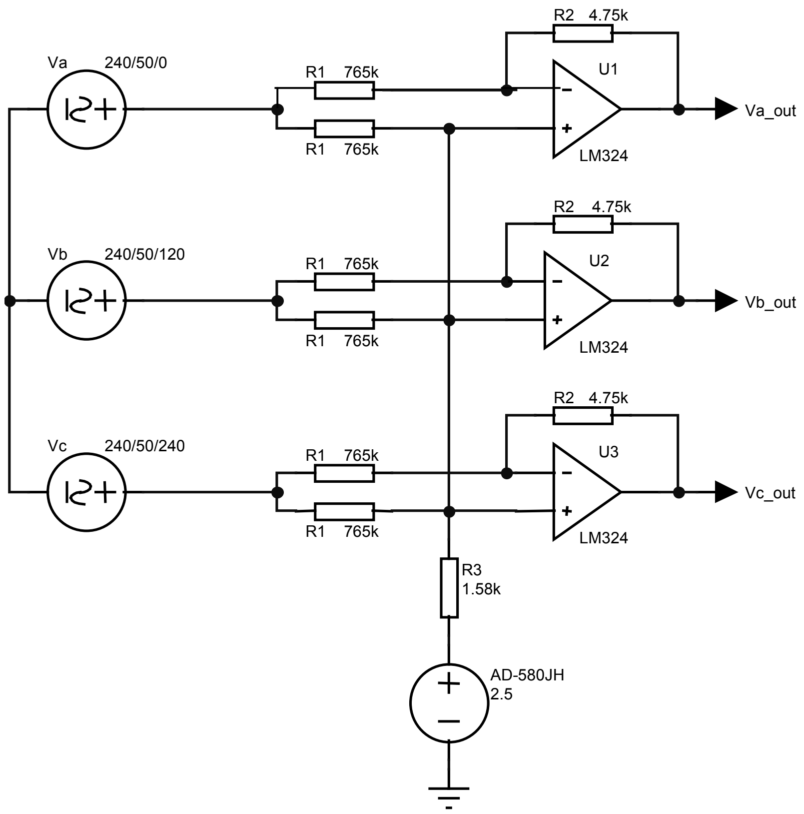


Figure 5. 1: Voltage conditioning unit

5.2.2 Current Measurement

The supply currents drawn at the motor terminals are measured by the LEM's LTA series Hall effect current transducers LTA50P/ST1. Its nominal output current is 1mA per primary ampere turn (the turns ratio is 1/1000) and the peak current range is 50A. To increase sensitivity, 3 turns and 5 turns of motor cable was passed through the core of the transducers for the DSP-PC and Plug-in-PC configurations respectively. This will give output in the range of ± 2.5V and ± 5V peak to peak voltage for peak motor currents of

10A. For DSP-PC, A/D converter in the ADMC200 motion coprocessor requires input analogue signal in range of 0-5V, the current signals, which are ± 2.5V peak to peak signals, must be shifted up by 2.5V to generate current signals in the range of 0-5V. The current signals are finally buffered by followers which draw very little current from the current measurement circuit and give the required voltage output for the A/D converter.

5.3 DSP and PC Configuration

The DSP part of hardware system was designed by Y. Liu [74] for his research on shunt active filter controller which is built around the Analog Device's ADSP-2101 [75] digital signal processor. The controller circuit includes an Analog Device's ADMC200 motion processor, peripheral interface circuit to the DSP, and phase locked loop (PLL) circuit. Voltage and current measurement units and peripheral interface between DSP serial output and PC serial input have been added for this project.

5.3.1 Hardware Configuration

The digital signal processor (DSP) unit is configured with an Analog Device's ADSP-21xx EZ-ICE Emulator attached on an EZ-LAB board. The emulator can be connected to a PC computer through a serial communication port or parallel port. This provides an easy way to program, debug and monitor the DSP unit. The emulator and the EZ-LAB board comprise a DSP system with its own program and data memory. A user interface connector is available which enables peripheral circuits to be attached.

5.3.1.1 Digital Signal Processor Unit

The ADSP-2100 Family processors are single-chip microcomputers, which are optimised for digital signal processing and other high-speed numeric processing applications. The ADSP-2101 includes computation units, data address generators and program sequencer. The ADSP-2101 combines the core DSP architecture with 1K data memory, 2K program memory, one interval timer and two serial communication ports [75].

The ADSP-2101 is a 16 bit fixed point DSP processor, and operates at 20 MHz with a 50 ns instruction cycle time. The internal program memory address bus (14 bits) and data memory address buses (14 bits) are multiplexed to an external address bus. Similarly, the internal program memory data bus (24 bits) and internal data memory data bus (16 bits) are multiplexed to an external data bus. This enables program memory and data memory/peripheral devices to be expanded externally up to maximum 16K respectively. The program and data memories are not in the same address space. They are independent of each other and can be selected with different control signals.

The EZ-ICE Emulator is a small, self-contained computer system. It emulates an ADSP-21xx DSP processor while providing information and control over the execution of the DSP program which is not available if the processor chip alone is used [76]. By physically placing the EZ-ICE Emulator in the target system DSP socket, the emulator can function as a DSP chip. It can start and stop DSP program execution at will, allow software break points, and examine and modify the internal state of the DSP's registers and memory.

The EZ-ICE Emulator system board contains an ADSP-2101 DSP device, an 8031 microcontroller, overlay memory RAM, and glue logic. All operations except the actual execution of DSP codes are managed by the 8031 microcontroller. The host PC computer is connected to the EZ-ICE system board using an RS232 serial connection. When the host interface program is executed on the host PC, it presents a user interface providing an easy access for programming and debugging.

The EZ-LAB is a small ready-made target board for laboratory uses. It comprises an ADSP-2101 DSP processor hosted in a socket, a program memory EPROM 27512, a four channel 8 bit digital to analogue converter (DAC), a CODEC (COder and DECoder) with its support circuits connected to an analogue input and an analogue output, a serial port connector and a user interface connector. Figure 5.2 is a block diagram of EZ-LAB board.

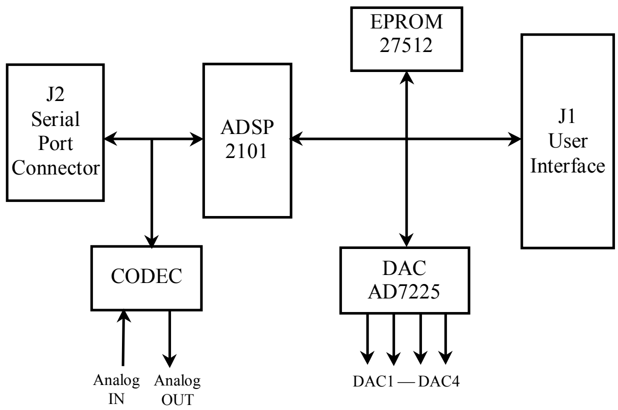


Figure 5. 2: Block Diagram of EZ-LAB Board

The EZ-LAB is a small complete application system on which many DSP application experiments can be carried out. With an EZ-ICE Emulator placed in the ADSP-2101 socket, a small development system is provided to facilitate programming, debugging and testing. Therefore, it is vary convenient to use. All the peripheral hardware is connected to the user interface on the EZ-LAB board.

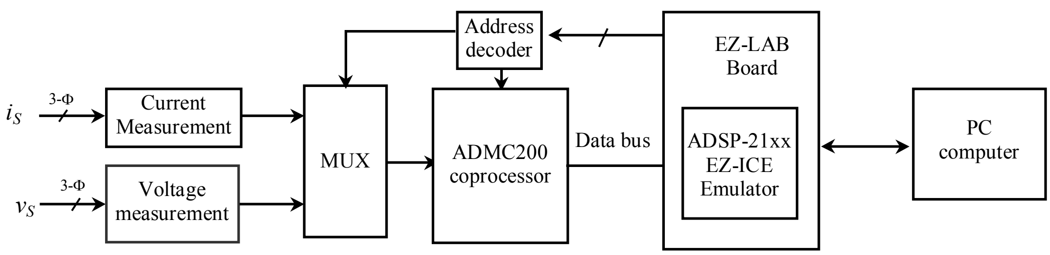


Figure 5. 3: Block Diagram of Stator fault detection hardware

5.3.1.2 Signal Sampling Using the ADMC-200 Coprocessor

The data acquisition is implemented with the separate motion coprocessor chip ADMC-200. The data acquisition collects all required voltage and current information for the stator fault detection algorithm. The DSP processes these data based on stator fault algorithm.

The ADMC200 is a specially designed motion coprocessor by Analog Devices. It comprises a 12-bit A/D converter, a PWM timer and a vector transformation unit and it is designed to interface with either microprocessor or digital signal processor very easily. The motion coprocessor ADMC200 A/D converter is the only feature used. The voltage signals and current signals are fed to the ADMC200 via a multiplexer and under DSP control, the ADMC200 converts the currents and voltages signals to data.

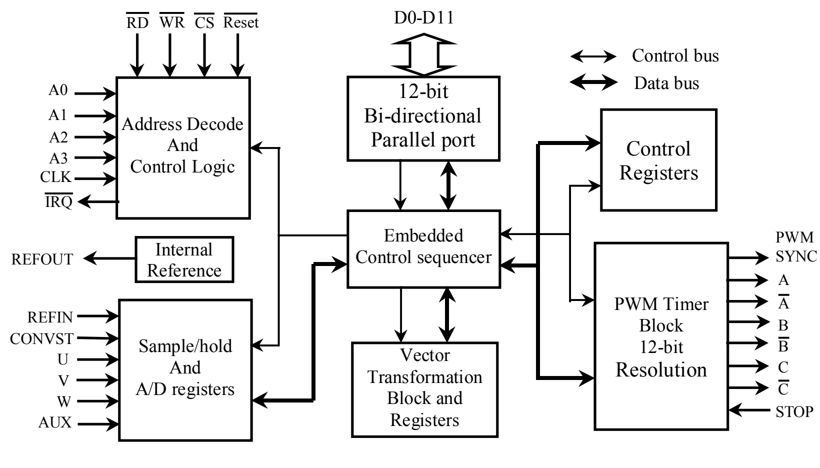


Figure 5. 4: ADMC-200 Functional Block Diagram

The Address Decode and Control Logic block along with the 12-bit Bi-directional Parallel Port provide all signals which are used to interface with a microprocessor or digital signal processor. The ADMC-200 is operated by writing commands to the control registers. The DSP can acquire all the necessary information by reading all the internal registers. The four-bit address bus provides up to 16 different register accesses. An interrupt signal can be driven active when the ADMC-200 finishes some operations, for example, on the completion of AD conversion.

In the analogue input block, there is an 11-bit successive approximation A/D converter with 2's complement output data. The analogue input range is 2.5V with a ±2.5V offset as defined by reference input. The A/D conversion time is determined by the system clock frequency, which can range from 6.25 to 12 MHz. 10 MHz was used. At the input stage to the A/D converter, there is a four-channel simultaneous sample and hold amplifier that allows four channels to be sampled and held simultaneously and then sequentially digitised. The sample and hold circuit acquisition time is 20-system clock cycles ($2\mu s$ at 10 MHz), and is independent of the number of channels sampled. The conversion time for each channel is 40 system clock cycles ($4\mu s$). The A/D conversion is initiated via the CONVST pin and an interrupt signal at the IRQ pin indicates the A/D conversions are complete, with the data stored in the output registers.

5.3.1.3 Interface between DSP and PC

The ADSP-2101 has two synchronous serial ports: SP0 and SP1. In order to communicate with the PC's asynchronous RS-232 port, some interface circuitry and protocol have to be added.

Interface MAX202 is used to change voltage level between TTL and RS-232. The MAX202 transceiver is design for RS-232 communication interface where $\pm 12\,\text{V}$ supplies are not available. On board charge pumps convert the +5V input to the $\pm 10\,\text{V}$ needed for RS-232 output level. There are five pins for each synchronous port of the DSP: serial clock; receive and transmit frame synchronization; and receive and transmit data. The transmit data pin is connected to RS-232's RX (receive) through MAX202, as shown in the diagram below.

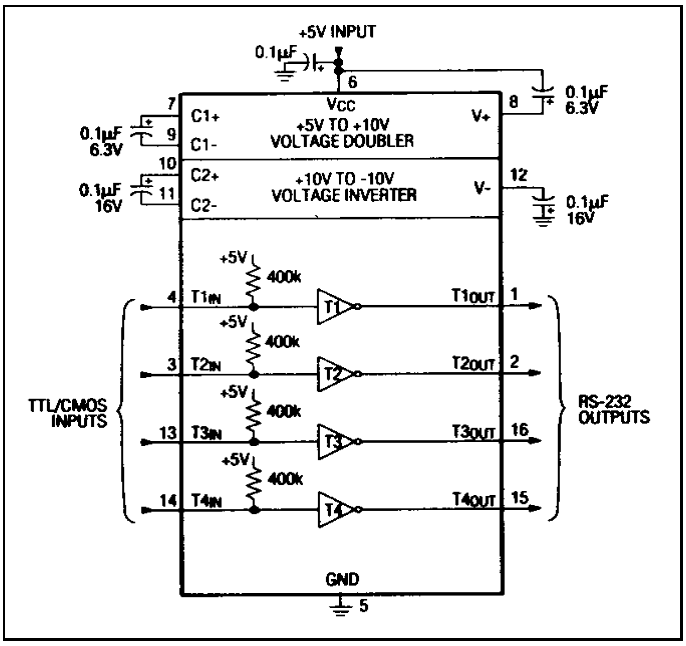


Figure 5. 5: Interface between DSP and PC

5.3.2 Software Configuration

The system software comprises two parts. The first one is written in Microsoft Visual BASIC and resides in the PC and controls the communication between the DSP and PC. It does the final processing of the data sent by the DSP and display results in numeric and graphic mode. Visual BASIC is good for its interface functions. The other part of the software resides in the program memory of the DSP. This part of the program is written in assembler code for fast pre-processing of sampled data.

5.3.2.1 Pre-processing: Assembler Programs

The pre-processing program is written in assembly language and debugged in the EZ-ICE 2101 emulator. The structure of program has an interrupt driven structure. There is an initial program which is executed whenever the DSP is reset, and then the DSP goes into an idle state awaiting interrupts. When an interrupt occurs the DSP responds to that interrupt, and executes a corresponding subroutine, at the end of which the DSP returns to the idle state waiting for next interrupt. There are six interrupts in ADSP2101, only three of which are used, as shown in the figure below.

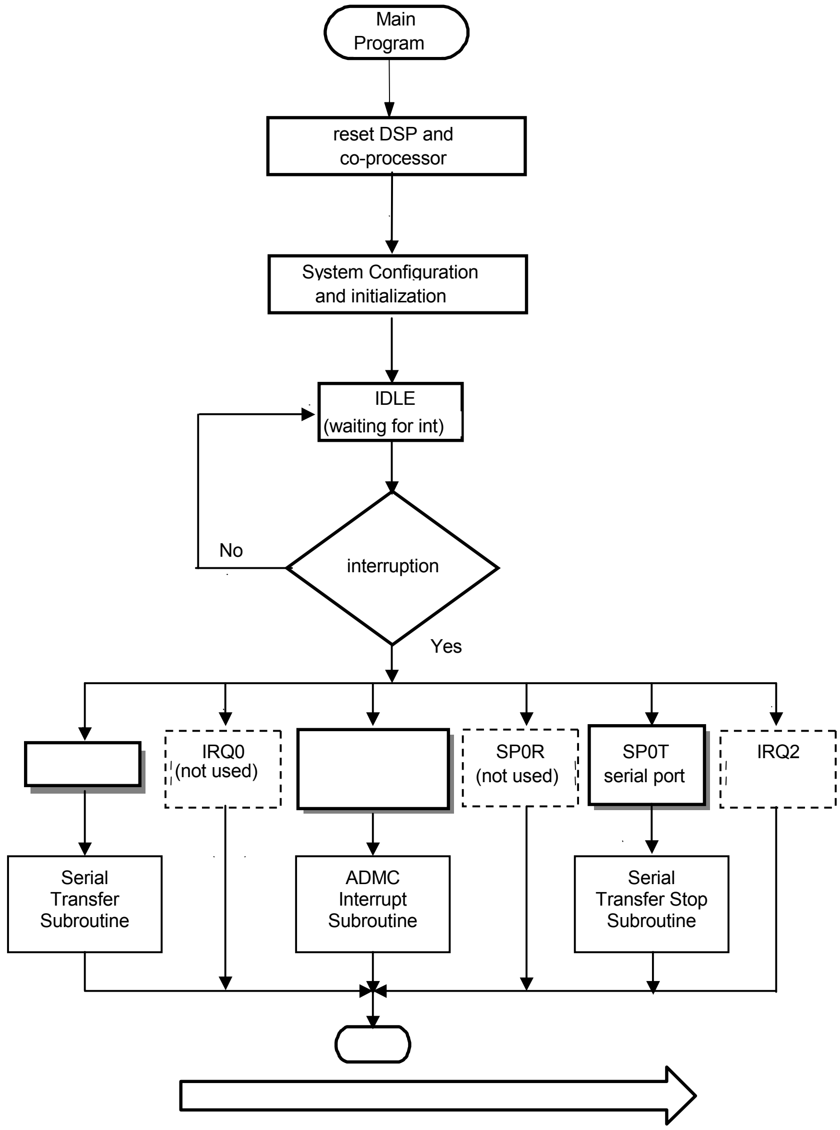


Figure 5. 6: Structure of Program

5.3.2.1.1 System Configuration and Initialisation

The program starts with system initialisation that sets up the program running environment such as control register configurations, control variables and algorithm control parameters etc. The system control variables and constants must first be defined. The DSP data memory must be allocated for these variables, constants and data buffers required by the algorithm being used. In addition, the programmable devices ADSP-2101 and ADMC-200 must be configured to meet the particular requirement. The control registers must be set to appropriate values for the real time processing. All of these preparations for the stator fault detection programming are made by the system configuration and initialisation program. Figure 5.7 is a block diagram (not flow chart) of the system configuration and initialisation.

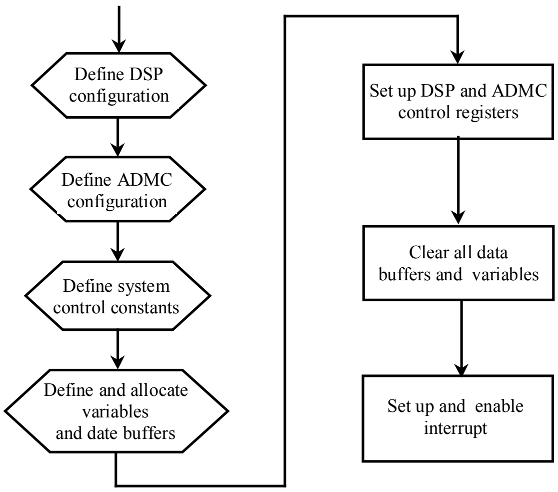


Figure 5. 7: System Configuration and Initialisation

The DSP configuration specifies the memory interface, timer, serial communication ports, wait-state, and interrupt system settings. The ADMC-200 configuration specifies the operation mode of A/D converter.

The constants are the parameters that used in the program. All the constants, variables and data buffers must be defined and allocated in the DSP data memories.

A short system initialisation program writes all the system configuration values to the DSP and the AMDC-200 registers and clears all the data buffers and variables for preparation of interrupt motivated control programmes. After system initialisation, the program is in waiting state for the interrupts.

5.3.2.1.2 Signal Sampling and Computation: ADMC Interrupt

This interrupt subroutine is the core program of the stator fault detection algorithm. On the completion of the A/D conversion, an edge based interrupt request signal is generated. This indicates that the voltage or the current data are available in the ADMC registers. The program itself contains two subprograms. One is for voltage sampling and other is for current sampling. To switch between these two subprograms two flags have been used, V_flag for voltage and I_flag for current.

If voltage data is to be sampled by the ADMC-200, the three phase voltages will be sampled for one cycle and decomposed into their positive and negative sequence component by the procedure defined in Chapter 3. After voltage sampling is completed, the external multiplexer is switched to the current, V_flag is set and all data buffers are cleared for current computation.

After voltage cycle is completed, the ADMC-200 begins to sample currents. The sampled currents will be decomposed into their negative and positive sequence component as voltages. After current sampling is completed, the multiplexer is switched back to voltage, I_flag is set and all data buffers are cleared. The main program then will wait for timer interrupt to send sequence components to PC. The assembler program is given in Appendix E.

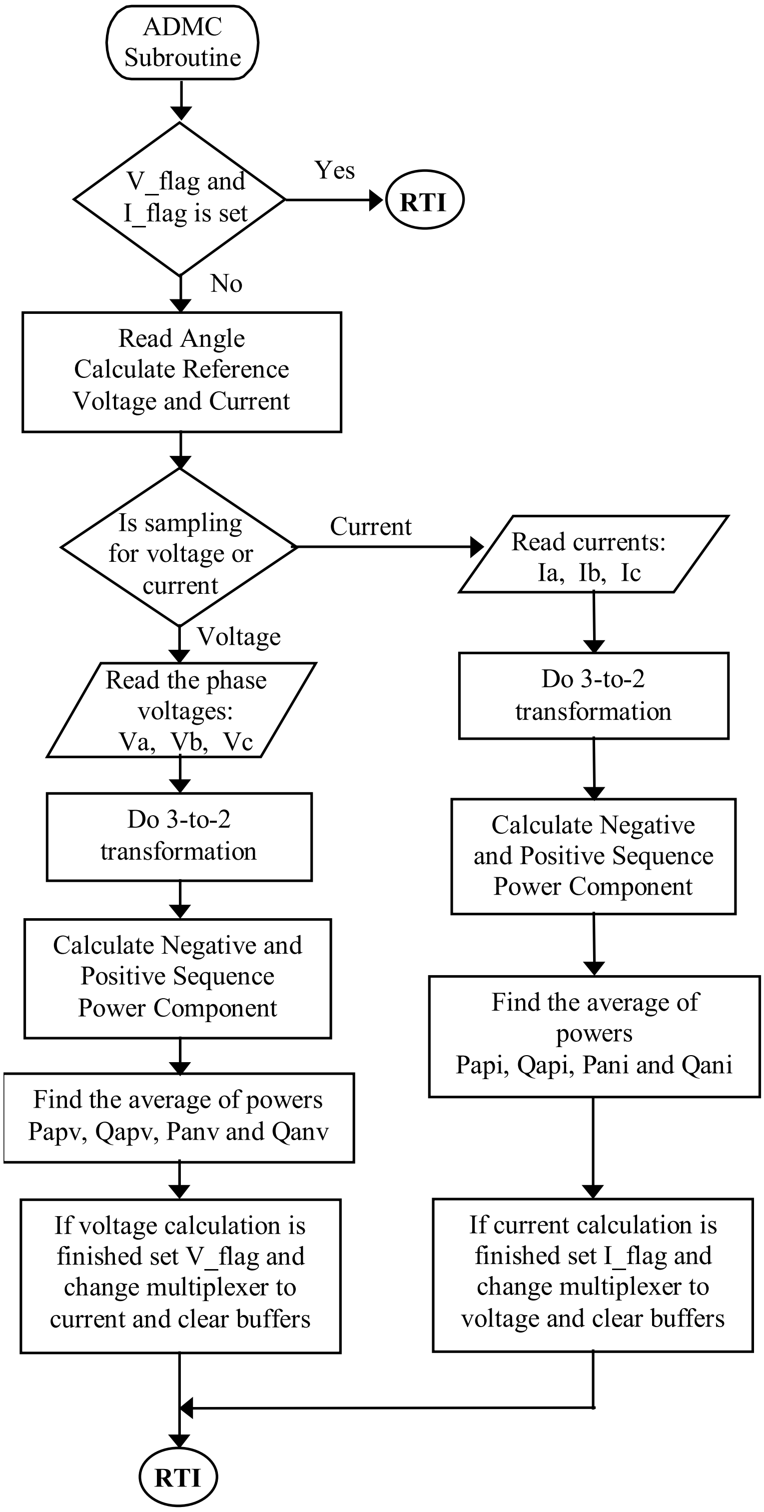


Figure 5. 8: ADMC Subroutine: Signal Sampling and Computations

5.3.2.1.3 Serial Transfer: Timer Interrupt

The TIMER interrupt provided regular interrupts, from the timer, with long interval (125ms), and provides the system with a continuous real time flowing axis, which is used for sending data (to PC).

The results of voltage and current decompositions are transferred from DSP to PC via the serial port at a rate of 19.6 kbaud. There are eight data numbers in total, four for voltage and four for current. Because these data are 16-bit words, first each of them converted to unsigned by representing each number between 0 to 2 instead of –1 to +1 range, and divided into two 8-bits data, then data length is set to eight plus one start bit and two stop bits. So that in total 16 data bytes are sent to the PC. These sixteen data are saved in Data Address Generator (DAG). Autobuffering has been used to transmit entire circular of data with an overhead of only one cycle per data word. Transfers between the SPORT and the circular buffer are automatic in this mode and do not require additional programming. The assembler subroutines for data transferring from DSP to PC are given in Appendix E.

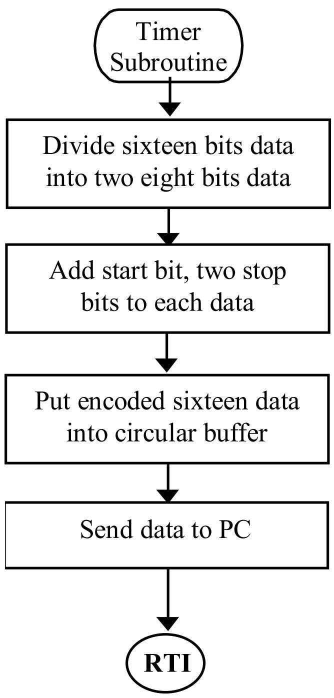


Figure 5. 9: Timer Subroutine: Data encoding and sending to PC

5.3.2.1.4 Transfer Stop: The Serial Port Transmitting Interrupt

SP0T is the serial port transmitting interrupt. After the DSP sends a group of data by autobuffering (to PC), an interrupt is generated, and its interrupt subroutine disables auto transferring of further data through the serial port (otherwise the auto-transmitting would not stop and the PC's serial port would overflow).

5.3.2.2 Final Processing: Visual Basic

The DSP is connected to a PC by serial port, and interfaced by a program written in Visual BASIC. The PC receives data from the DSP and displays the data on the screen.

Visual BASIC is a Windows based, event driven language, which has a good user interface, and provides a good display interface. Every part of the program is triggered by an event except for the main subroutine. Control buttons and display boxes are put in to 'form' routines, which are used to display result at the run time.

The diagram shows the flow of a Visual BASIC program. At the beginning of the program, variables are initialized, communication port and display forms are validated, then the program goes into a loop to update the display. A global variable stores the data received from DSP by the communication event which is a serial port interrupt.

The internal data in DSP memory (16 words) are sent by DSP to the PC every 125ms, so real time control data can be observed on the screen whilst experiments are being conducted.

The received data is then processed into the negative and positive sequence components. The stator fault algorithm that has been discussed in Chapter 3 is then applied to get supply, load and voltage, and motor fault current. Motor characterization also can be carried out at that stage on-line or data can be saved into file for later analysis. The results are displayed on the screen both in numeric and graphic mode. VB program is given in Appendix E.

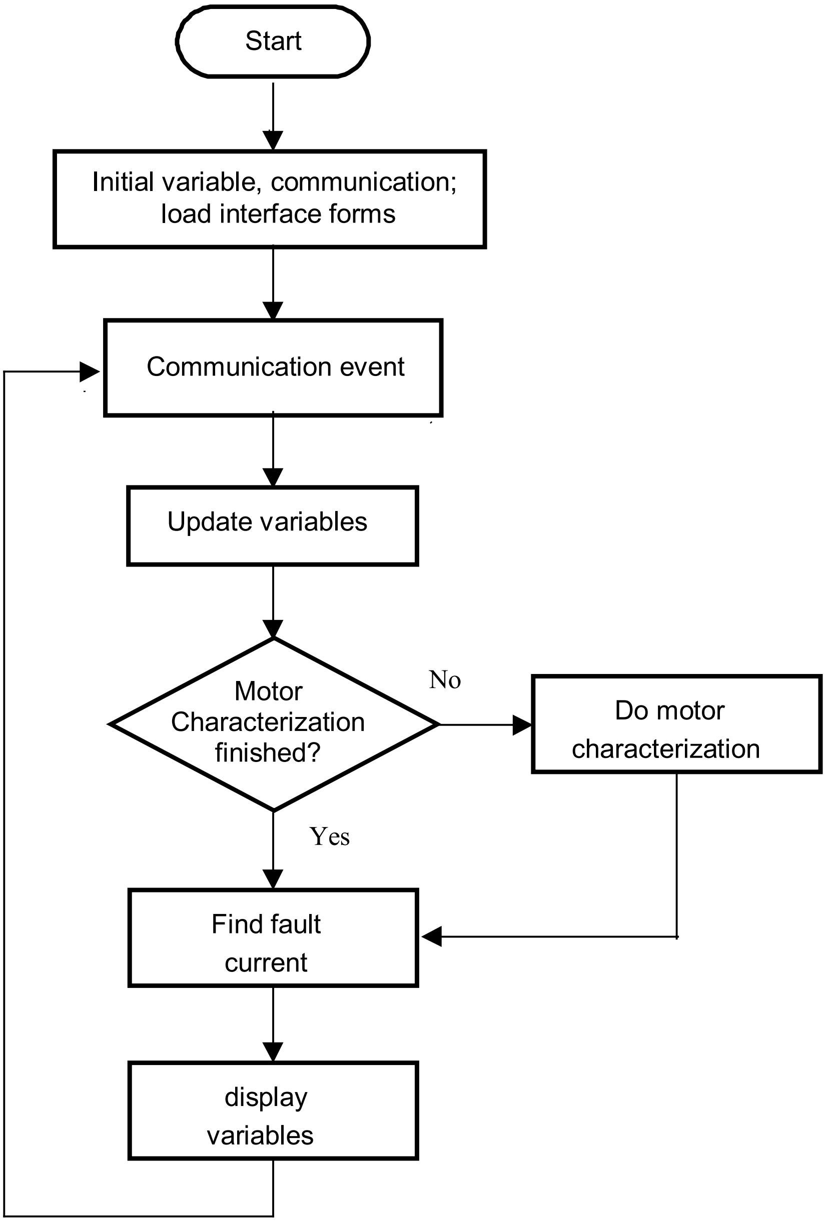


Figure 5. 10: VB Flow Chart

5.4 Plug-in-card and PC Configuration

The second configuration used in this research was a Data Acquisition Plug-in-card and PC configuration. Personal computers not only offer high speed, flexibility, adaptability, and mass memory, but also essential part of the data acquisition system. Although a PC is versatile, it can not readily accept sensor outputs because it is a digital device. Thus data conversion is required and this is achieved by using an Eagle Technology data acquisition card, PC30GS16.

Hewlett-Packard Virtual Engineering Environment (HP VEE) software is used with Data Acquisition. HP VEE is a visual programming language (graphical programming language) optimized for building test and measurement applications.

5.4.1 Hardware: Plug-in-card-PC30GS16

The PC30GS16 is high accuracy analog board for PC. It provides 12-bits resolution at a sampling rate up to 100kHz with sixteen channels. The board input range is between ± 5 volts peak-to-peak. It can sample all channels simultaneously. The board consists internally four quads of channels. Each quad consists of four channels. This board offers 300 ps uncertainty on sampling within each individual "quad" of channels and 20 ns uncertainty between all channels. This time difference in sampling between the various channels is called "aperture matching".

In addition to aperture matching, there are gain and offset errors between quads

-Gain errors in quads are less than 0.005%, and between quads 0.05%.

-Offset errors in quads are less than 0.2 LSB, and between quads 7mV.

To minimise gain and offset errors voltages are sampled by one quad, currents by a different quad.

5.4.2 Software: HP VEE Program for Stator Fault Algorithm

As mentioned before HP VEE software is used for implementation of Stator fault detection algorithm. The PC30GS16 card has a driver program available for HP VEE. The driver takes commands from the high level language and translates them into the addresses and commands required to operate under the PC30GS16. The driver consists of two DLLs, a VXD (Virtual Device Driver) and a library of HP VEE functions. The VXD handles all board functions that involve interrupts or DMA. The base DLL's provide a simple interface to the VXD as well as many other useful functions. The HP VEE library functions wrap low level calls to the DLL into more powerful, easy to use, and high level functions.

The data acquisition program written in HP VEE for stator fault detection algorithm contains several subsystem blocks (or user objects):

Initialization block, board configuration, reference signal, signal sampling, decomposition block, fault block, result display blocks

Constants, arrays, motor initial parameters, voltage and current scaling factors are defined in the initialization block.

Before starting the program, the drivers have to be loaded, functions that are supplied have to be imported and the plug-in-card has to be set. The driver and functions can be loaded by using Import Library object. Transfer mode, sampling frequency, channels to be sampled and number of cycles are set in the board configuration block.

After sampling frequency and number of cycles to be sampled have been set, reference signals will remain the same all the time. The reference signals are produced in reference signal block and saved in arrays. They have to be changed only if the sampling frequency and number of cycles are changed. Figure 5.9 shows the board configuration and reference signal block.

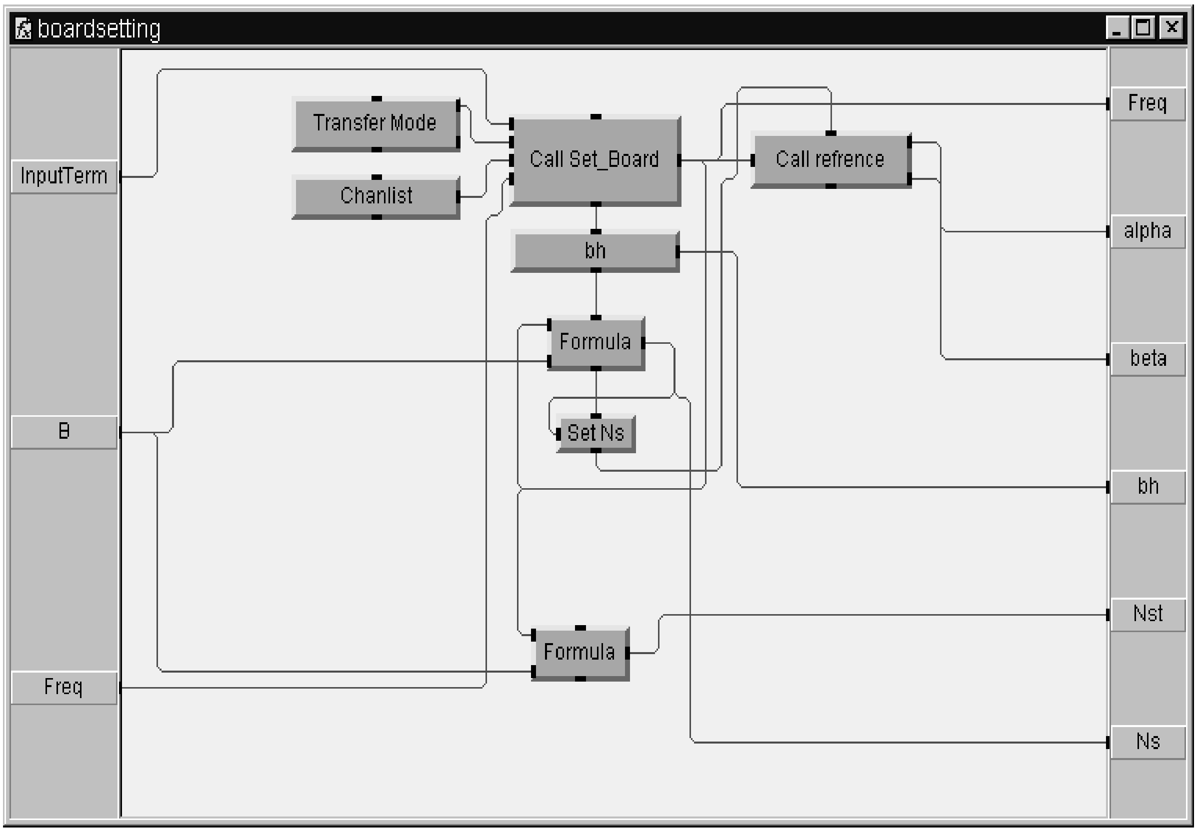


Figure 5. 11: Board setting object

These sub-blocks are loaded at the beginning of the program. After these blocks have been run the program will go to into a do-until-break loop.

Signal sampling: Sampling is carried out by using the background sampling functions. The block of sampled data can then be transferred from the card to into the memory by supported five different ways: Polled I/O, Interrupts, single channel DMA, dual channel DMA, and streaming. All of the transfer modes except polled I/O can be used for background sampling. Streaming to memory has been used, which uses a single channel DMA in a circular buffer mode. By using this mode data can be transferred without the CPU taking any action. Figure 5.12 shows signal sampling block.

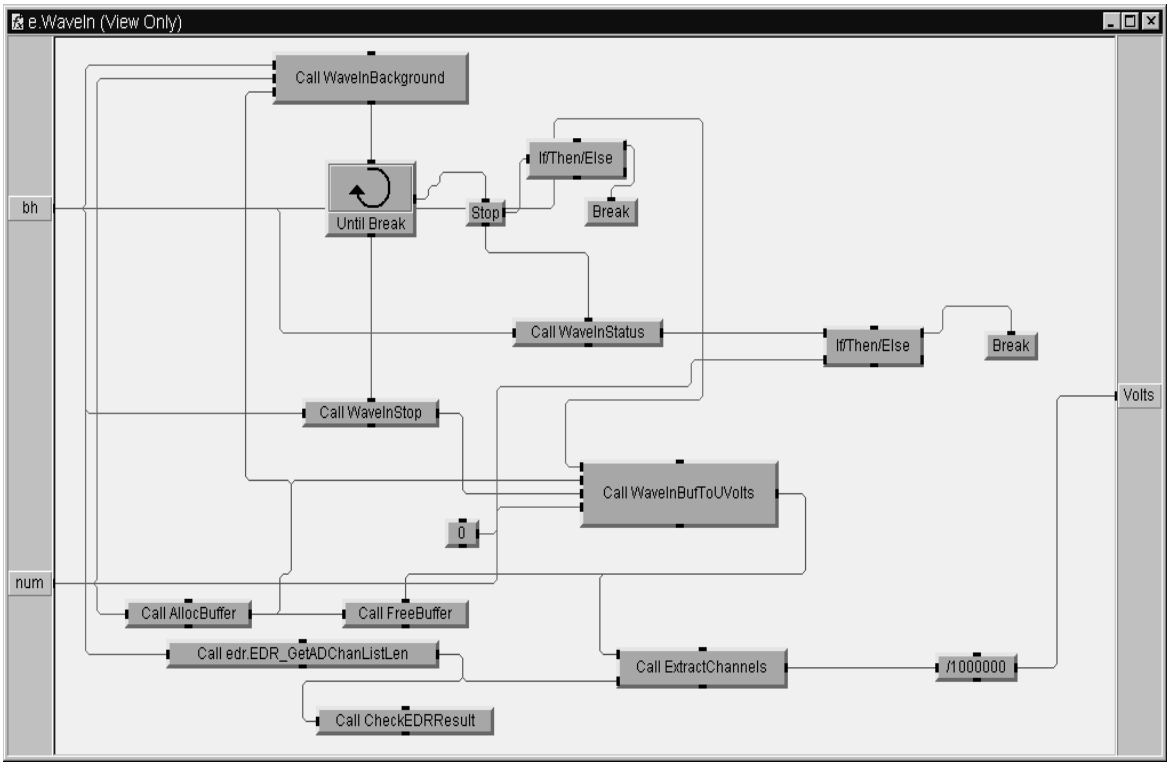


Figure 5. 12: Signal sampling object (Wavein)

Sampled voltage and current then are fed to 3-to-2 transformation block. Transferred signals are multiplied by reference signal and the powers defined in Chapter 3 are calculated. From these powers, positive and negative sequences voltages and currents are found in the sequence components block.

The final block is fault block. This block also contains several sub-block which are the temperature independent method fault current block (Imnf block), the PDT fault current block, characterization block for negative sequence inductance (Xhn block) and the characterization block for voltage and load variation (Invl block). Figure 5.13 shows the fault block.

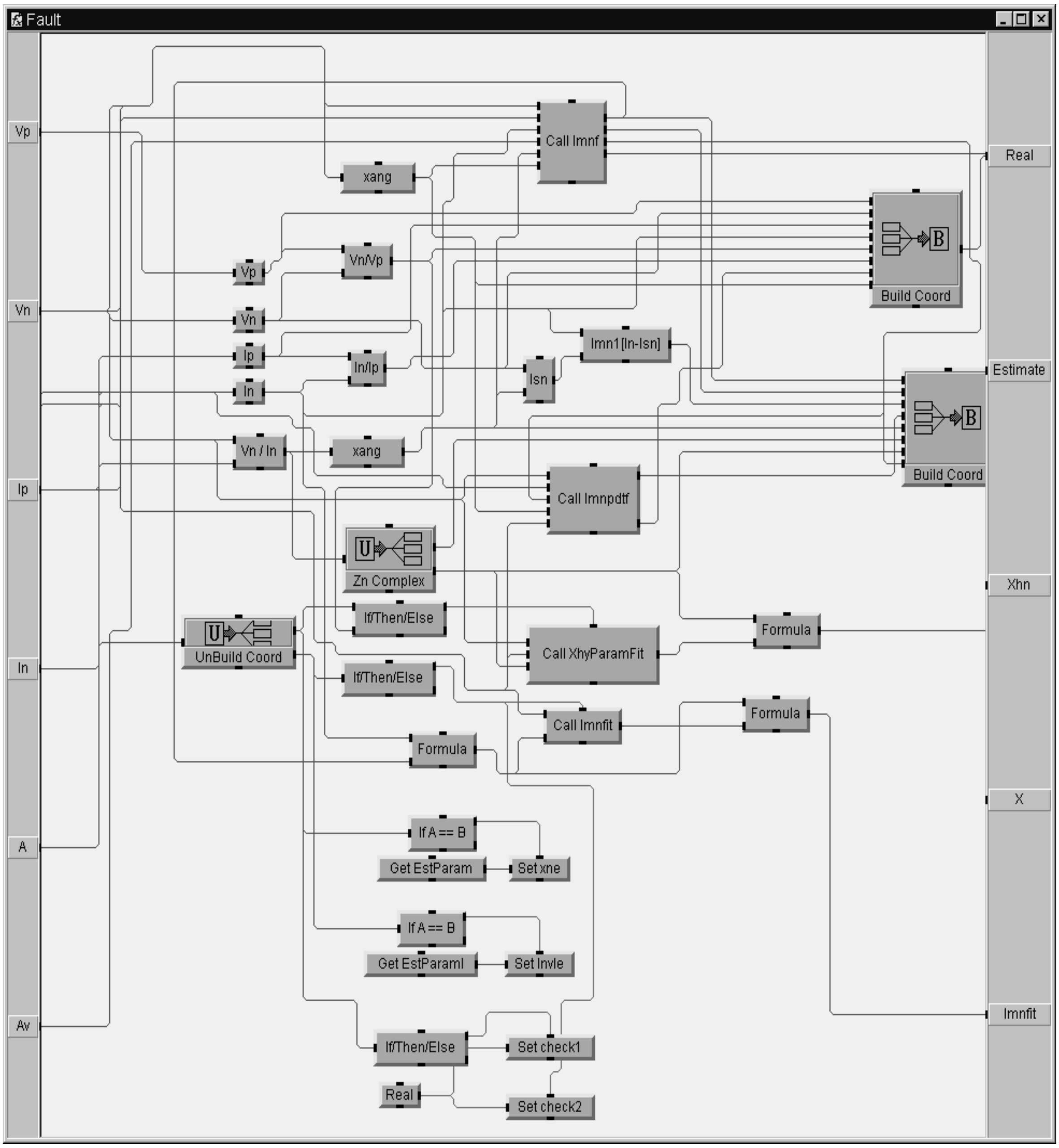


Figure 5. 13: Fault object

Overall program and its panel are shown in Figure 5.14 and Figure 5.15. Panel is user operating interface. It can be used for control and display computation results. The fault current also is displayed in an oscilloscope type display so that variation can be seen in graphical mode.

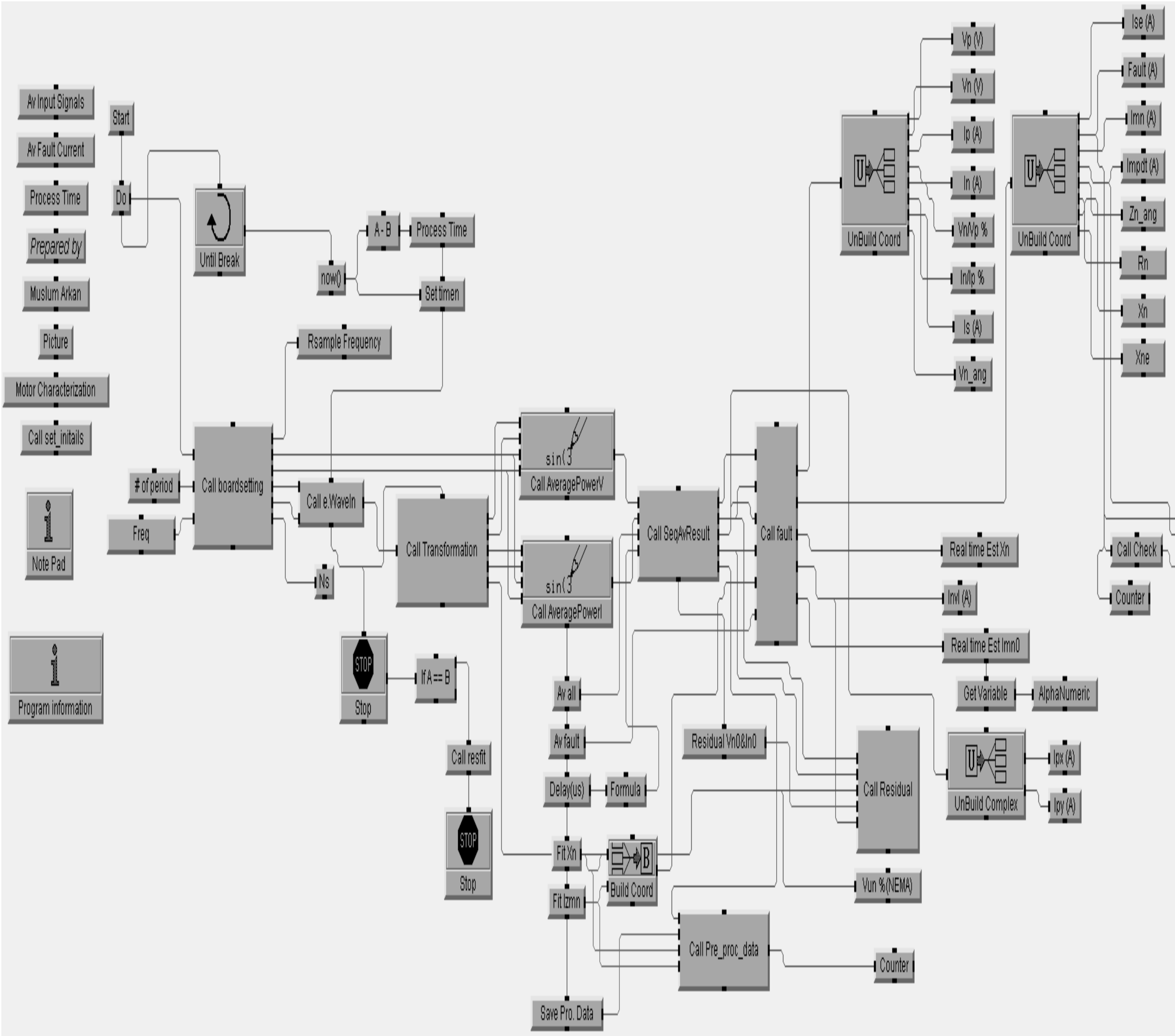


Figure 5. 14: Main program

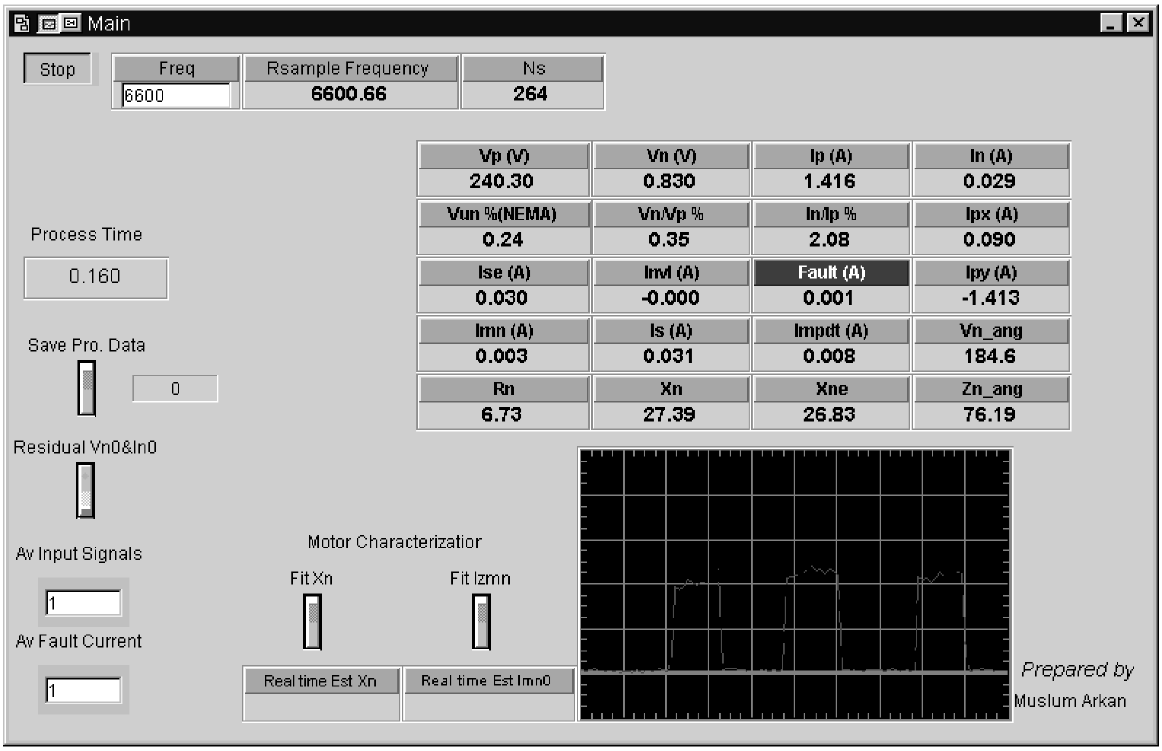


Figure 5. 15: Main program panel view

Fault diagnosis program has been debug by using known software produced signals, and currents and voltages from simulation results (Chapter 4). These signals have been used instead of signal sampling block. The results have shown that the program is bug free.

5.5 Conclusion

In this chapter, the hardware and software designs for the stator fault detection algorithm have been discussed. In order to implement stator fault detection, two different hardware configurations have been used. The discussion started with signals conditioning which was the same for both configurations. Then is followed by hardware and software design for DSP-PC configuration. The devices that were used in this configuration such as ADSP-2101 and ADMC-200 have been briefly described. Assembler program subroutines and VB program and their functions have been given. PC30GS16 card and HP VEE program that are used for second configuration have been explained.

Chapter 6

Experimental Setup and Test Results

6.1 Introduction

Experimental results for stator fault diagnosis are discussed in this chapter. The experimental setups and tested motors are introduced briefly. The test results for supply unbalance, mains voltage and load variation, and inter-turn faults under different operating conditions as well as for different degrees of fault are presented. Motor characterization parameters are given. The sensitivity of the algorithms is discussed using experimental and simulation results together.

6.2 Experimental Setup

Experiments have been conducted on several motors. Results will be given for three: A Generalized machine (4.3 HP), an ABB motor (2.95 HP) and a Reliance motor (2 HP).

Generalized machine: The stator of the machine has a stator wound in 48 slots. The ends of all 48 coils are brought out to 96 terminals symmetrically arranged in four concentric circles, so that the stator windings can be connected in either 2-phase or 3-phase form and, by suitable interconnection of the coils, for several different pole numbers. The principal details of the machine are listed in Table 6.1. The 96 terminals enable different asymmetrical stator connections to be made by means of appropriate short circuiting links. Although there are 62 turns in each coil between adjacent terminals, the machine is convenient to test turn-to-turn faults within or between windings.

Generalized Machine		
	Stator	Rotor
Number of slots	48	36
Number of coils	48	36
Number of turns per coils	62	16
Winding connection	2 parallel paths each of 496 turns, delta	Semi-squirrel cage
Line voltage	240 V	
Full load line current	10.3 A	
Horse power	4.3 HP	
Rated speed @ 50 Hz	1435 rpm	

Table 6.1: The machine is connected as 4-pole, delta connected, semi-squirrel cage induction motor.

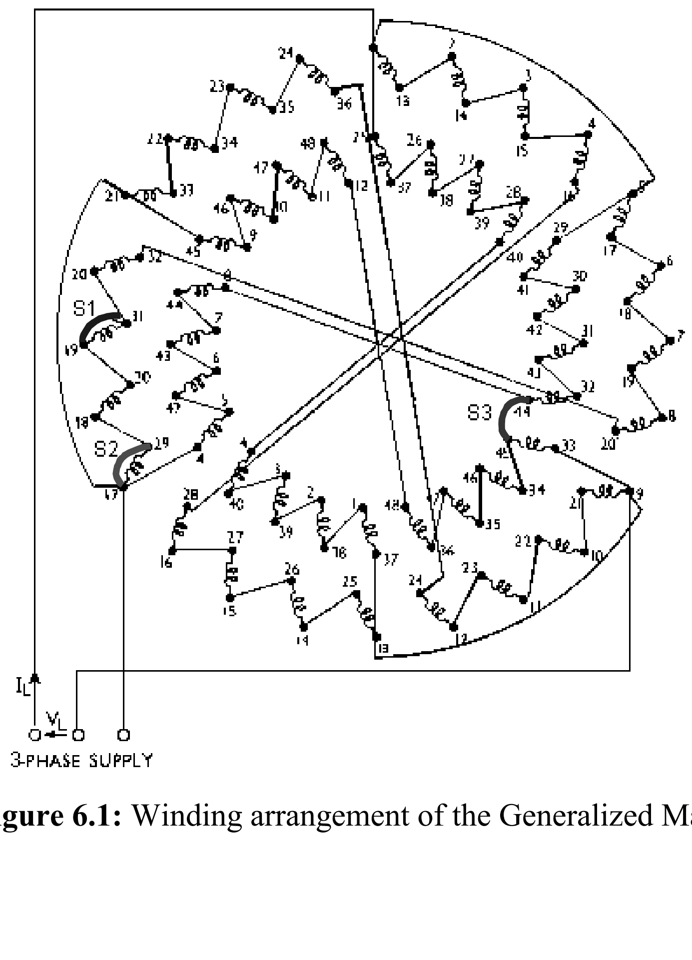


Figure 6.1: Winding arrangement of the Generalized Machine.

***Reliance Motor**:* Reliance motor has been specially wound for testing smaller inter-turn faults. Table 6.2 shows the specification of motor. The motor has 5 taps for each of two stator phases. Each tap corresponds to one turn, which allow us to short circuit 1, 2, 3, or 4 turns. The taps have been brought to a switch box where the number of shorted turns and phase can be selected.

Reliance Motor	
Line voltage	460 V
Full load line current	2.7 A
Horse power	2 HP
Rated speed @ 60 Hz	1752 rpm
Number of poles	4
Power factor	0.815
Number of stator turns	252
Rotor bars	44
Service factor	1.15

Table 6.2: The motor is a star connected, squirrel cage induction motor.

***ABB Motor**:* ABB motor has been rewound for testing inter-turn faults. Table 6.3 shows the specification of motor. The motor has 4 taps for each of two coils in the same phase. The taps are made at 1, 10 and 42 turns in each coil, which when combined allow us to short circuit 1, 9, 10, 32, 41, or 42 turns. The taps have been brought to a switch box where the number of shorted turns and coil can be selected.

ABB Motor	
Line voltage	420 V
Full load line current	5.2 A
Horse power	2.95 HP
Rated speed @ 50 Hz	1430 rpm
Number of poles	4
Power factor	0.81
Number of slots	36
Number of coils	18
Winding connection	2 parallel paths each of 252 turns

Table 6.3: The motor is a star connected, squirrel cage induction motor.

Tests were carried out by connecting a variable resistance *r_ext* between shorted terminal pairs (see Figure 6.15). The resistance allowed the fault current to be controlled and limited the short circuit current to avoid damaging the winding.

6.3 Experimental Results

6.3.1 Motor Characterization

Motor characterization for variation in supply unbalance, mains voltage, and load, is carried out in two stage as explained in Chapter 3. Firstly, the motor is characterized for supply unbalance, described by the γ parameters in Equation 3.33; and then for mains voltage and load variation as described by the α parameters in Equation 3.30. Contribution to the negative sequence current from non-linearities in the motor were measured before introducing shorts, and parameters for the semi-empirical formulae were fitted, so that the effects of supply unbalance, mains voltage and load variation on the measured fault current could be virtually eliminated by subtracting their contribution to the observed negative sequence current. Any residual negative sequence current remaining after this subtraction acts sets a background value which fault current must exceed to be detectable. *Hence the success of the characterization process sets the sensitivity threshold for stator short circuit detection*.

6.3.1.1 Characterization for Supply Unbalance

Characterization for supply unbalance has been conducted by using three variacs. The variacs allow unbalanced supply voltages to be applied to the three phases both above and below nominal voltage, in order to vary the applied negative sequence voltage $\mathbf{V}_{sn}$. Test results confirm that the healthy motor negative sequence reactance X_{hn} has to be treated as a variable quantity to eliminate the contribution of supply unbalance. Figures 6.2 and 6.3 show the variation in the measured negative sequence reactance X_n under supply unbalance at no-load for the different motors. The variation could be as much as 10%.

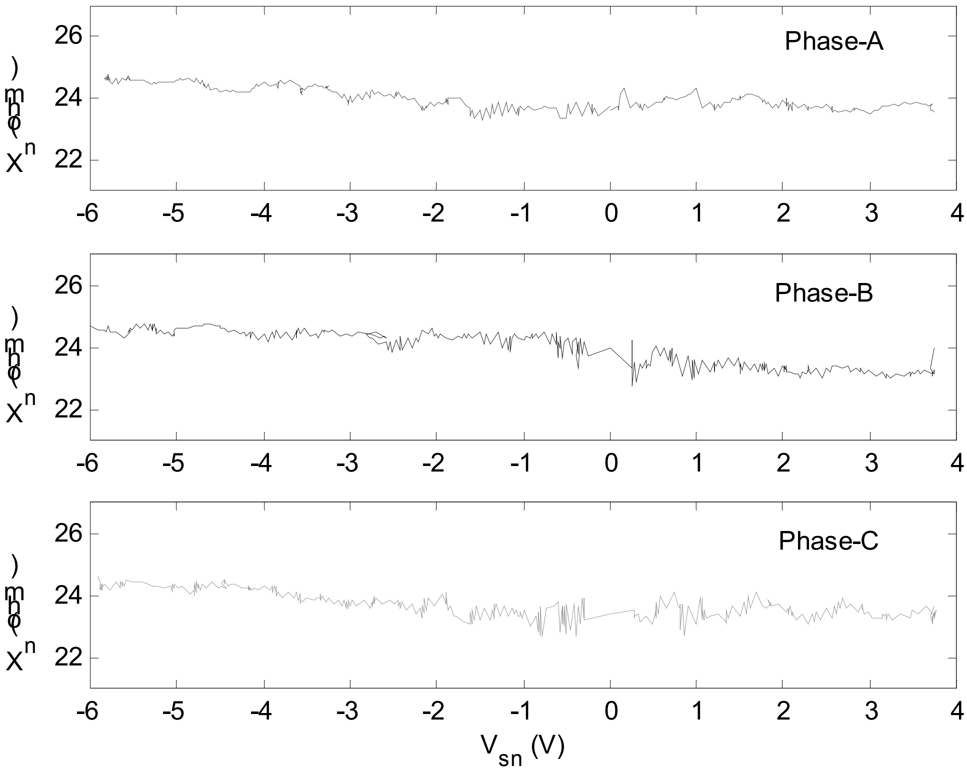


Figure 6.2: Negative sequence reactance versus $\mathbf{V}_{sn}$ for Reliance motor under supply unbalance variation (Figure 6.5). Supply unbalance has been varied between 2.5% below ($-\mathbf{V}_{sn}$) to 1.5% above ($+\mathbf{V}_{sn}$) nominal voltage for phase *as, bs,* and *cs* respectively.

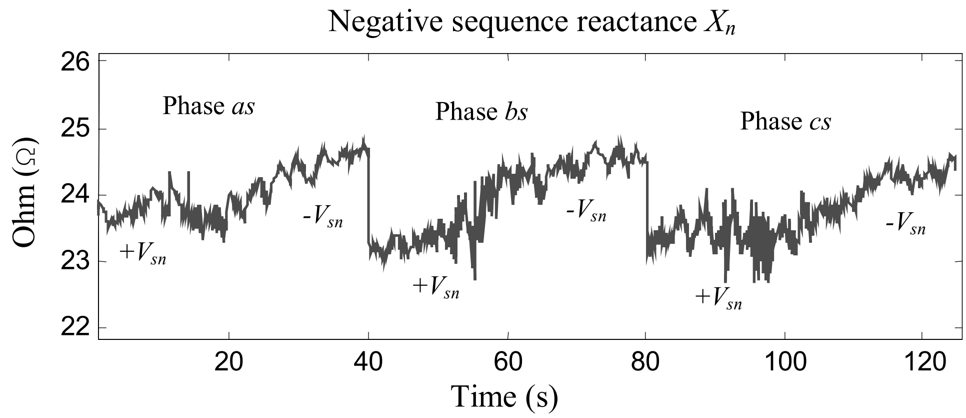


Figure 6.3: Negative sequence reactance for Reliance motor under supply unbalance variation (Figure 6.5). Supply unbalance has been varied between 2.5% below ($-\mathbf{V}_{sn}$) to 1.5% above ($+\mathbf{V}_{sn}$) nominal voltage for phase *as, bs,* and *cs* respectively.

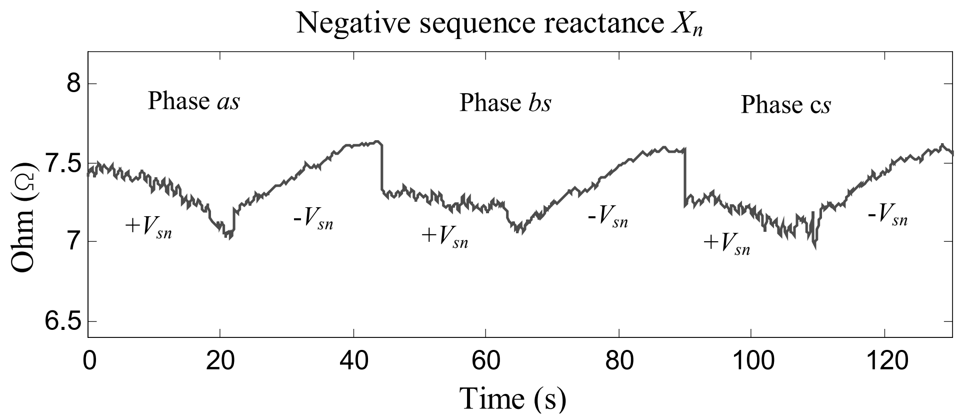


Figure 6.4: Negative sequence reactance for Generalized machine under supply unbalance variation. Supply unbalance has been varied between 4% below ($-\mathbf{V}_{sn}$) to 2% above ($+\mathbf{V}_{sn}$) nominal voltage for phase *as, bs,* and *cs* respectively.

As has been shown in Chapter 2, negative sequence impedance is nearly independent of load variation, if the motor is symmetrical (healthy). The maximum variation in the negative sequence impedance with load change for the motors tested is between 2% to 5%. Although negative sequence reactance is independent of load (slip) variation, experiments showed that with variation in supply unbalance, X_n could vary between 10% to 50%. This variation is different from motor to motor (size and manufacturer). Figures 6.5 and 6.6 show the variation in X_n under 1% supply unbalance with the load. Motors have been loaded from no-load to full-load whilst supply unbalance has been kept constant.

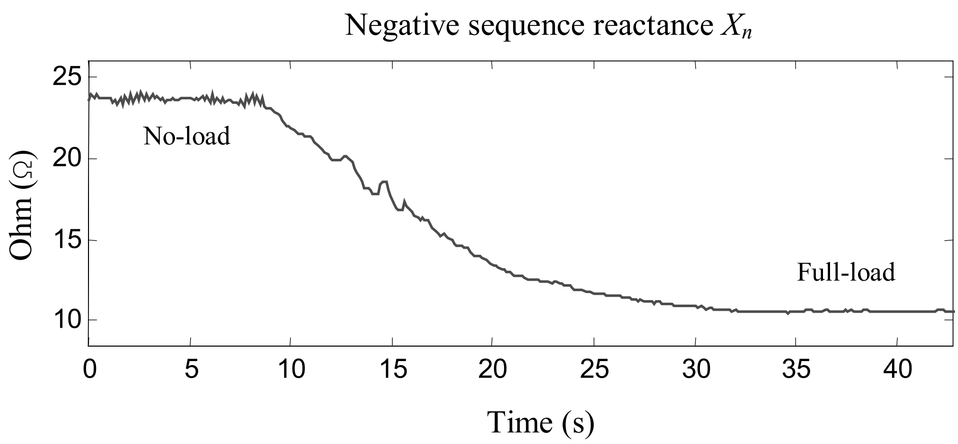


Figure 6.5: Negative sequence reactance X_n (for Reliance motor) with variation from no-load to full-load under 1% supply unbalance.

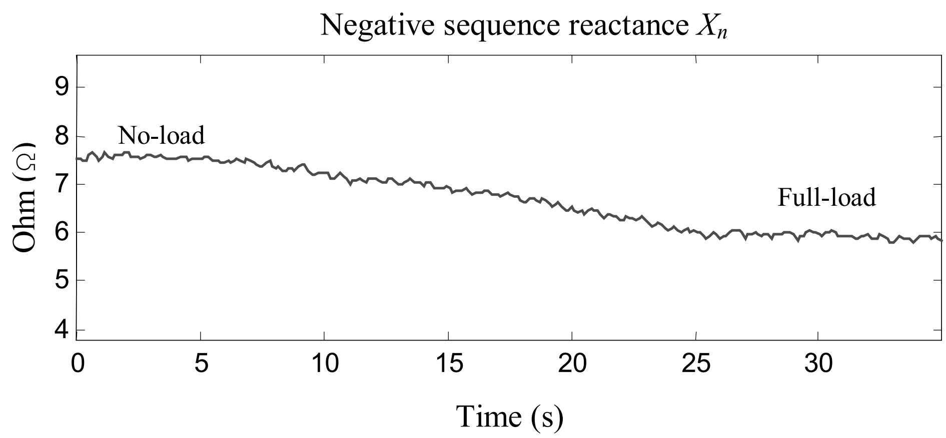


Figure 6.6: Negative sequence reactance X_n (for ABB motor) variation from no-load to full-load under 1% supply unbalance.

To eliminate supply unbalance effect, negative sequence reactance X_{hn} is modelled (Equation 3.33) by

$$\frac{1}{X_{hn}} = \gamma_0 + \gamma_1 V_n + \gamma_2 \sin 2\phi_n + \gamma_3 \cos 2\phi_n + \gamma_4 I_{px} + \gamma_4 I_{py}^2 \qquad 6.1$$

To obtain γ parameters by supply unbalance characterization, negative sequence voltage is applied to motor. Voltage of each phase is varied between 2.5% below to 1.5% above nominal to cause supply unbalance (Figure 6.7). The unbalance percentage is changed linearly over time whilst sampling and fitting is carried out on-line. By using the recursive least square method, γ parameters have been found for Reliance motor as

$$\frac{1}{X_{hn}} = 0.01561 - 0.0009 \cdot V_n - 0.00027 \cdot \sin 2\phi_n - 0.00032 \cdot \cos 2\phi_n + 0.0302 \cdot I_{px} + 0.0093 \cdot I_{py}^2 \qquad 6.2$$

Figure 6.8 shows estimated X_{hn} and measured X_n together for Reliance motor.

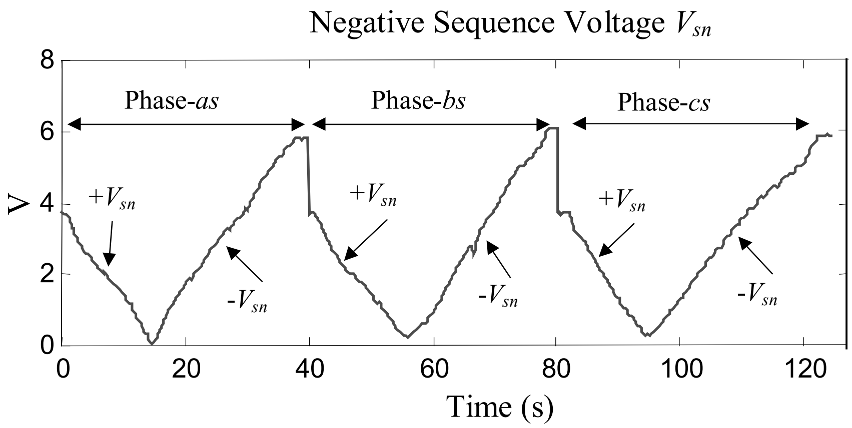


Figure 6.7: Supply negative sequence voltage $\mathbf{V}_{sn}$ variation between 2.5% below ($-\mathbf{V}_{sn}$) to 1.5% above ($+\mathbf{V}_{sn}$) nominal voltage for phase *as, bs,* and *cs* respectively.

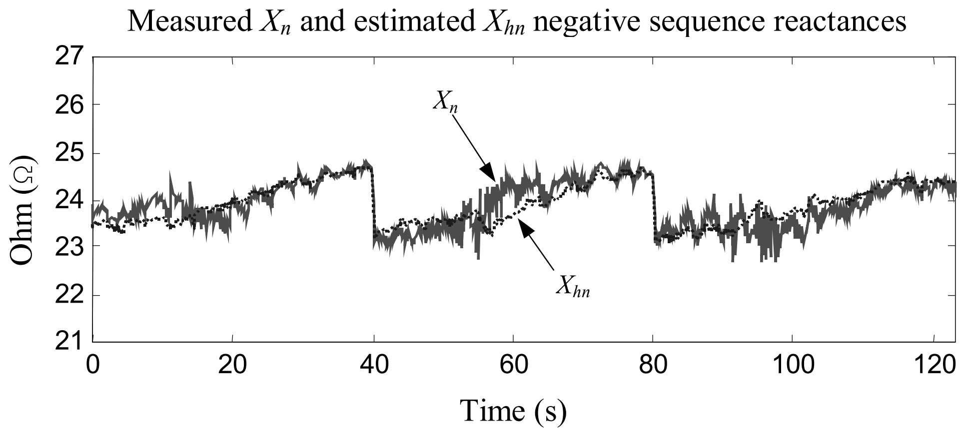


Figure 6.8: Measured X_n and estimated X_{hn} negative sequence reactances for supply unbalance variation between 2.5% below ($-\mathbf{V}_{sn}$) to 1.5% above ($+\mathbf{V}_{sn}$) nominal voltage for phase *as, bs,* and *cs* respectively (Reliance motor).

Figures 6.9 shows the fault current I_{mn} for the temperature independent, variable X_{hn} method, and fault current I_{mn_zn} for the constant impedance method respectively under the supply unbalance variation shown in Figure 6.7. Note that there is no actual short circuit fault. The only quantity being varied is supply negative sequence voltage. Figure 6.9c shows that the contribution to the measured negative sequence current from variation in supply unbalance is almost completely removed by the characterization process.

I_{mn} remains almost zero while supply unbalance is varied. However I_{mn_zn} which is obtained using constant negative sequence impedance $\mathbf{Z}_n$ value is not zero. As it can been seen from the figures error is nearly 15 mA at no fault condition, which is well over the threshold of 10 mA corresponding to two shorted turns (Section 6.3.2).

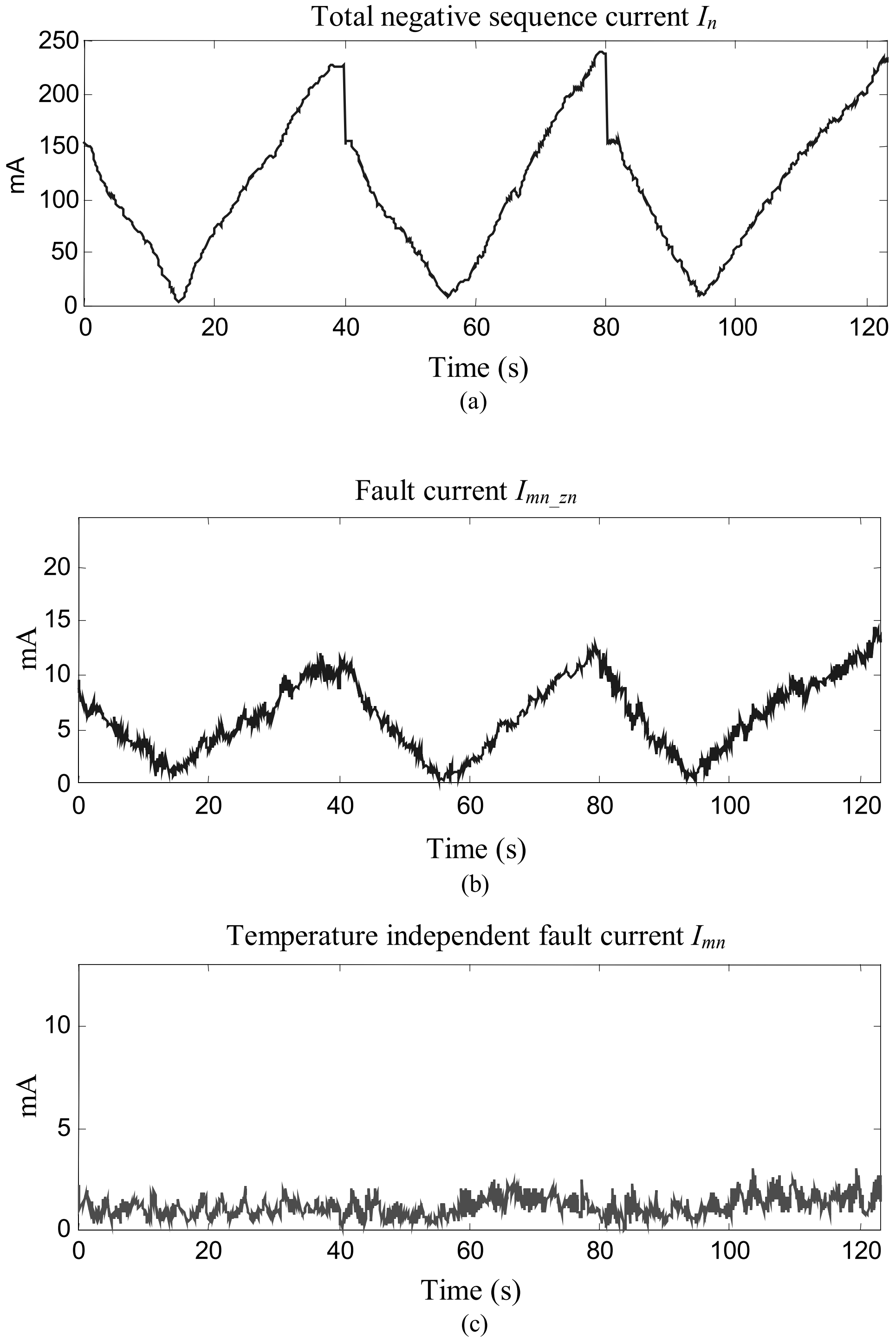


Figure 6.9: a) Measured negative sequence current I_n, **b)** Fault current I_{mn_zn} after subracting $\mathbf{I}_{sn} = \mathbf{V}_{sn} / \mathbf{Z}_n$ from $\mathbf{I}_n$, and **c)** Fault current I_{mn} after subracting $I_{sn} = V_{sn} \sin\theta_n / X_{hn}$ from I_n. Supply negative sequence voltage variation between -2.5% to +1.5% of normal voltage for phases *as, bs* and *cs* successively.

6.3.1.2 Characterization for Mains Voltage and Load Variation

Even with a perfectly balanced supply, mains voltage variation and load variation can give a negative sequence current in excess of that caused by a single turn short circuit. This variation in negative sequence current I_n may appear as an apparent fault current. It arises because of inherent asymmetry in the motor, saturation etc. The effect of load and voltage variation can be eliminated in the same way as supply unbalance effect, by defining a healthy motor negative sequence current I_{mnlv} (Equation 3.30) due to load and supply (positive sequence) voltage variation to be

$$I_{mnlv} = \alpha_0 + \alpha_1 I_{px} + \alpha_2 I_{px}^2 + \alpha_3 I_{py} + \alpha_4 I_{py}^2 \qquad 6.3$$

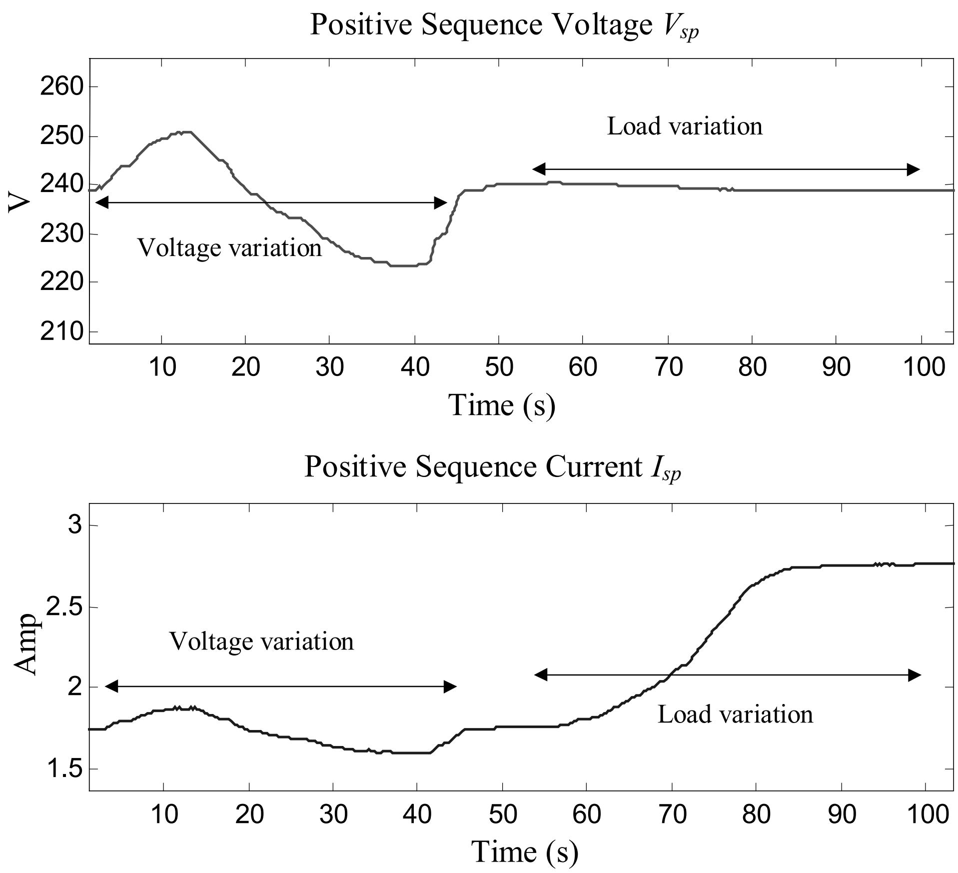


Figure 6.10: Positive sequence voltage V_{sp} and current I_{sp} plotted as V_{sp} is varied between $\pm 5\%$ nominal voltage (during 0-50 sec) and load is changed from zero to full load (during 50-100 sec). Supply is balanced.

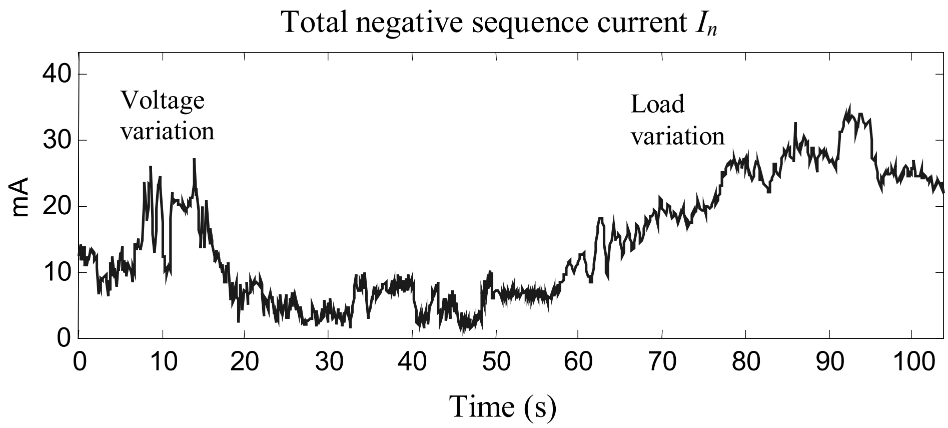
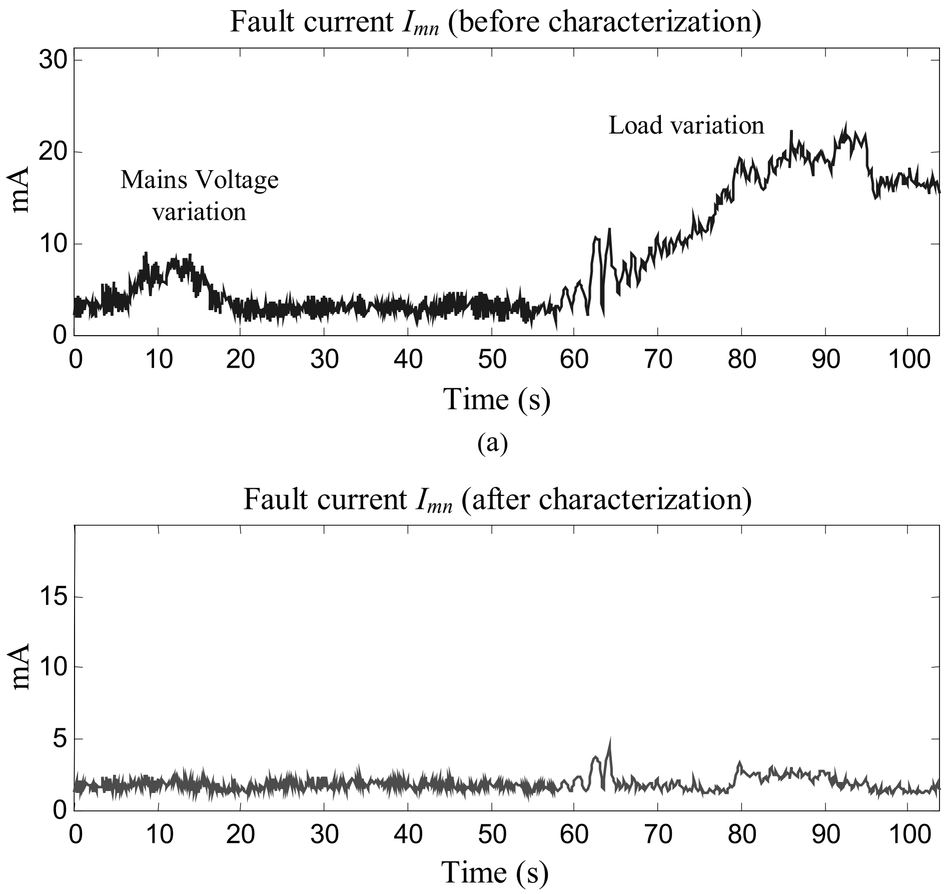


Figure 6.11: Measured total negative sequence current I_n as result of mains voltage variation between $\pm 5\%$ and load variation from no-load to full-load (as in Figure 6.10). Supply is balanced.

Characterization for voltage is carried out by a three phase variable transformer, which is used in order to vary supply mains voltage over the $\pm 5\%$ voltage range (Figure 6.10). The load test was conducted by linearly increasing the motor load from zero to full load at rated voltage. In both conditions supply was kept balanced and there was no fault. Parameters for Equation 6.3 are fitted recursively as voltage and load are changed. Parameters for Reliance motor have been found as

$$I_{mnlv} = 0.1691 - 0.0057 \cdot I_{px} + 0.0027 \cdot I_{px}^2 + 0.238 \cdot I_{py} + 0.089 \cdot I_{py}^2 \qquad 6.4$$

Figure 6.12 shows the I_{mn} before and after characterization for voltage and load variation.



(a)

(b)

Figure 6.12: Fault current I_{mn} for healthy motor under mains voltage and load varitions **a)** before characterization, and **b)** after characterization. Supply is balanced and there is no fault.

Residual effects of supply unbalance and voltage-load variation may appear as an apparent fault current (Figures 6.9b and 6.12a), which have to be separated from total measured negative sequence current before being able to measure fault current caused by actual the short circuits. The results given above show that the given empirical functions can be used to remove the non-linear effects. The resulting fault currents are shown in Figure 6.9c and 6.12b for the healthy motor. This residual current value acts as a threshold for detection of true fault current. Without characterization, the high apparent fault currents in Figures 6.9b and 6.12a prevent the effects of small number of shorted turns being distinguished from healthy motor imperfections.

6.3.1.3 Thermal Effect

As it has been shown in Chapter 2, negative sequence impedance of induction motor is equal to

$$\mathbf{Z}_n = r_s + \frac{r_r}{2 - s_p} + j(X_s + X_r) = R_n + jX_n \qquad 6.5$$

Normal operating conditions will heat up the motor and this will directly alter the resistive part R_n, of the motor negative sequence impedance $\mathbf{Z}_n$. In Chapter 3 section 3.5 a method of minimisation of thermal effects has been proposed.

Table 6.4 show results obtained from simulation by adding equal external resistance R_{ther} to the three stator phase winding to give the effect of temperature rise on the resistance part of stator winding under constant 1 % supply unbalance for Reliance motor.

As can be seen from results, there is linear relation between temperature effect and fault current I_{mn_zn}, which use constant negative sequence impedance. On other hand the temperature independent fault current I_{mn} remains zero.

R_{ther} (Ω)	I_{mn} (mA)	I_{mn_zn} (mA)	I_n (mA)
0.5	1	12	233
1	1	22	227
1.5	1	33	220
2	1	43	214

Table 6.4: Simulation result for temperature effect on total negative sequence and faults currents (Reliance motor).

To show clear picture of temperature effect in experiment, three external variable resistances R_{ther} have been used between supply terminals and motor terminals to add equal external resistance in each of the motor phases. They had been connected between measurement unit and motor. Figure 6.13 shows the experiment setup. The resistors can be used to test experimentally the effect of increase in stator resistance due to motor heating.

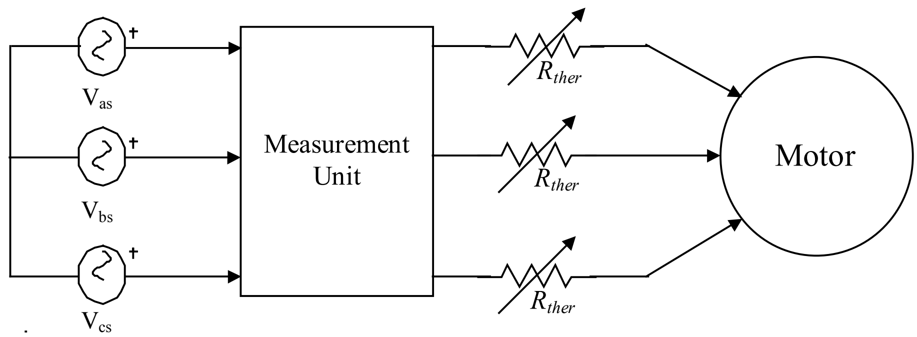


Figure 6.13: Experimental setup for temperature effect measurement

To test the effect of temperature on negative sequence current measurement, tests were taken at no-load under 1% supply unbalance. Figure 6.14 shows measured negative sequence reactance X_n and resistance R_n, before and after resistances were introduced to give the effect of a 25% increase in stator resistance, corresponding to approximately 60 ^{0}C rise in winding temperature. As expected, there is no change in the measured value of X_n, whereas the resistive component R_n is affected directly.

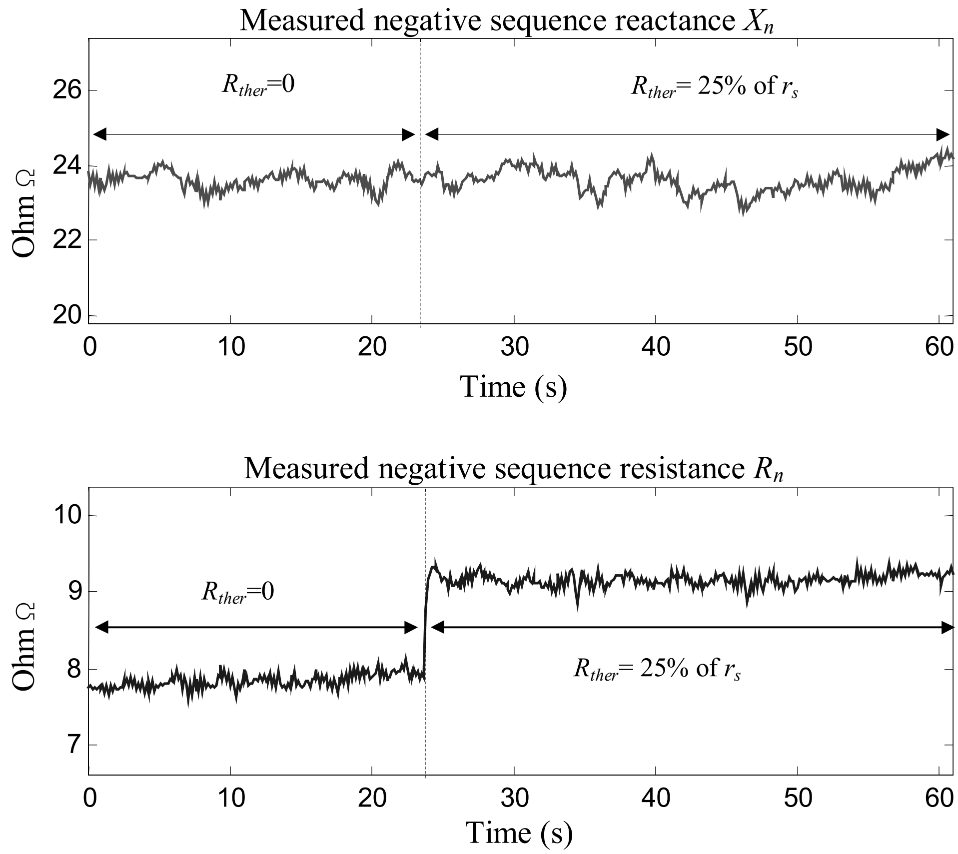


Figure 6.14: Measured negative sequence reactance X_n and resistance R_n before and after r_s has been increased 25% by R_{ther} in each phase.

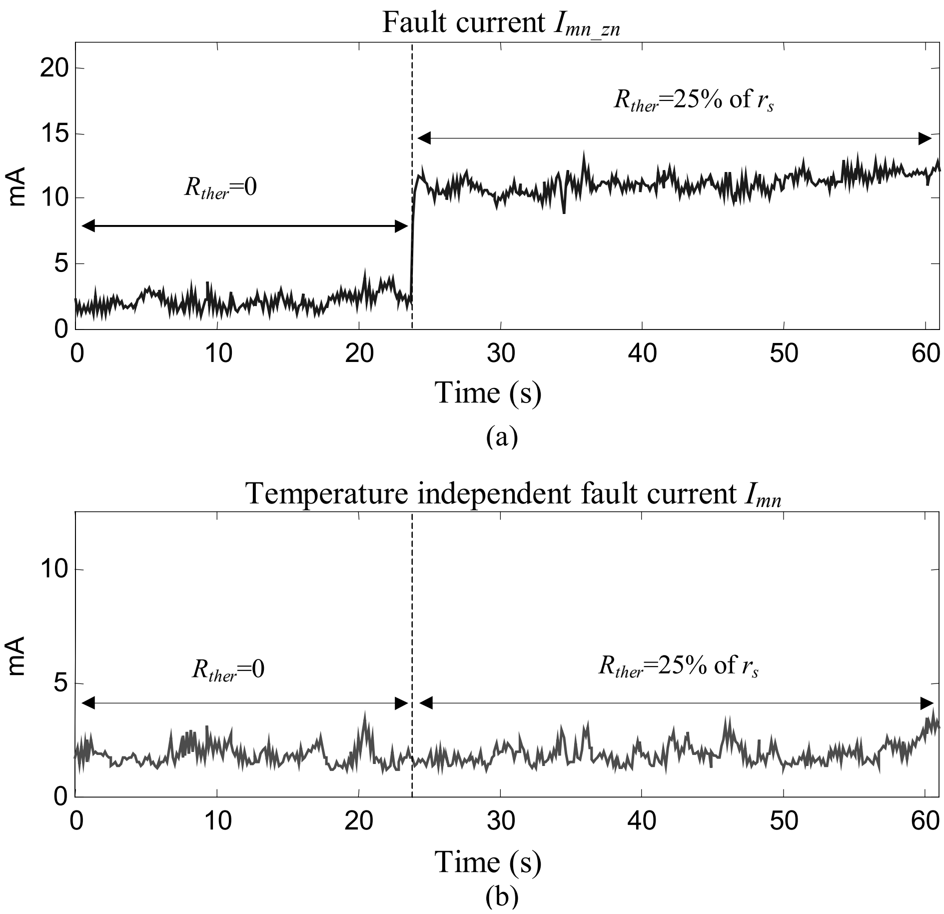


Figure 6.15: a) Fault current I_{mn_zn}, and **b)** temperature independent fault current I_{mn} before and after 25% increase in the stator resistance of all phases. I_{mn_zn} changes because R_n changes and would give fault indication although there is no fault. I_{mn} stays nearly zero as it should be.

Figure 6.15 shows test results for fault current I_{mn_zn} (constant $\mathbf{Z}_n$ method), and fault current I_{mn} (temperature independent, variable X_n). As it can been seen from the figures while I_{mn_zn} is directly effected by the increase in stator phase resistance r_s, the fault current I_{mn} calculated using only X_n (temperature independent method) remains essentially zero, as it should in the absence of a short circuit fault.

Simulation and experimental results show that relation between negative sequence fault current I_{mn_zn}, supply negative sequence voltage V_{sn}, and R_{ther} can be summaries as follows

$$I_{mn\_zn} \propto (R_{ther}, V_{sn}) \quad \text{and} \quad R_{ther} \propto T \qquad 6.6$$

where T is temperature.

6.3.2 Results for Inter-Turn Faults

Results here are presented for different numbers of short circuited stator turns under different operating conditions after characterization. An external resistance *r_ext* may be used to control the short circuit current to prevent winding damage. I_{r_ext} is the short circuit current flowing in the external variable resistance *r_ext* (Figure 6.16). The displayed fault current I_{mn} is temperature independent, variable X_n motor fault current which is obtained by subtracting contribution of supply negative sequence current I_{sn}, and mains voltage and load variation negative sequence current I_{mnlv} from the total measured negative sequence current I_n. With no turns shorted, I_{mn} has to stay zero or under threshold. In Figures 6.17-6.20 the external resistance has been adjusted to allowed current in a single shorted turn to be full load current (2.8 A). As explained above, a maximum of 4 turns can be shorted for the Reliance motor, which is 1.5% of total number of turns in one phase.

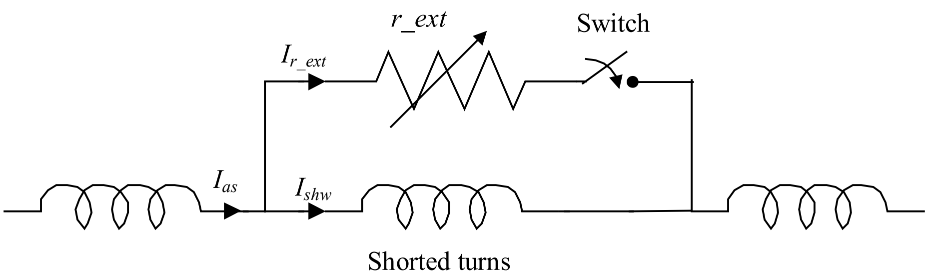


Figure 6. 16: Short circuiting with external resistance.

Figure 6.17 to 6.20 show the fault current I_{mn} for the tested motor at zero load and balance supply. Fault has been switched on and off at nearly constant intervals to see the response of I_{mn} to the fault.

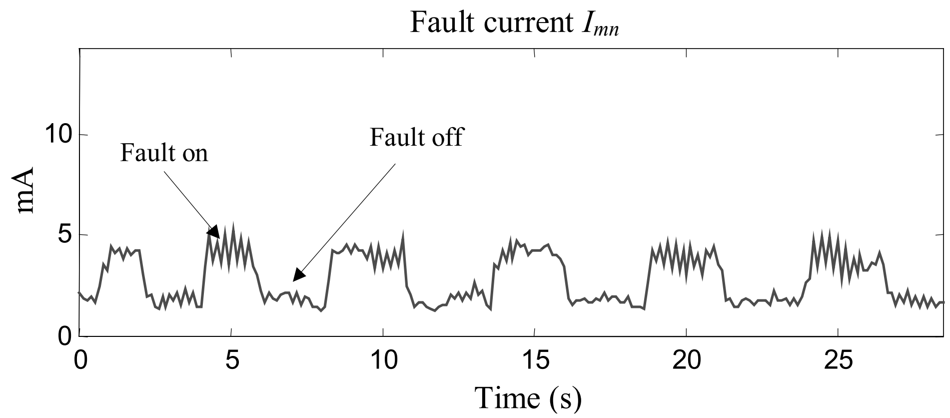


Figure 6.17: Fault current I_{mn} for 1 turn shorted at no-load with a balanced supply. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 2.8A.

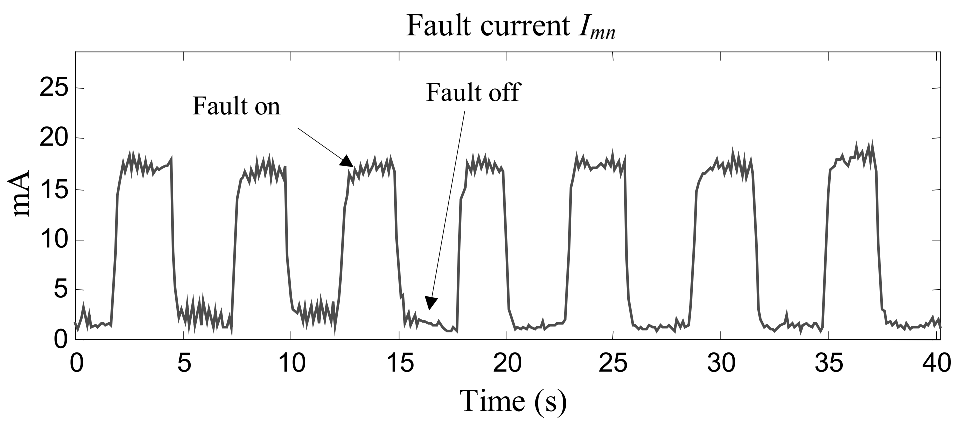


Figure 6.18: Fault current I_{mn} for 2 turns shorted at no-load with a balanced supply. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 5.5A.

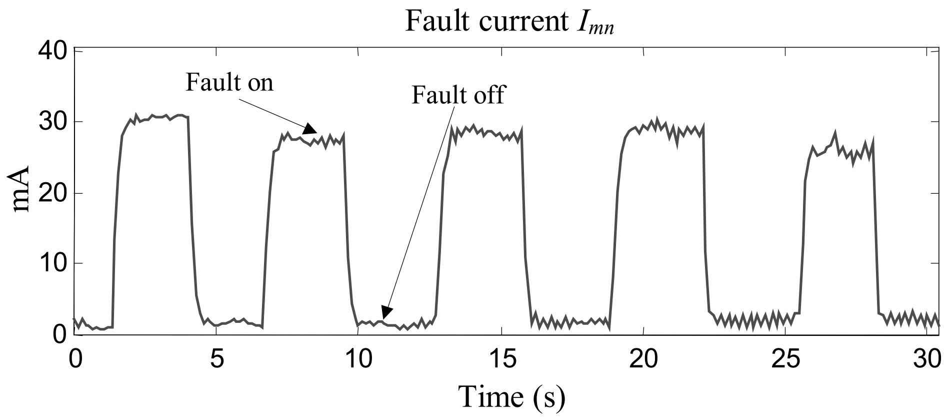


Figure 6.19: Fault current I_{mn} for 3 turns shorted at no-load with balanced supply. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 9.2A.

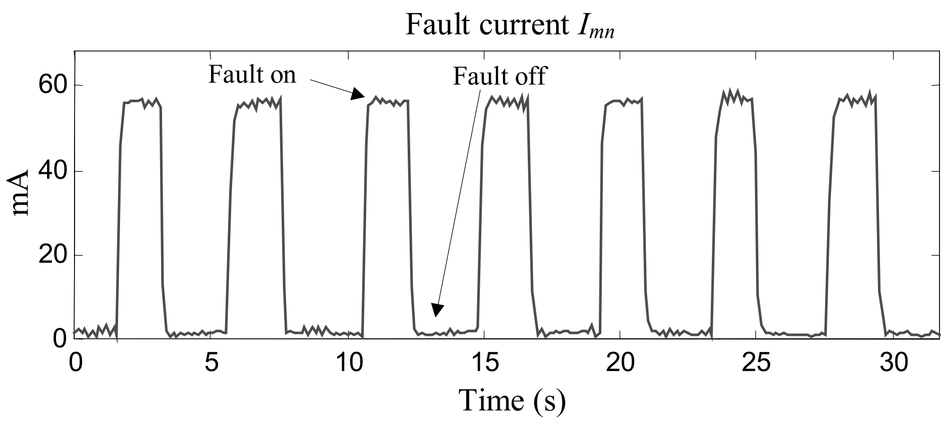


Figure 6.20: Fault current I_{mn} for 4 turns shorted at no-load with a balanced supply. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 10.3A.

Figure 6.21 and 6.22 show the fault current I_{mn} under the full-load with a balanced supply.

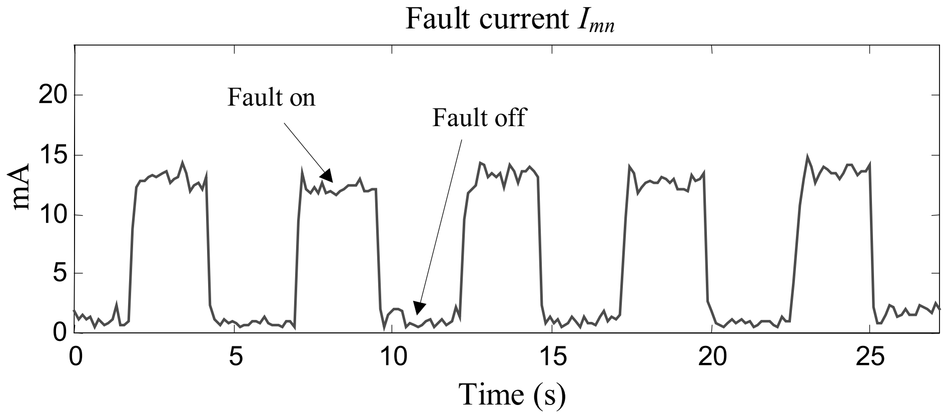


Figure 6.21: Fault current I_{mn} for 2 turns shorted at full-load with balanced supply. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 5.5A.

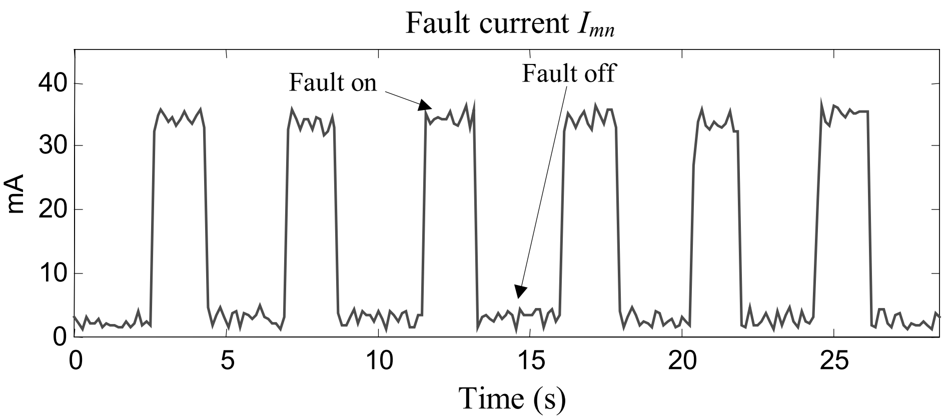


Figure 6.22: Fault current I_{mn} for 3 turns shorted at full-load with a balanced supply. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 9 A.

Figure 6.23 to 6.26 show the fault current I_{mn} under 1% supply unbalance and zero-load condition. The short circuit is switched on and off.

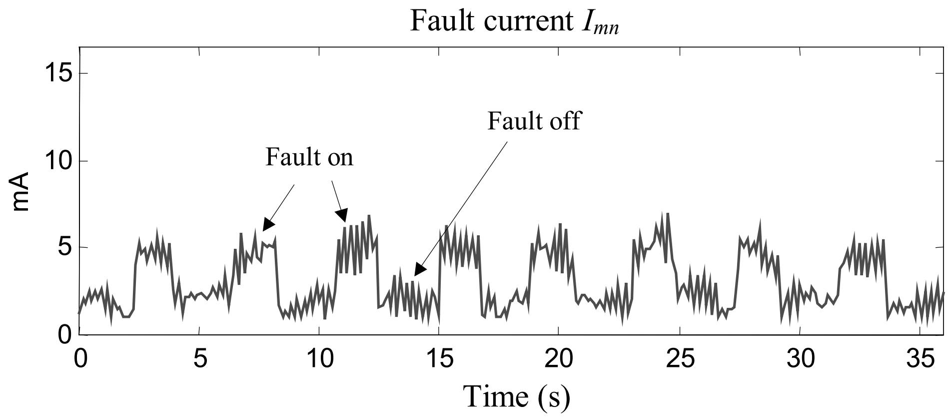


Figure 6.23: Fault current I_{mn} for 1 turn shorted at 1% supply unbalance and no-load. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 2.8. A.

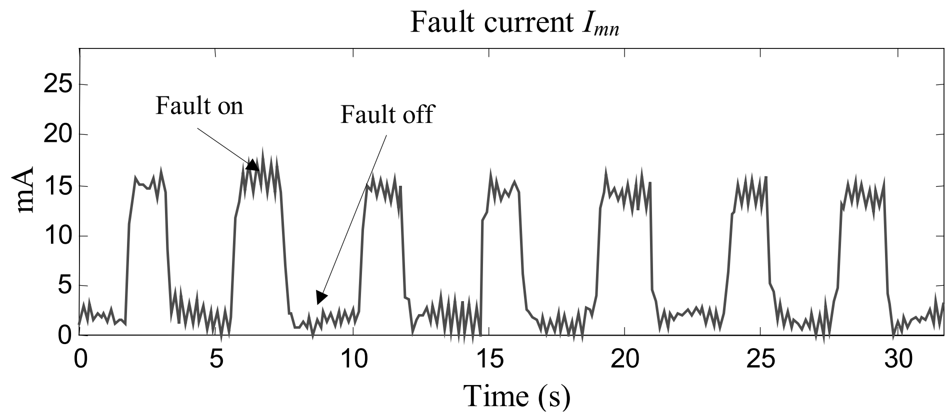


Figure 6.24: Fault current I_{mn} for 2 turns shorted at 1% supply unbalance and no-load. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 5 A.

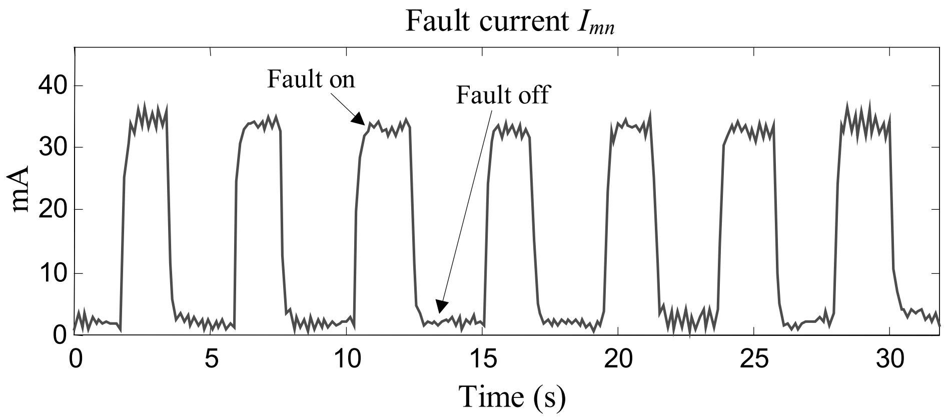


Figure 6.25: Fault current I_{mn} for 3 turns shorted at 1% supply unbalance and no-load. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 8.6 A.

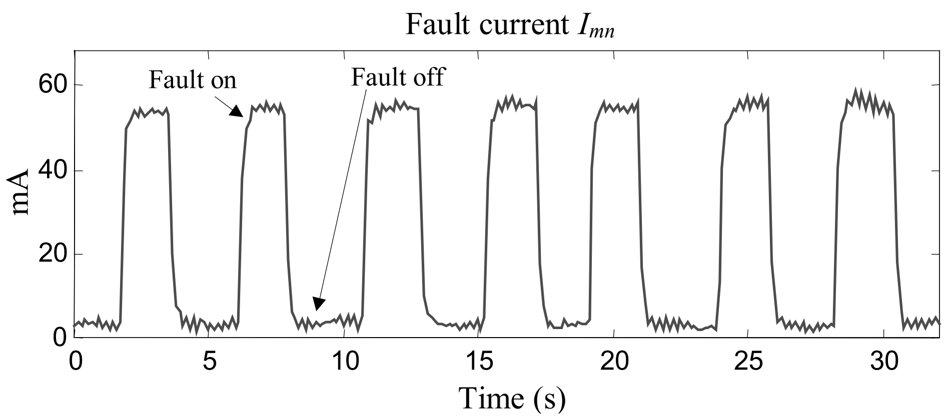


Figure 6.26: Fault current I_{mn} for 4 turns shorted at 1% supply unbalance and no-load. Fault is switched on and off to show the response. Short circuit current I_{r_ext} is 9.8 A.

Next two figures show the fault current for 1, 2, 3, and 4 turns shorted respectively at the same figure under different operating conditions. As it can be seen, fault current will clearly increase with the shorted number of turns.

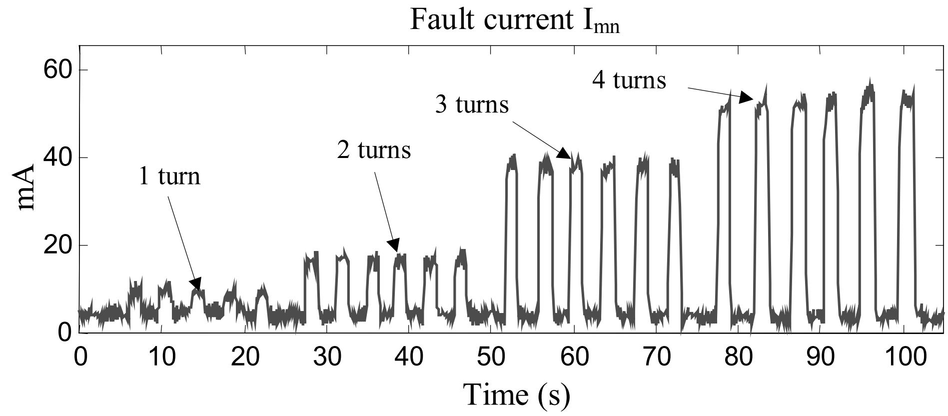


Figure 6.27: Fault current I_{mn} for 1, 2, 3, and 4 turns shorted respectively at 1% supply unbalance and no-load. Fault is switched on and off for each one to show the response.

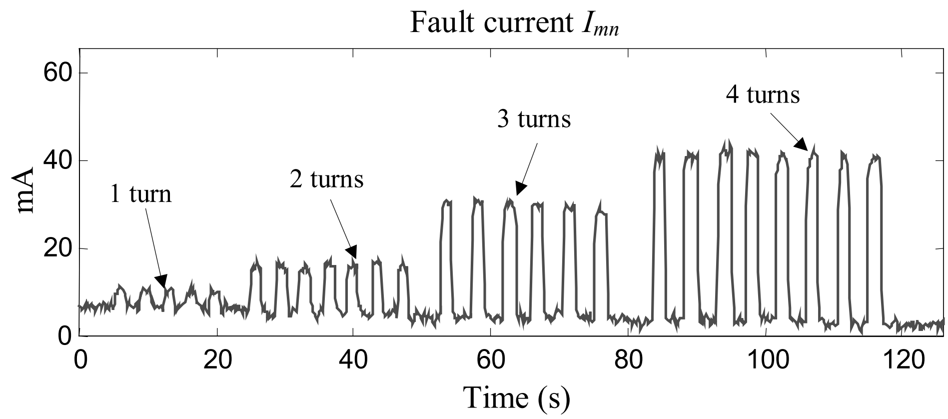


Figure 6.28: Fault current I_{mn} for 1, 2, 3, and 4 turns shorted respectively at 1% supply unbalance and full-load. Fault is switched on and off for each one to show the response.

It can be seen that single shorted turns give a clear visible signal, and in figure 6.21 to 6.28 that the fault signal amplitude is independent of load and supply unbalance.

6.3.3 Interpretation of Results - Sensitivity

Test results show that under all operating conditions of load, supply unbalance, and voltage variation, short circuit faults can be clearly seen from motor negative sequence fault current.

If the short circuit current I_{shw} flows in n turns, then the contribution to motor phase current is expected by transformer theory to be given by

$$I'_{shw} = \frac{n}{N} I_{shw} \tag{6.7}$$

where n is number of shorted turns, and N is the total number of turns in the phase winding. By assuming the introduced current is divides equally between the other two phases, the negative sequence current can be calculated by using symmetrical components and gives

$$I_{mn} = \frac{1}{6} I'_{shw} \tag{6.8}$$

The experimental results and simulation results show that current is divided in more complex way between phases and the estimated current I_{mn} has to be multiplied by factor of $\frac{3}{2}$. So that result will be

$$I_{mn\_\min} = \frac{3}{2}\frac{1}{6} I'_{shw} = 0.25 I'_{shw} = 0.25 \frac{n}{N} I_{shw} \tag{6.9}$$

For the Reliance motor $N = 252$. The measurable fault current decreases as n decreases, so that a one turn short circuit will be the most difficult to detect. A one turn short with a fault current equal to the motor full load current (2.7 amps) can probably be tolerated without rapid motor failure, and for detection will require that we can detect a minimum negative sequence current of

$$I_{mn\_est} = 0.25 \cdot 2.7 \frac{1}{252} \text{amps} = 3\,\text{mA} \tag{6.10}$$

The experiments show that when limiting short circuit current to about 8.98 A in 3 shorted turns, the observable fault current is of the order 30 mA. The formula above for I_{mn} would predict $0.25 \cdot 8.98 \cdot 3 / 252$ amps i.e. 27 mA, so that observations seem to agree with the formula for I_{mn}. Hence, a target negative sequence current sensitivity is about 5 mA in a full load current of 2.7 amps i.e. 1 part in 500. Figures 6.8 and 6.11, in particular, show that this sensitivity is achievable. They also show that there is no chance of achieving this sensitivity without motor characterization.

The result in Table 4 compare experimental result with simulation results obtained for the Reliance motor with up to 4 turns shorted. Short circuit current in simulation has been limited by using 0.3 ohm r_ext to get similar current in experiment.

	Experimental result			Simulation result		
n	I_{mn} (mA)	I_{shw} (A)	I_{mn_est} (mA)	I_{mn} (mA)	I_{r_ext} (A)	I_{mn_est} (mA)
1	4	2.7	3	4	3.3	3
2	15	5.3	11	13	5.8	12
3	30	8.98	27	26	8.1	24
4	54	10	40	43	10.3	41

Table 6.5: Experimental and simulation results for different number of turns with limited short circuit current.

6.3.4 Threshold Setting

Although test results show that a single shorted turn can be seen very clearly after characterization, motor fault current could vary above 5 mA especially at full load with high supply unbalance variation in different phases. Figure 6.29 show this effect. In Figure 6.29 supply unbalance varied from +1% to -1% in phase *as, bs,* and *cs* successively. Because of non-linearities of motor even after characterization supply unbalance variation in different phases show different effect on negative sequence current. While fault current I_{mn} remains zero in one phase, it can rise as much as 10 mA in other phases with same supply unbalance variation at full load. So that threshold has to be set to a bigger value (instead of single turn value) to avoid false alarm, which for

Reliance motor 10 mA will give better result. Note that simulation results suggest that fault current will be much bigger than 5 mA for a single turn shorted with zero external resistance.

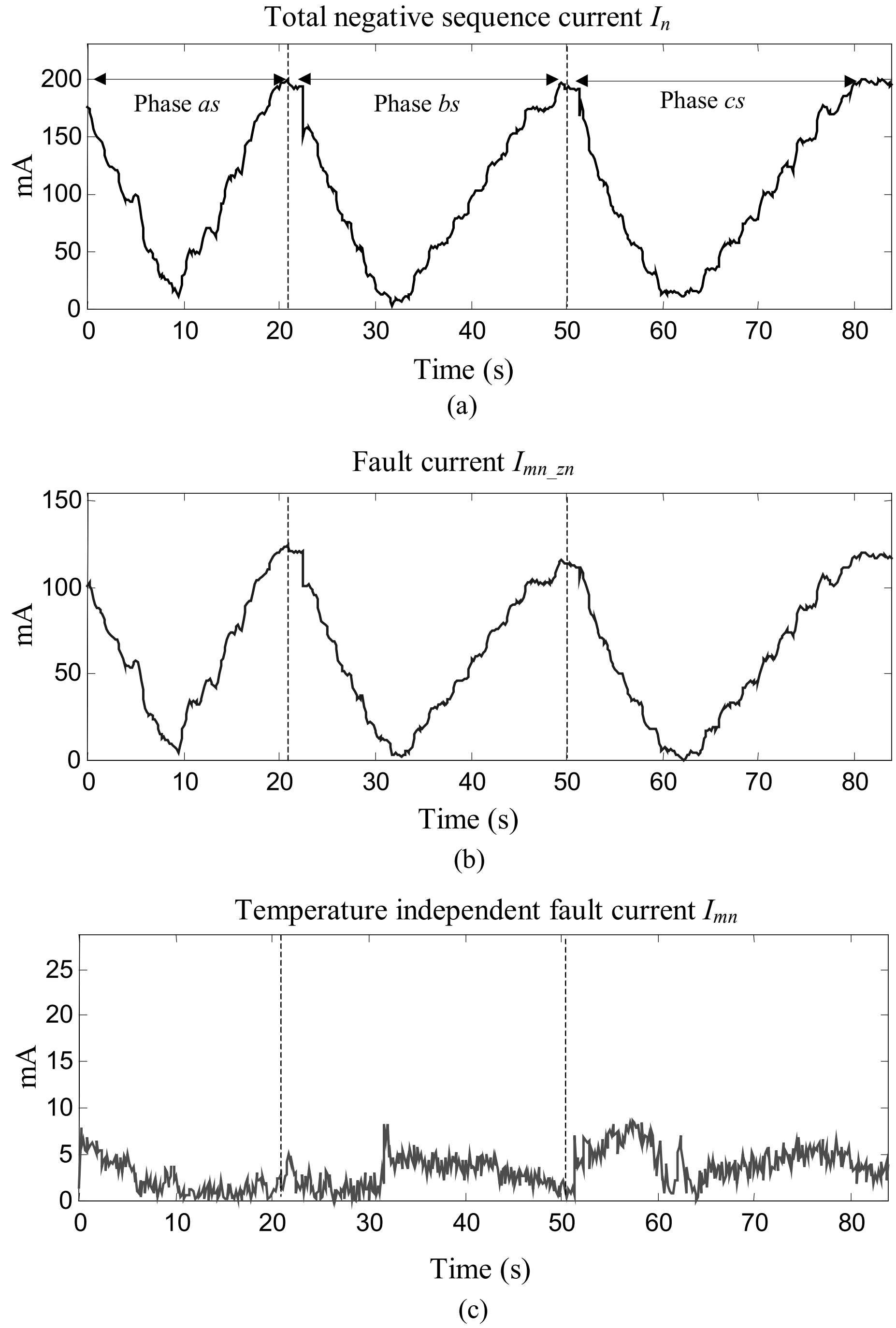


Figure 6.29: a) Measured negative sequence current I_n, **b)** Fault current I_{mn_zn} after subracting $\mathbf{I}_{sn} = \mathbf{V}_{sn} / \mathbf{Z}_n$ from $\mathbf{I}_n$, and **c)** Fault current I_{mn} after subracting $I_{sn} = V_{sn} \sin\theta_n / X_{hn}$ from I_n. Supply negative sequence voltage variation between -1% to +1% of normal voltage under full load for phases *as, bs* and *cs* successively.

6.4 Conclusion

The stator fault diagnosis algorithm presented in Chapter 3 has been applied to motors with added shorting taps. Experimental setups have been explained and the tested motors have been introduced. Tests were performed in the steady state for inter turn short circuits under different operating conditions. Characterization has been carried out for the tested motors to remove contribution of negative sequence current arising from motor non-linearities, and results of fitted parameters for empirical formulae were presented. By using these parameters the effects of supply unbalance, mains voltage, and load variation have been minimized.

Minimization of temperature effect on negative sequence current has been presented both for simulation and experimental results with relation between negative sequence current, negative sequence voltage, and temperature.

A sensitivity requirement has been defined from simple short circuit current turn relation. Threshold setting was explained by using test results and sensitivity results.

The results show that it is not possible to achieve high sensitivity without motor characterization.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The algorithm for detecting stator inter-turn short circuit faults by monitoring negative sequence current has been developed which is one of simplest but most powerful techniques for stator fault detection.

An arbitrary three phase signal can be divided into positive, negative, and zero sequence components using time phasors described in the conventional theory of symmetrical component. The Power Decomposition Technique (PDT) method was used to obtain these sequence components. In this technique, sequence components were obtained directly in the time domain by separating positive and negative sequence quantities expressed in a two-dimensional *dq* co-ordinate system.

Supply unbalance is one of the major problems of using negative sequence current as a stator inter-turn fault indicator. The analysis of induction motor under different operating conditions in Chapter 2 showed that as long as iron saturation is ignored and speed may be assumed constant due to a sufficiently large inertia, when both positive and negative sequence voltages are applied simultaneously, the current of the two sequence do not react upon each other. This property can be used to define sequence impedances which are simply the ratio of sequence voltage to sequence current. The negative sequence impedance was used to separate negative sequence current into a negative sequence current due to supply negative sequence voltage and a negative current due to stator impedance imbalances.

In the real world, due to manufacturing limitations all motors posses some degree of unbalance. At different operating condition because of motor iron saturation, winding unbalance, rotor static eccentricity and other inherent imbalance in the motor, motor positive and negative sequences can be interdependent. These effects were eliminated by considering theoretical and experimental results together. The motors were characterized by using semi-empirical formulae for load, voltage and supply unbalance variations to get residual negative sequence currents for a healthy motor under different operating conditions.

Motor negative sequence impedance is directly altered by temperature changes and in a medium sized motor the negative sequence impedance can vary with slip. The effects of temperature change and variation in the negative sequence resistance with slip were minimized by using only negative sequence reactance to find supply negative sequence current.

A compensation method has been introduced for non-simultaneous sampling. With non-simultaneous sampling, the negative sequence components will be dependent on positive sequence components and vice versa. This will introduce more complexity in the fault detection algorithm. By using the compensation method introduced, effect of non-simultaneous signal sampling can be eliminated.

The hardware and software designs for the stator fault detection algorithm have been described. Implementation of the algorithm has been carried out using two different hardware configurations, which were explained in Chapter 5.

Orthogonal axis theoretical models for three-phase induction motor having asymmetrical winding on the stator were presented. The first model assumed that each motor phase has a different number of turns. The second model assumes phase *as* has two windings in series, and uses the results of the first model to transfer phase *as* to *qd* axis for applying a short circuit to one part of the *as* winding. The developed models are based on general machine parameters.

The analysis of simulation results has revealed many interesting aspects of stator turn faults. The sensitivity of the algorithm can be checked from simulation results, as well as the effects of temperature and the number of shorted turns on negative sequence current. The analysis has shown that any change in the negative sequence current is much reduced if the short circuit current is limited by external resistance, even in the simulation.

The experimental investigations were performed on specially wound motors in the steady state for inter-turn short circuits under different operating conditions. Characterization results have been presented for supply unbalance, and mains voltage and load variations. Minimization of temperature effects on negative sequence current has been presented both for simulation and experimental results. The results obtained from simulation and experiments gave confidence that the developed stator fault detection algorithm achieves high sensitivity under all operating conditions.

7.2 Future Work

The developed algorithm was implemented under laboratory conditions. Tests were carried out with several motors under different conditions. Though these experiments can be considered to be similar to real processes, testing of motors in industrial settings needed to be performed.

The parameters selected for the semi-empirical formulae that have been used to eliminate supply unbalance, and load and mains voltage variation, can be further developed to give parameter selection a more theoretical basis by more detailed physical modelling of motor imperfections (saturation, load dependence of saturation of slot teeth).

The observation that negative sequence impedance changes under supply unbalance and load variation suggested that further investigation is needed to explained the reason behind this variation (variation is between 10% to 50%), possibly using the full negative sequence impedance matrix by incorporating saturation, winding imperfection etc.

Negative and positive sequence voltage and current information, which contains all the information needed for motor condition monitoring, can be used to form a fault signature for a neural network to handle non-linear motor effects instead of using semi-empirical formulae.

References

[1] Huang, S.P., "Active Mains Supply Harmonic Filtering", *University of Sussex, PhD Thesis 1997.*

[2] Kohler, J.L., Sottile, J., and Trutt, F.C., "Alternatives for Assessing the Electrical Integrity of Induction Motors", *IEEE Trans. On Industry Applications, Vol.28, No.5, pp.1109-1117, Sep/Oct. 1992.*

[3] Kliman, G.B., Premerlani, W.J., Koegl, R.A., and Hoeweler, D., " A New Approach to On-Line Turn Fault Detection in AC Motors", *IAS Annual Meeting, pp.687-693, October 1996.*

[4] Finley, W.R., and Burke, R.R., "Troubleshooting Motor Problems", *IEEE Trans. On Industry Applications, Vol. 30, No.5, pp. 1383-1396, September/October 1994.*

[5] Thomson, W.T., "A Review of On-line Condition Monitoring Techniques for Three-Phase Squirrel-Cage Induction Motors- Past Present and future", *The 1999 IEEE International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives, IEEE SDEMPED'99, pp. 3-18, 1999*

[6] Belmans, R., and Hameyer, K., "Different approaches to the preventive Maintenance of induction motors", *ICEM'96, pp. 423-428, 1996*

[7] Paoletti, G.J., and Rose, A., "Improving Existing Motor Protection for Medium Voltage Motors", *IEEE Trans. On Industry Applications, Vol.25, No.3, pp.456-464, May/June 1989.*

[8] Bonnett, A.H., and Soukup, G.C., "Cause And Analysis Of Stator And Rotor Failures In Three-Phase Squirrel-Cage Induction Motors", *IEEE Trans. On Industry Applications, Vol. 28, No.4, pp. 921-937, July/August 1992.*

[9] Motor Reliability working Group, "Report of Large Motor Reliability Survey of Industrial and Commercial Installations", *Part I, and II*, *IEEE Trans. On Industry Applications, Vol. IA, No.4, 1985.*

[10] "Improved Motors for Utility Applications", *EPRI EL-2678, Vol 1, 1763-1, final report, 1982,*

[11] Thorsen, O.V., and Dalva, M., "Failure Identification and Analysis for High-Voltage Induction Motors in the Petrochemical Industry", *IEEE Trans. On Industry Applications, Vol. 35, No.4, pp. 810-818, July/August 1999.*

[12] Wang, Y., "The ZCT Method of Induction Motor Failure Prediction and Speed Monitoring", *University of Sussex, PhD Thesis, 1997*.

[13] Tanver, P.J., and Penman, J., "Condition Monitoring of Electrical Machines", *Research Studies Press Ltd., 1987*

[14] Moreau, S., Trigeassou, J.C., Champenois, G., and Gaubert, J.P., "Diagnosis of Induction Machines: A Procedure for Electrical Fault Detection and Localization", *The 1999 IEEE International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives, IEEE SDEMPED'99, pp. 225-229, 1999*.

[15] Chow, M.Y., "Methodologies of Using Neural Network and Fuzzy Logic Technologies for Motor Incipient Fault Detection", *World Scientific Publication Co. Pte. Ltd., 1997, ISBN 981-02-3265-9*.

[16] Farag, S.F., Bartheld, R.G., Habetler, T.G., "An Integrated, On-Line, Motor Protection System*", IEEE IAS Annual Meeting, pp. 117-122, 1994*

[17] Calis, H., "Current Based Detection of Mechanical Faults In Induction Motors", *University of Sussex, PhD Thesis 1998*.

[18] Vetcha, S.B., "Fault Diagnosis in Pumps by Unsupervised Neural Networks", *University of Sussex, PhD Thesis 1998*.

[19] Kliman, G.B., and Stein, J., "Induction Motor Fault Detection Via Passive Current Monitoring- A Brief Survey", *Proc. Of the 44th Meeting of the Mechanical Failures Prevention Group, pp. 49-65, April 1990*,

[20] Hodowanec, M.M., and Bezesky, D.M., "Field Motor Testing: Limiting Risk", *IEEE Industry Applications Magazine, pp. 38-48, May/June 1999*.

[21] Thorsen, O.V. and Dalva, M., "A Survey Of Faults On Induction Motors In Offshore Oil Industry, Petrochemical Industry, Gas Terminals, And Oil Refineries", *IEEE Trans. On Industry Applications, Vol. 31, No. 5, pp. 1186-1196, September/October 1995*.

[22] Maier, R., "Protection Of Squirrel-Cage Induction Motor Utilizing Instantaneous Power And Phase Information", *IEEE Trans. On Industry Applications, Vol. 28, No.2, pp.376-380, March/April 1992.*

[23] Penman, J., Sedding, H.G., Lloyd, B.A., and Fink, W.T., "Detection and Location of Interturn Short Circuits In The Stator Windings of Operating Motors", *IEEE Trans. On Energy Conversion, Vol. 9, No. 4, pp. 652-658, 1994.*

[24] Cardoso, A.J.M., Saraiva, E.S., Mateus, M.L.S., and Ramalho, A.L., "On-Line Detection Of Airgap Eccentricity In 3-Phase Induction Motors, By Park's Vector Approach", *International Conf. On Electrical Machines and Drives, No. 341, pp. 61-66, 1996.*

[25] Natarajan, R., "Failure Identification of Induction Motors by Sensing Unbalanced Stator Currents", *IEEE Trans. On Energy Conversion, Vol. 4, No. 4, pp. 585-590, December 1989*

[26] Cash, M.A., Habetler, T.G., and Kliman, G.B., "Insulation Failure Prediction In Induction Machines Using Line-Neutral Voltages", *IEEE IAS Annual Meeting, Vol. 1, pp. 208-212, October 1997*.

[27] Thomson, W.T., and McRae, C.J., "On-Line Current Monitoring To Detect Inter-Turn Stator Winding Faults In Induction Motors", Proc. 24th Universities Power Engineering Conf., pp. 477-481, 1989.

[28] Cardoso, A.J.M., Cruz, S.M.A., and Fonseca, D.S.B., "Inter-Turn Stator Winding Fault Diagnosis In Three-Phase Induction Motors, By Park's Vector Approach", *1997 IEEE International Electric Machines and Drives Conference, pp. MB1-5.1-MB1-5.3, May 1997*.

[29] Hurst, K.D., and Habetler, T.G., "A Thermal Monitoring And Parameter Tuning-Scheme For Induction Machines" *IEEE IAS Annual Meeting, Vol. 1, pp. 136-142, October 1997*.

[30] Schoen, R.R., Lin, B.K., Habetler, T.G., Schlag, J.H., and Farag, S., "An Unsupervised, On-Line System For Induction Motor Fault Detection Using Stator Current Monitoring", *IEEE Trans. On Industry Applications, Vol. 31, No.6, pp. 1280-1286, November/December 1995.*

[31] Fortescue, C.L., "Method of Symmetrical Co-ordinates Applied to the Solution of Polyphase Network", *AIEE Trans. Vol.37, pp.1027-1140, 1918.*

[32] Leonard, R.A., and Thomson, W.T., "Vibration and Stray Flux Monitoring for Unbalanced Supply and Inter-Turn Winding Fault Diagnosis in Induction Motors", *1st U.K. International Conference on Condition Monitoring, pp. 340-354, 1984.*

[33] Ho., S.L., and Lau, K.M., "Detection of Faults in Induction Motors Using Artificial Neural Networks", *Electrical Machines and Drives, Conference Publication No. 412, pp. 176-181, September 1995*.

[34] Cameron, J.R., Thomson, W.T., and Dow, A.B., "Vibration and Current Monitoring for Detecting Airgap Eccentricity In Large Induction Motors", *IEE Proceedings, Vol. 133, Pt. B, No. 3, pp. 155-163, May 1986*

[35] Tetrault, S.M., Stone, G.C., and Sedding, H.G. "Monitoring Partial Discharges on 4-kV Motor Windings", *IEEE Trans. On Industry Applications, Vol. 35, No.3, pp. 682-688, May/June 1999*.

[36] Stone, G.C., and Kapler, J., "Using Partial Discharge Measurement Technology To Implement Predictive Maintenance In High Voltage Motor And Generator Stator Windings" *IEEE/PCA Cement Industry Technical Conference, pp. 63-68, April 1997.*

[37] Stone, G.C., and Sedding, H.G., "Application Of Partial Discharge Testing To Motor And Generator Stator Winding Maintenance" *IEEE Trans. On Industry Applications, Vol. 32, No. 2, pp. 459-464, March/April 1996*.

[38] Kemp, I.J., "Partial Discharge Plant-Monitoring Technology: Present And Future Developments", *IEE Proc.-Sci. Meas. Technol., Vol. 142, N0. 1, pp. 4-10, January 1995.*

[39] Stone, G.C. and Sedding, H.G., "In-Service Evaluation Of Motor And Generator Stator Windings Using Partial Discharge Tests", *IEEE Trans. On Industry Applications, Vol.31, No.2, pp.299-303, March/April 1995*.

[40] Tanaka, T., "Partial Discharge Pulse Distribution Pattern Analysis", *IEE Proc.-Sci. Meas. Technol., Vol. 142, N0. 1, pp. 46-50, January 1995.*

[41] Borsi, H., Gockenbach, E., and Wenzel, D., "Separation Of Partial Discharges From Pulse-Shaped Noise Signals With The Help Of Neural Networks", *IEE Proc.-Sci. Meas. Technol., Vol. 142, N0. 1, pp. 69-74, January 1995.*

[42] Hucker, T., and Kranz, H.G., "New Approach In Partial Discharge Diagnosis And Pattren Recognition" *IEE Proc.-Sci. Meas. Technol., Vol. 142, N0. 1, pp. 89-94, January 1995.*

[43] Filippetti, C., Franceschini, G., F., Tassoni, and Vas, P., "Integrated Condition Monitoring And Diagnosis Of Electrical Machines Using Minimum Configuration Artificial Intelligence", *EPE'97, Vol. 2, pp. 983-988, 1997.*

[44] Filippetti, C., Franceschini, G., F., Tassoni, and Vas, P., "AI Techniques In Induction Machines Diagnosis Including The Speed Ripple Effect", *IAS Annual Meeting, pp.655-662, October 1996.*

[45] Nuscheler, R., and Potoradi, D., "Problems and Failure Sources with the Parameter Identification of Asynchronous Machines and Their Dependence on the Identification Method", *ICEM'96, pp. 130-135, 1996.*

[46] Keyhani, A., and Miri, S.M., "Observers for Tracking of Synchronous Machine Parameters and Detection of Incipient Faults", *IEEE Trans. On Energy Conversion, Vol. EC-1, No. 2, pp. 184-192, 1996.*

[47] Chow, T.W.S, "Condition Monitoring Of Electrical Machines Using Third-Order Spectrum Analysis", *IEEE IAS Annual Meeting, Vol. 1, pp. 679-686, October 1996.*

[48] Cummings, P.B., Dunki-Jacobs, J.R., and Kerr, R.H., "Protection of Induction Motors Against Unbalanced Voltage Operation", *IEEE Trans. On Industry Applications, Vol. IA-21, No.4, pp.778-792, May/June 1985*

[49] Wagner, C.F., and Evans, R.D., "Symmetrical Components*" McGraw-Hill Book Company, Inc. New York, 1933.*

[50] Myatt, L.J., "Symmetrical Components", *Pergamon Press, 1968.*

[51] Clarke, E., "Circuit Analysis of AC Power Systems", *Volume 1'Symmetrical and Related Components', John Wiley and Sons, Inc.,*

[52] Tang, K.Y., "Alternating- Current Circuits", *International Textbook Comp.*

[53] Madhu, S., "Linear Circuit Analysis", *Prentice-Hall Englewood Cliffs, 1988*

[54] Bekku, T., "On Short Circuit Current of AC Generator", *JIEE Japan, pp. 805, 1927*

[55] Yamamura, S., "Spiral Vector Theory of AC Circuits and Machines*", Oxford Science Publications, 1992.*

[56] Kliman, G.B., "Turn Fault Detection", *United States Patent, 5,477,163, Dec. 1995*

[57] Jha, C.S., and Murthy, S.S., "Generalised Rotating-field Theory of Wound-rotor Induction Machines Having Asymmetry in Stator and/or Rotor Windings", *Proc. IEE, Vol. 120, No. 8, August 1973.*

[58] Brown, J.E., and Butler, O.I., "A General Method Of Analysis Of Three-Phase Induction Motors With Asymmetrical Primary Connections", *Proceeding IEE, Vol. 100, pp. 25-34, 1953*.

[59] Toliyat, H.A., and Lipo, T.A., "Transient Analysis Of Cage Induction Machines Under Stator, Rotor Bar End Ring Faults" *IEEE Trans. On Energy Conversion, Vol. 10, No. 2, pp. 241-247, June 1995*.

[60] Consoli, A., and Lipo, T.A., "Orthogonal Axis Models For Asymmetrically Connected Induction Machines", *IEEE Trans. On Power Apparatus And Systems, Vol. PAS-101, No. 12, pp. 4518-4526, December 1982*.

[61] Wallace, A.K., Ward, E.S., and Wright, A., "Sources of Harmonic Currents in Slip-ring Motors", *Proc. IEE, Vol. 121, No. 12, December 1974.*

[62] Novotny, D.W., and Lipo, T.A., "Vector Control and Dynamics of AC Drives", *Clarendon P. Oxford, 1996*.

[63] Alger, P.L., "Induction Machines: Their Behaviour and Uses", *Gordon & Breach, 2nd rev., 1970*.

[64] Ong, C.M., "Dynamic Simulation of Electric Machinery", *Prentice Hall PTR, 1998, ISBN 0-13-723785-5*.

[65] Krause, P., Wasynczuk, O., and Sudhoff, S.D., "Analysis of Electric Machinery", *IEEE Inc., 1995, ISBN 0-7803-1101-9*.

[66] Williamson, S., and Mirzoian, K., "Analysis of Cage Induction Motors With Stator Winding Faults", *IEEE Trans. On Power Apparatus and Systems, Vol. PAS-104, N0.7, pp.1838-1842, July 1985.*

[67] Vas, P., "Vector Control of AC Machines" *Oxford Science Publications, 1994*.

[68] Vas, P., "Electrical Machines and Drives: A Space-Vector Theory Approach" *Oxford Science Publications, 1992*.

[69] Lyons, R.G., "Understanding Digital Signal Processing", *Adisson-Wesley, 3rd Printing, 1998. ISBN 0-201-63467-8*.

[70] Agaki, H., and et al., "Instantaneous Reactive Power Compensators Comprising Switching Devices Without Energy Storage Components", *IEEE Trans. On Industry Applications, Vol. IA-20, pp.625-630, 1984.*

[71] Sobczyk, T.J., Vas, P., and Tassoni, C., "Model for Induction Motors with Air-Gap Asymmetry for Diagnostic purposes", *ICEM'96, Vol. 2, pp. 79-84, 1996*.

[72] Hsu, J.S., and Scoggins, B.R., "Field Test Of Motor Efficiency And Load Changes Through Air-Gap Torque", *IEEE Trans. on Energy Conversion, Vol. 10, No.3, pp.477-483, September 1995.*

[73] Matsui, M., and Fukao, T., "A Detecting Method For Active-Reactive-Negative-Sequence Powers And Its Application", *IEEE Trans. On Industry Applications, Vol. 26, No.1, pp. 99-106, January/February 1990.*

[74] Liu, Y., "Active Supply Harmonic Filtering", *University of Sussex, PhD Thesis 1999.*

[75] Analog Devices: "ADSP-2100 Family DSP Microcomputers Data Sheet", *Analog Devices, Inc., 1994*.

[76] Analog Devices: "ADSP-21xx EZ-ICE® Emulator Preliminary Release 5.x User's Manual", *Analog Devices, Inc., 1993*.

[77] Analog Devices: "Motion Coprocessor ADMC-200 Data Sheet", *Analog Devices, Inc., 1995*.

[78] Dister, C.J., and Schiferl, R., "Using Temperature, Voltage, and/or Speed Measurements To Improve Trending of Induction Motor RMS Currents In Process Control and Diagnostics", *IEEE IAS Annual Meeting, Vol. 1, pp. 312-318, October 1998*.

[79] Cash, M.A., and Habetler, T.G., "Insulation Failure Detection In The Stator Windings of ASD-Driven Induction Machines Using Standard Deviation of Line-Currents", *IEEE IAS Annual Meeting, Vol. 1, pp. 299-303, October 1998*.

[80] "User Manual for PC30F and PC30G Series Boards", *Eagle Technology, March 1996*

[81] Soderstrom, T. and Stoica, P., "System Identification" *Prentice Hall, London, 1989. (ISBN 0-13-881236-5)*

[82] Hsia, T.C, "System Identification-Least Square Methods" *Lexington Books, Massachusetts Toronto, 1977 (ISBN 0-669-99630-0)*.

[83] Williamson, S., and Begg, M.C., "Calculation of the Bar Resistance and Leakage Reactance of Cage Rotors with Closed Slots", *IEE Proceedings, Vol. 132, Pt. B, No. 3, pp. 125-132, May 1985.*

[84] Healey, R.C., Williamson, S., and Smith, A.C., "Improved Cage Rotor Models for Vector Controlled Induction Motors", *IEEE Trans. On Industry Applications, Vol. 31, No.4, pp. 812-822, July/August 1995.*

[85] Smith, A.C., Healey, R.C., and Williamson, S., "A Transient Induction Motor Model Including Saturation and Deep Bar Effect", *IEEE Trans. On Energy Conversion, Vol. 11, No.1, pp. 8-15, March 1996.*

[86] Williamson, S., and McClay, C.I., "Optimization of the Geometry of Closed Rotor Slots for Cage Induction Motors", *IEEE Trans. On Industry Applications, Vol. 32, No.3, pp. 560-568, May/June 1996.*

Appendix A

A.1 The Instantaneous Symmetrical Components [55, 68]

The method of instantaneous symmetrical components, first was introduced by Bekku [54], is closely related to the space-vector method. However, in contrast to the space vector method which utilizes only one type of a space-vector quantity and which could also be termed as a positive sequence space vector, instantaneous symmetrical component theory utilizes three quantities. These are the positive sequence, negative sequence, and zero sequence instantaneous symmetrical components respectively [68].

In the instantaneous symmetrical component theory, in general the positive (x_p), negative (x_n), and zero sequence (x_0) instantaneous symmetrical components of a general quantity x are defined as follows:

$$\begin{bmatrix} x_0 \\ x_p \\ x_n \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 \\ 1 & \alpha & \alpha^2 \\ 1 & \alpha^2 & \alpha \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} \qquad \text{A.1}$$

where $\alpha = e^{j\frac{2\pi}{3}}$, and, x_a, x_b, x_c are the instantaneous values of the given quantity (which can be a voltage or a current).

Since the three-phase $x_a, x_b,$ and x_c have real values, the following relation holds in Equation A.1

$$x_p = x_n^* \qquad \text{A.2}$$

The positive and negative sequence components are conjugate to each other and are not independent. At this point, Bekku's symmetrical component method is different from Fortescue's, whose positive and negative sequence components are not conjugate and are independent.

A.2 Spiral Vector and Spiral Vector Symmetrical Component Method [55]

A.2.1 Spiral Vector

Performance equations for electric circuits and machines can be written in the form of differential equations. Their general solutions can be written in terms of exponential functions of time as follows:

$$i = A\,\mathrm{e}^{\delta\,t} \tag{A.3}$$

where δ, $\delta = -\lambda + j\omega$, is complex and as time progresses i depicts a spiral in the complex plane. Yamamura call i a spiral vector.

When i represents a current, it can be expressed in various ways. For example, when i is a cosine function of time, we can write

$$i = |I|\cos(\omega\,t + \phi) \tag{A.4}$$

which also can be expressed as

$$i = \mathrm{Re}\left[|I|\,e^{j(\omega t + \phi)}\right] = \mathrm{Re}\left[\mathbf{I}\,e^{j\omega t}\right] = \mathrm{Re}\left[\dot{I}\right] \tag{A.5}$$

where $\mathrm{Re}[x]$ denotes the real part of x, and $\mathbf{I} = |I|\,e^{j\phi}$ is the phasor of i. We can also represent i by the following spiral vector

$$i = |I|e^{j(\omega t + \phi)} = \mathbf{I}\,e^{j\omega t} = \dot{I} \tag{A.6}$$

when $\lambda = 0$, the spiral vector becomes the vector of Equation A.6, which depicts a circle in the complex plane and represents steady-state AC current, and

$$|I| = |\dot{I}| \tag{A.7}$$

A.2.2 Spiral Vector Symmetrical Component Method

In spiral form, the asymmetrical three-phase currents can be written as follows

$$x_a = \left|\dot{X}_0\right| e^{j(\omega t+\phi_0)} + \left|\dot{X}_p\right| e^{j(\omega t+\phi_p)} + \left|\dot{X}_n\right| e^{j(\omega t+\phi_n)}$$

$$x_b = \left|\dot{X}_0\right| e^{j(\omega t+\phi_0)} + \left|\dot{X}_p\right| e^{j(\omega t+\phi_p-\frac{2\pi}{3})} + \left|\dot{X}_n\right| e^{j(\omega t+\phi_n+\frac{2\pi}{3})} \qquad \text{A.8}$$

$$x_c = \left|\dot{X}_0\right| e^{j(\omega t+\phi_0)} + \left|\dot{X}_p\right| e^{j(\omega t+\phi_p-\frac{4\pi}{3})} + \left|\dot{X}_n\right| e^{j(\omega t+\phi_n+\frac{4\pi}{3})}$$

Equation A.1 transforms them into the following components

$$x_0 = \left|\dot{X}_0\right| e^{j(\omega t+\phi_0)}, \quad x_p = \left|\dot{X}_p\right| e^{j(\omega t+\phi_p)}, \quad x_n = \left|\dot{X}_n\right| e^{j(\omega t+\phi_n)} \qquad \text{A.9}$$

These terms are independent of each other, and Fortescue's symmetrical components $\left|\dot{X}_0\right|$, $\left|\dot{X}_p\right|$, and $\left|\dot{X}_n\right|$ are separated in each of the symmetrical components x_0, x_p, and x_n. For the transient state, variables are expressed as spiral vectors.

Appendix B

Software Correction for Non-Simultaneous Sampling

The phase voltages in a three phase supply can be written as a summation of positive and negative sequence voltages as follows

$$x = u_p + u_n$$

$$y = u_p \cdot e^{-j\frac{2\pi}{3}} + u_n \cdot e^{j\frac{2\pi}{3}} \qquad \text{B.1}$$

$$z = u_p \cdot e^{-j\frac{4\pi}{3}} + u_n \cdot e^{j\frac{4\pi}{3}}$$

x, y and z are the physical phase voltages and u_p and u_n are positive and negative sequence component phasor voltages. For non simultaneous sampling, with a delay angle δ between samples, x, y, and z, would be measured as

$$\bar{x} = x$$

$$\bar{y} = y \cdot e^{j\delta} \qquad \text{B.2}$$

$$\bar{z} = z \cdot e^{j2\delta}$$

where the bar indicates signals derived by non simultaneous sampling, and $\delta = 2\pi f_s\, t_{delay}$.

From these three measured phasors $\bar{x}, \bar{y}$ and $\bar{z}$, positive and negative sequence voltages can be derived using classical symmetrical components theory as

$$\bar{u}_p = \frac{\bar{x} + \alpha\bar{y} + \alpha^2\bar{z}}{3} \qquad \text{B.3}$$

$$\bar{u}_n = \frac{\bar{x} + \alpha^2\bar{y} + \alpha\bar{z}}{3} \qquad \text{B.4}$$

where $\alpha = e^{j\frac{2\pi}{3}}$. Equations B.3 and B.4 can be expended to:

$$\bar{u}_p = \frac{u_p}{3}(1 + e^{j\delta} + e^{j2\delta}) + \frac{u_n}{3}(1 + \alpha^2 e^{j\delta} + \alpha e^{j2\delta}) \qquad \text{B.5}$$

$$\bar{u}_n = \frac{u_n}{3}(1 + e^{j\delta} + e^{j2\delta}) + \frac{u_p}{3}(1 + \alpha\ e^{j\delta} + \alpha^2 e^{j2\delta}) \qquad \text{B.6}$$

$\bar{u}_p$ and $\bar{u}_n$ are measured values and u_p and u_n are actual values.

When δ is zero (simultaneous sampling), $\bar{u}_p$ and $\bar{u}_n$ equal u_p and u_n respectively, since $(1 + e^{j0} + e^{j2\cdot 0}) = 3$, $(1 + \alpha^2 e^{j0} + \alpha\ e^{j2\cdot 0}) = 0$ and $(1 + \alpha\ e^{j0} + \alpha^2 e^{j2\cdot 0}) = 0$.

By using Maclaurin series expansion of e^{js},

$$e^{js} = 1 + js - \frac{s^2}{2!} - j\frac{s^3}{3!} + \frac{s^4}{4!} + \dots\dots\dots \qquad \text{B.7}$$

the first bracketed term in Equations B.5 and B.6 can be written as

$$\begin{aligned} 1 + e^{j\delta} + e^{j2\delta} &= 1 + (1 + j\delta) + (1 + j2\delta) \\ &= 3 + j3\delta \\ &= 3e^{j\delta} \end{aligned} \qquad \text{B.8}$$

Terms with δ^n *(n=2,3,…)* terms have been ignored, because δ^n <<0 when n≥2. The second bracketed terms in Equations B.5 and B.6 can be simplified as

$$\begin{aligned} 1 + \alpha^2 e^{j\delta} + \alpha\ e^{j2\delta} &= 1 + (-\frac{1}{2} - j\frac{\sqrt{3}}{2}) \cdot (1 + j\delta) + (-\frac{1}{2} + j\frac{\sqrt{3}}{2}) \cdot (1 + j2\delta) \\ &= \delta \cdot (-\frac{\sqrt{3}}{2} - j\frac{3}{2}) \\ &= -\sqrt{3}\delta e^{j\frac{\pi}{3}} \end{aligned} \qquad \text{B.9}$$

and

$$1+\alpha e^{j\delta}+\alpha^2 e^{j2\delta}=1+(-\frac{1}{2}+j\frac{\sqrt{3}}{2})\cdot(1+j\delta)+(-\frac{1}{2}-j\frac{\sqrt{3}}{2})\cdot(1+j2\delta)$$

$$=\delta\cdot(\frac{\sqrt{3}}{2}-j\frac{3}{2}) \quad \text{B.10}$$

$$=\sqrt{3}\delta e^{-j\frac{\pi}{3}}$$

by using the results of Eqs-8,9, and 10, Eq-5 and Eq-6 can be rewritten as

$$\bar{u}_p=\frac{u_p}{3}\cdot 3e^{j\delta}-\frac{u_n}{3}\cdot\sqrt{3}\delta e^{j\frac{\pi}{3}}=u_p\cdot e^{j\delta}-\frac{1}{\sqrt{3}}u_n\cdot\delta\cdot e^{j\frac{\pi}{3}}$$

$$\bar{u}_p=u_p e^{j\delta}-k\delta u_n e^{j\frac{\pi}{3}} \quad \text{B.11}$$

and

$$\bar{u}_n=\frac{u_n}{3}\cdot 3e^{j\delta}+\frac{u_p}{3}\cdot\sqrt{3}\delta e^{-j\frac{\pi}{3}}=u_n\cdot e^{j\delta}+\frac{1}{\sqrt{3}}u_p\cdot\delta\cdot e^{-j\frac{\pi}{3}}$$

$$\bar{u}_n=u_n e^{j\delta}+k\delta u_p e^{-j\frac{\pi}{3}} \quad \text{B.12}$$

where $k=\frac{1}{\sqrt{3}}$.

Equation B.11 and B.12 can then be rearranged as

$$u_p=\bar{u}_p e^{-j\delta}+k\delta u_n e^{j(\frac{\pi}{3}-\delta)} \quad \text{B.13}$$

$$u_n=\bar{u}_n e^{-j\delta}-k\delta u_p e^{-j(\frac{\pi}{3}+\delta)} \quad \text{B.14}$$

and by putting Equation B.14 into B.13 the real positive sequence voltage can be found as

$$u_p = \overline{u}_p e^{-j\delta} + k\delta e^{j(\frac{\pi}{3}-\delta)}\left(\overline{u_n} e^{-j\delta} - k\delta u_p e^{-j(\frac{\pi}{3}+\delta)}\right)$$

$$u_p = \overline{u}_p e^{-j\delta} + k\delta \overline{u}_n e^{j(\frac{\pi}{3}-2\delta)} - k^2\delta^2 u_p e^{-j2\delta}$$

$$u_p = \beta \overline{u}_p e^{-j\delta} + \beta\, k\delta \overline{u}_n e^{j(\frac{\pi}{3}-2\delta)} \qquad \text{B.15}$$

where $\beta = \dfrac{1}{1 + k^2\delta^2 e^{-j2\delta}} \cong 1.$

By using Equation B.15 negative sequence voltage can be found as

$$u_n = \overline{u}_n e^{-j\delta} - k\delta e^{-j(\frac{\pi}{3}+\delta)}\left(\beta \overline{u}_p e^{-j\delta} + \beta\, k\delta \overline{u}_n e^{j(\frac{\pi}{3}-2\delta)}\right)$$

$$u_n = \overline{u}_n e^{-j\delta} - \beta\, k\delta \overline{u}_p e^{-j(\frac{\pi}{3}+2\delta)} - \beta\, k^2\delta^2 \overline{u}_n e^{-j3\delta}$$

$$u_n = \overline{u}_n e^{-j\delta}\left(1 - \beta\, k^2\delta^2 e^{-j2\delta}\right) - \beta\, k\delta \overline{u}_p e^{-j(\frac{\pi}{3}+2\delta)}$$

$$u_n = \beta \overline{u}_n e^{-j\delta} - \beta\, k\delta \overline{u}_p e^{-j(\frac{\pi}{3}+2\delta)} \qquad \text{B.16}$$

Equation B.15 and B.16 are equations by which actual values of negative and positive sequence voltages may be recovered from negative and positive sequence voltages measured using non-simultaneous voltage sampling. It is obvious that when δ is zero (simultaneous sampling) u_p and u_n (actual values) are equal to $\overline{u}_p$ and $\overline{u}_n$ (measured values) respectively. By using these equations, non-simultaneous errors can be virtually eliminated.

Appendix C

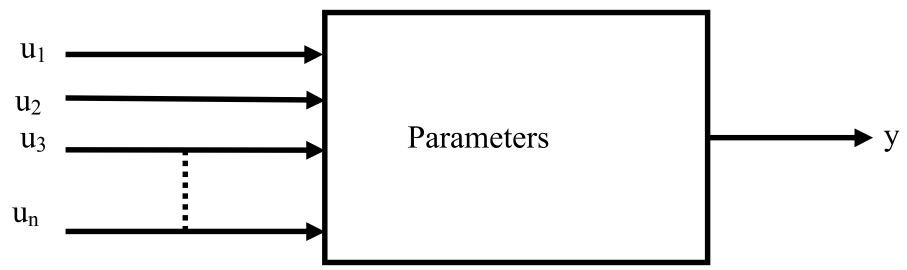
Recursive Least Squares Identification Method [81, 82]

In recursive (also called on-line) identification methods, parameter estimates are computed recursively in real time. This means that if there is an estimate $\theta(t-1)$ based on data up to time *t-1*, then $\theta(t)$ is computed by some 'simple modification' of $\theta(t-1)$. The counterparts to on-line methods are the so-called off-line or batch methods, in which all the recorded data is used simultaneously to estimate the parameters.

Most adaptive systems (e.g. adaptive control systems) are based explicitly or implicitly on recursive identification. Then an up to date estimated model of the process is available at all times.

Recursive identification methods require only a modest amount of information to be stored, which does not increase with time.

The Recursive Least Squares Method



To start the formal discussion of the recursive least squares (RLS) method, consider the scalar system (dim y=1) given by

$$y(t) = \varphi^T(t)\theta + \varepsilon(t) \qquad \text{C.1}$$

Where $y(t)$ is a measurable output quantity (the apparent fault current I_{zmn}), $\varphi^T(t) = \begin{bmatrix} u_1 & ..... & u_n \end{bmatrix}$ is an n-vector of measured quantities given by $\varphi^T(t) = \begin{bmatrix} 1 & I_{px} & I_{px}^2 & I_{py} & I_{py}^2 \end{bmatrix}$ for our curve fitting. $\theta = \begin{bmatrix} \alpha_0 & ..... & \alpha_{n-1} \end{bmatrix}^T$ is an n-vector of the unknown parameters to be found, and $\varepsilon(t)$ is scalar error.

Then the parameter estimate is given by

$$\overline{\theta}(t)_{n\times 1} = \left[\sum_{s=1}^{t} \varphi(s)\varphi^T(s)\right]_{n\times n}^{-1} \left[\sum_{s=1}^{t} \varphi(s)y(s)\right]_{n\times 1} \qquad \text{C.2}$$

The argument t has been used to stress the dependence of $\overline{\theta}$ on time. The expression C.2 can be computed in a recursive fashion. Introducing the notation

$$P(t) = \left[\sum_{s=1}^{t} \varphi(s)\varphi^T(s)\right]_{n\times n}^{-1} \qquad \text{C.3}$$

and since trivially

$$P^{-1}(t) = P^{-1}(t-1) + \varphi(t)\varphi^T(t) \qquad \text{C.4}$$

it follows that

$$\overline{\theta}(t) = P(t)\left[\sum_{s=1}^{t} \varphi(s)y(s) + \varphi(t)y(t)\right]$$

$$= P(t)\left[P^{-1}(t-1)\overline{\theta}(t-1) + \varphi(t)y(t)\right]$$

$$= \overline{\theta}(t-1) + P(t)\varphi(t)\left[y(t) - \varphi^T(t)\overline{\theta}(t-1)\right]$$

Thus

$$\overline{\theta}(t) = \overline{\theta}(t-1) + K(t)\varepsilon(t) \qquad \text{C.5a}$$

$$K(t) = P(t)\varphi(t) \qquad \text{C.5b}$$

$$\varepsilon(t) = y(t) - \varphi^T(t)\overline{\theta}(t-1) \qquad \text{C.5c}$$

$K(t)$ is $n \times 1$ dimension.

Here the term $\varepsilon(t)$ should be interpreted as a prediction error. It is the difference between the measured output $y(t)$ and the one-step-ahead prediction $\overline{y}(t|t-1) = \varphi^T(t)\overline{\theta}(t-1)$ of *y(t)* made at time *t - 1* based on the model corresponding to the estimate of the parameters $\overline{\theta}(t-1)$ at time (*t - 1*). If $\varepsilon(t)$ is small, the estimate $\overline{\theta}(t-1)$ is 'good' and should not be modified very much. The vector *K(t)* in Equation C.5b should be interpreted as a weighting or gain factor showing how much the value of $\varepsilon(t)$ will modify the different elements of the parameter vector.

To complete the algorithm, Equation C.4 must be used to compute *P(t)* which is needed in C.5b. However, the use of Equation C.4 needs a matrix inversion at each time step. This would be a time-consuming procedure. Using a matrix inversion lemma, however, Equation C.4 can be rewritten in a more useful form, giving an updating equation for *P(t)*, namely

$$P(t) = P(t-1) - P(t-1)\varphi(t)\varphi^T(t)P(t-1) / \left[1 + \varphi^T(t)P(t-1)\varphi(t)\right] \qquad \text{C.6}$$

Note that in Equation C.6 there is now a scalar division instead of a matrix inversion. The algorithm consisting of Equation C.5 - C.6 can be simplified further. From Equations C.5b, C.6,

$$K(t) = P(t-1)\varphi(t) / \left[1 + \varphi^T(t)P(t-1)\varphi(t)\right] \qquad \text{C.7}$$

This form for *K(t)* is more convenient to use for implementation than Equation C.5b. The reason is that the right-hand side of Equation C.7 must anyway be computed in the updating of *P(t)* (see Equation C.6).

The derivation of the recursive least squares (RLS) method is now complete. The RLS algorithm consists of Equations C.5a, C.5c, C.6 and C.7. The algorithm also needs initial values $\bar{\theta}(0)$ and *P(0)*. If *P(0)* is small then *K(t)* will be small for all *t* and the parameter estimates will therefore not change too much from $\bar{\theta}(0)$. On the other hand, if *P(0)* is large, the parameter estimates will quickly jump away from $\bar{\theta}(0)$. Without any a *priori* information it is common practice to take

$$\bar{\theta}(0) = 0 \quad P(0) = \xi \cdot I \tag{C.8}$$

where ξ is a 'large' number and *I* is the identity matrix. Beginning from Equation C.6 , C.7, C.5c and C.5a, the RLS algorithm can easily be applied.

Summary

The RLS algorithm can be summarised as below, with matrices defined by

$\theta_{n\times 1}$ is the set of parameters to be determined (e.g. five α values)

$P_{n\times n}$ is a matrix used in the calculation.

$I_{n\times n}$ is the identity matrix, ξ is a large number e.g. 1000.

$\varphi_{n\times 1}(t)$ is the t^th^ set of input data $\begin{bmatrix}1 & I_{px} & I_{px}^2 & I_{py} & I_{py}^2\end{bmatrix}^T$

$y(t)$ is the t^th^ output value (fault current I_{mn})

$\varepsilon(t)$ is a scalar error in the fitted t^th^ output value.

$\theta(t)$ is the t^th^ estimate of the parameter set.

Step 1: $\bar{\theta}(0) = 0_{n\times 1} \qquad P(0) = \xi \cdot I_{n\times n}$

Step 2: Read *n x 1* input vector $\varphi(t)$, and output value $y(t)$

Step 3: $A(t) = 1 + \varphi^T(t)P(t-1)\varphi(t)$

Step 4 $P(t) = P(t-1) - P(t-1)\varphi(t)\varphi^T(t)P(t-1)\,/\,A(t)$

Step 5: $K(t) = P(t-1)\varphi(t) / A(t)$

Step 6: $\varepsilon(t) = y(t) - \varphi^T(t)\overline{\theta}(t-1)$

Step 7: $\overline{\theta}(t) = \overline{\theta}(t-1) + K(t)\varepsilon(t)$

Step 8: Set new $\overline{\theta}(t-1) = \overline{\theta}(t)$ and $P(t-1) = P(t)$

Step 9: If $\varepsilon(t)$ is small enough finish otherwise go to Step 2

Appendix D

D.1 Matrix Elements for Induction Motor with Different Numbers of Stator Turns

D.1.1 The Stator qd0 Resistance Matrix

The *qd0* resistances are obtained by $\mathbf{r}_{qd0}^{s} = \mathbf{T}_{qd0}(0)\mathbf{r}_{abc}^{s}\mathbf{T}_{qd0}(0)^{-1}$ transformation and matrix elements are:

$$r_{11}^{s} = \frac{2}{3}(r_{as} + \frac{1}{4}r_{bs} + \frac{1}{4}r_{cs}) \quad \text{D.1}$$

$$r_{12}^{s} = \frac{\sqrt{3}}{6}(r_{bs} - r_{cs}) \quad \text{D.2}$$

$$r_{13}^{s} = \frac{1}{3}(2r_{as} - r_{bs} - r_{cs}) \quad \text{D.3}$$

$$r_{21}^{s} = r_{12}^{s}$$

$$r_{22}^{s} = \frac{1}{2}(r_{bs} + r_{cs}) \quad \text{D.4}$$

$$r_{23}^{s} = -\frac{1}{2}r_{12}^{s}$$

$$r_{31}^{s} = \frac{1}{2}r_{13}^{s}$$

$$r_{32}^{s} = -r_{12}^{s}$$

$$r_{33}^{s} = \frac{1}{3}(r_{as} + r_{bs} + r_{cs}) \qquad \text{D.5}$$

where $r_{as} = \frac{N_a}{N_s} r_s$, $r_{bs} = \frac{N_b}{N_s} r_s$, and $r_{cs} = \frac{N_c}{N_s} r_s$

When $N_a = N_b = N_c = N_s$, $r_{11}^{s} = r_{22}^{s} = r_{33}^{s} = r_s$ and $r_{12}^{s} = r_{13}^{s} = r_{21}^{s} = r_{23}^{s} = r_{31}^{s} = r_{32}^{s} = 0$

D.1.2 The Stator qd0 Self and Mutual Inductances

Transformation of $\mathbf{L}_{abc}^{ss}$ to $\mathbf{L}_{qd0}^{ss}$ is given by $\mathbf{L}_{qd0}^{s} = \mathbf{T}_{qd0}(0)\mathbf{L}_{abc}^{ss}\mathbf{T}_{qd0}(0)^{-1}$ and matrix elements are:

$$L_{11}^{ss} = \frac{2}{3}(L_{asas} + .25L_{bsbs} + .25L_{\csc s} - L_{asbs} - L_{ascs} + .5L_{bscs}) \qquad \text{D.6}$$

$$L_{12}^{ss} = \frac{1}{2\sqrt{3}}(L_{bsbs} - L_{\csc s} - L_{asbs} + L_{ascs}) \qquad \text{D.7}$$

$$L_{13}^{ss} = \frac{2}{3}(L_{asas} - .5L_{bsbs} - .5L_{\csc s} + .5L_{asbs} + .5L_{ascs} - L_{bscs}) \qquad \text{D.8}$$

$$L_{21}^{ss} = \frac{1}{\sqrt{3}}(.5L_{bsbs} - .5L_{\csc s} - L_{asbs} + L_{ascs}) \qquad \text{D.9}$$

$$L_{22}^{ss} = \frac{1}{2}(L_{bsbs} + L_{\csc s} - 2L_{bscs}) \qquad \text{D.10}$$

$$L_{23}^{ss} = \frac{1}{\sqrt{3}}(-L_{bsbs} + L_{\csc s} - L_{asbs} + L_{ascs}) \qquad \text{D.11}$$

$$L_{31}^{ss} = \frac{1}{2}L_{13}^{ss}$$

$$L_{32}^{ss} = \frac{1}{2}L_{23}^{ss}$$

$$L_{33}^{ss} = \frac{1}{3}(L_{asas} + L_{bsbs} + L_{\csc s} + 2L_{asbs} + 2L_{ascs} + 2L_{bscs}) \qquad \text{D.12}$$

When $N_a = N_b = N_c = N_s$, $L_{asas} = L_{bsbs} = L_{\csc s}$ and $L_{asbs} = L_{ascs} = L_{bscs}$. And by using Equation 4.25-to-4.33 it can be shown that, $L_{12}^{ss} = L_{13}^{ss} = L_{21}^{ss} = L_{23}^{ss} = L_{31}^{ss} = L_{32}^{ss} = 0$ and $L_{11}^{ss} = L_{22}^{ss} = L_{ls} + L_m$, $L_{33}^{ss} = L_{ls}$.

D.1.3 The Stator-to-Rotor qd0 Mutual Inductances

The stator-to-rotor mutual inductances are given by transformation of $\mathbf{T}_{qd0}(0)\mathbf{L}_{abc}^{sr}\mathbf{T}_{qd0}(-\theta_r)^{-1}$. Because of rotor symmetry, the coefficients of stator-to-rotor mutual inductances can be simplified as $L_{asar} = L_{asbr} = L_{ascr} = L_{asr}$, $L_{bsar} = L_{bsbr} = L_{bscr} = L_{bsr}$, and $L_{csar} = L_{csbr} = L_{\csc r} = L_{csr}$, and transformation result will be

$$L_{11}^{sr} = L_{asr} + .25L_{bsr} + .25L_{csr} \qquad \text{D.13}$$

$$L_{12}^{sr} = \frac{\sqrt{3}}{4}(L_{bsr} - L_{csr}) \qquad \text{D.14}$$

$$L_{21}^{sr} = L_{12}^{sr}$$

$$L_{22}^{sr} = \frac{3}{4}(L_{bsr} + L_{csr}) \qquad \text{D.15}$$

$$L_{31}^{sr} = .5L_{asr} - .25L_{bsr} - .25L_{csr} \qquad \text{D.16}$$

$$L_{32}^{sr} = -L_{12}^{sr}$$

When $N_a = N_b = N_c = N_s$, $L_{asar} = L_{asbr} = L_{ascr} = L_{bsar} = L_{bsbr} = L_{bscr} = L_{csar} = L_{csbr} = L_{\csc r}$, mutual inductances will be $L_{11}^{sr} = L_{22}^{sr} = \frac{N_r}{N_s}L_m$ and $L_{12}^{sr} = L_{21}^{sr} = L_{31}^{sr} = L_{32}^{sr} = 0$.

D.1.4 The Rotor qd0 Self and Mutual Inductances

The rotor self and mutual inductances are

$$L_{11}^{rr} = L_{22}^{rr} = L_{lr} + \frac{3}{2} L_{mar} = L_{lr} + \frac{{N_r}^2}{{N_s}^2} L_m \qquad \text{D.17}$$

$$L_{33}^{rr} = L_{lr} \qquad \text{D.18}$$

D.1.5 The Rotor qd0 Resistances for Fault Condition

The *qd0* resistances for rotor are obtained by $\mathbf{r}_{qd0}^{r} = \mathbf{T}_{qd0}(-\theta_r)\mathbf{r}_{abc}^{s}\mathbf{T}_{qd0}(-\theta_r)^{-1}$ transformation and matrix elements are:

$$r_{11}^{r} = \frac{2}{3}\left[r_{ar}\cos^2(\theta_r) + r_{br}\cos^2\left(\theta_r + \frac{2\pi}{3}\right) + r_{cr}\cos^2\left(\theta_r - \frac{2\pi}{3}\right)\right] \qquad \text{D.19}$$

$$r_{12}^{r} = -\frac{1}{3}\left[r_{ar}\sin(2\theta_r) + r_{br}\sin\left(2\left(\theta_r + \frac{2\pi}{3}\right)\right) + r_{cr}\sin\left(2\left(\theta_r - \frac{2\pi}{3}\right)\right)\right] \qquad \text{D.20}$$

$$r_{13}^{r} = \frac{2}{3}\left[r_{ar}\cos\theta_r + r_{br}\cos\left(\theta_r + \frac{2\pi}{3}\right) + r_{cr}\cos\left(\theta_r - \frac{2\pi}{3}\right)\right] \qquad \text{D.21}$$

$$r_{21}^{r} = r_{12}^{r}$$

$$r_{22}^{r} = \frac{2}{3}\left[r_{ar}\sin^2(\theta_r) + r_{br}\sin^2\left(\theta_r + \frac{2\pi}{3}\right) + r_{cr}\sin^2\left(\theta_r - \frac{2\pi}{3}\right)\right] \qquad \text{D.22}$$

$$r_{23}^{r} = \frac{2}{3}\left[r_{ar}\sin\theta_r + r_{br}\sin\left(\theta_r + \frac{2\pi}{3}\right) + r_{cr}\sin\left(\theta_r - \frac{2\pi}{3}\right)\right] \qquad \text{D.23}$$

$$r_{31}^{r} = \frac{1}{2} r_{13}^{r}$$

$$r_{32}^{r} = \frac{1}{2} r_{23}^{r}$$

$$r_{33}^{r} = \frac{1}{3}\left(r_{ar} + r_{br} + r_{cr}\right) \qquad \text{D.24}$$

When $r_{ar} = r_{br} = r_{cr} = r_r$ and by using trigonometric relation it can be shown that $r_{11}^{r} = r_{22}^{r} = r_{33}^{r} = r_r$ and $r_{12}^{r} = r_{13}^{r} = r_{21}^{r} = r_{23}^{r} = r_{31}^{r} = r_{32}^{r} = 0$.

D.2 Matrix Elements for Induction Motor with Inter-Turn Stator Short Circuit

D.2.2 The Stator qd Self and Mutual Inductances

From Equation D.6 and D.10, the stator qd self and mutual inductances can be defined as follows

$$\begin{aligned} L_{11}^{ss} &= \frac{2}{3}(L_{asas} + .5L_{bsbs} - 2L_{asbs} + .5L_{bscs}) \\ &= \frac{2}{3}\left[\left(L'_{asas} + .5L_{bsbs} - 2L'_{asbs}\right) + \left(L_{assh} - L_{shbs}\right)\right] + \frac{2}{3}\left[L_{shsh} + \left(L_{assh} - L_{shbs}\right)\right] \\ &= \left(L_{q}^{s} + L_{q}^{ssh}\right) + \left(L_{q}^{sh} + L_{q}^{ssh}\right) \end{aligned} \qquad \text{D.25}$$

$$L_{22}^{ss} = \frac{1}{2}(L_{bsbs} + L_{\mathrm{csc}s} - 2L_{bscs}) = L_{bsbs} - L_{bscs} = L_{ls} + L_{m} = L_{d}^{s} \qquad \text{D.26}$$

D.2.2 The Stator-to-Rotor qd Mutual Inductances

From Equation D.13 and D.15, the stator-to-rotor *qd* mutual inductances can be defined as follows

$$\begin{aligned} L_{11}^{sr} &= L_{asr} + .25L_{bsr} + .25L_{csr} = L_{asr} + .5L_{bsr} \\ &= \left(L'_{asr} + .5L_{bsr}\right) + L_{shar} \\ &= L_q^{sr} + L_q^{shr} \end{aligned} \qquad \text{D.27}$$

$$L_{22}^{sr} = \frac{3}{4}(L_{bsr} + L_{csr}) = \frac{3}{2}L_{bsr} = L_d^{sr} \qquad \text{D.28}$$

The rotor self and mutual inductances and resistances will be the same as before.

D.3 Motor Parameters

Reliance motor parameters:

$r_s = 4.05\ \Omega$	Stator winding resistance
$L_s = 13.974\ \text{mH}$	Stator leakage inductance
$r_r = 2.6\ \Omega$	Rotor resistance (stator referred)
$L_r = 13.974\ \text{mH}$	Rotor leakage inductance (stator referred)
$L_m = 538.68\ \text{mH}$	Magnetising inductance
$P = 4$	Number of poles
$J = 0.06\ \text{kg m}^2$	Rotor inertia

Appendix E

E.1 Assembler Program for ADSP2101 DSP Processor

```
{=====================================================================}
{====    This assembler program finds positive and negative sequence    ======}
{====    voltage and current components from sampled motor currents     ======}
{====    and supply voltages. The pre-processed results are sent to a PC ======}
{====    from the DSP by serial communication.                          ======}
{=====================================================================}

.MODULE/RAM/ABS=0       PDT02;
.INCLUDE <REG.H>;                {Include ADSP-2101 Memory mapped Register Addresses}
.INCLUDE <PORTS.H>;              {Include Ports name for reading and writing information}
.INCLUDE <CONST128.H>;           {Include all constants that are used by program}
.INCLUDE <NVARIAB.H>;            {Include all variables(for store data to data or memory) }


      {====       INTERRUPT VECTOR TABLE          ====}
      {==== This section is for interrupt subroutines. ====}
      {==== All interrupt are in priority order       ====}

          JUMP STARTM;RTI;NOP;NOP;NOP;       {System Reset}
          RTI;NOP;NOP;NOP;
          JUMP SPORT0T;RTI;NOP;NOP;NOP;      {Sport0 Transmit interrupt}
          RTI;NOP;NOP;NOP;
          JUMP ADMCINT;RTI;NOP;NOP;          {IRQ1is used for ADMC Interrupt}
          RTI;NOP;NOP;NOP;
          JUMP TIME;RTI;NOP;NOP;NOP;         {Timer interrupt}

{========            MAIN PROGRAM            ========}

STARTM:   CALL START;            {Call Initialization Subroutine }
          ENA TIMER;             {Enable Timer for Serial communication}
          ICNTL=0x0007;          {Set Interrupt control register for IRQ0, IRQ1 and IRQ2}
          IMASK=0x0015;          {Timer, IRQ1 and SPORT0 Transmit enable}
GO_IDLE:  IDLE;                  {Waiting for interrupt}
          NOP;
          JUMP GO_IDLE;

{====          ADMC INTERRUPT ORPERATION CODE           ====}

ADMCINT: AX0=DM(ADMC_STAT);      { ADMC interrupt checking}
          AY0=V_INT_STAT;        {test if the interrupt is     }
          AR=AX0 AND AY0;        {caused by vector section }
          IF NE RTI;             { yes, then return }
```

```
            AX0=DM(V_FLAG);
            AY0=DM(I_FLAG);
            AR=AX0 AND AY0;                 {Check Voltage and current sampling are completed}
            IF NE RTI                       {If yes return from ADMCINT interrupt}

{=== Produce Reference Voltage and Current              ===}

AX0=DM(ANGLE);                              {Angle initially is taken zero}
            DM(ANGLE1)=AX0;
            CALL SIN_SUB;                   {Call subroutine SIN_SUB for find sin ωt }
            SR=ASHIFT AR BY -4 (HI);        {Shift result four bits right for convert to integer}
            DM(SINA)=SR1;                   {Save fourteen bits integer result }
            AX0=DM(ANGLE);                  {Find  cos ωt as  sin ωt }
            AY0=0X4000;                     {0X4000 is equal to 90 degrees }
            AR=AX0+AY0;                     { cos ωt = sin(ωt + 90) }
            AX0=AR;
            CALL SIN_SUB;                   {Calculate sine function}
            SR=ASHIFT AR BY -4 (HI);        {Shift result four bits right for convert to integer}
            DM(COSA)=SR1;                   {Save fourteen bits integer result }
            AX0=DM(ANGLE);
            AY0=0X0200;                     {Increase angle by 2.8125 degree for next calculation}
            AR=AX0+AY0;
            DM(ANGLE)=AR;                   {Save result to memory}

{For different sampling frequency, for example for 6.4kHz, 3.2kHz and 3.6kHz, angle have to be increase
as follows:
        for 128 samples: ANGLE + 0x0200
        for  64 samples:     "     + 0x0400
        for  72 samples:     "     + 0x038E  }

            AX0=DM(SMPL_FLAG);              {Load the flag for checking reading is for current }
            AR=PASS AX0;                    {or voltage                             }
            IF EQ JUMP READ_CRENT;          { is for current or not?}

        {==   Read Sampled Voltages and Find the Power due to Voltages     ===}

        AX0=DM(ADC_U);                      {Read phase-A voltage from A/D}
            AY0=0XFFF0;                     {Remove last four bits that A/D is twelve bits}
            AR=AX0 AND AY0;
            AR=-AR;                         {Invert signal to back from hardware reason}
            SR=ASHIFT AR BY -4 (HI);        {Shift four bits to right for convert 12 bits integer}
            DM(U_LOAD)=SR1;                 {Save result to memory}
            AX0=DM(ADC_V);                  {Read phase-B voltage from A/D}
            AR=AX0 AND AY0;                 {Remove last four bits that A/D is twelve bits}
            AR=-AR;                         {Invert signal to back from hardware reason}
            SR=ASHIFT AR BY -4 (HI);        {Shift four bits to right for convert 12 bits integer}
            DM(V_LOAD)=SR1;                 {Save result to memory}
            AX0=DM(ADC_W);                  {Read phase-B voltage from A/D}
            AR=AX0 AND AY0;                 {Remove last four bits that A/D is twelve bits}
            AR=-AR;                         {Invert signal to back from hardware reason}
            SR=ASHIFT AR BY -4 (HI);        {Shift four bits to right for convert 12 bits integer}
            DM(W_LOAD)=SR1;                 {Save result to memory}

        {===          Do 3/2 Transformation            ===}

            AX0=DM(V_LOAD);                 {Load Vb}
            AY0=DM(W_LOAD);                 { Load Vc }
```

```
AR=AX0+AY0;               { $V_b+V_c$ }
IF AV CALL OV_ADD;        {If overflow exit call subroutine OV_ADD}
AR=-AR;                   { $-(V_b+V_c)$ }
MY0=0X4000;               { MY0=1/2}
MR=AR*MY0 (RND);          { $-1/2(V_b+V_c)$ }
IF MV SAT MR;             {if overflow exits saturate result}
AY0=DM(U_LOAD);           { Load $V_a$ }
AR=MR1+AY0;               { $V_a-1/2(V_b+V_c)$ }
IF AV CALL OV_ADD;
MY0=COEFF2;               { COEFF2=2/3 }
MR=AR*MY0 (RND);          {$2/3(V_a-1/2(V_b+V_c))$ }
IF MV SAT MR;
DM(VX_LOAD)=MR1;          {Save $V_\alpha$ }
AY0=DM(W_LOAD);           { Load $V_c$ }
AR=AX0-AY0;               { $V_b-V_c$ }
IF AV CALL OV_ADD;
MY0=COEFF1;               {COEFF1=$1/\sqrt{3}$ }
MR=AR*MY0 (RND);          {$1/\sqrt{3}(V_b-V_c)$ }
IF MV SAT MR;
DM(VY_LOAD)=MR1;          { Save $V_\beta$}

{===        Find Powers        ===}

MY0=DM(COSA);             {Load reference voltages}
MY1=DM(SINA);
MX0=DM(VX_LOAD);          {Load supply voltages}
MX1=DM(VY_LOAD);
ENA M_MODE;               {Enable Integer Mode }
MR=MX0*MY0 (SS);          { $P_\alpha^v = V_\alpha \cos(wt)$ }
MR=MR+MX1*MY1 (SS);       { $P_{\alpha p}^v = P_\alpha^v + P_\beta^v = V_\alpha \cos(wt) + V_\beta \sin(wt)$ }
DM(PAPH)=MR1;             {Save positive active power $P_{\alpha p}^v$ (result is 32 bits)}
DM(PAPL)=MR0;
MR=MX0*MY0 (SS);          { $P_\alpha^v = V_\alpha \cos(wt)$ }
MR=MR-MX1*MY1 (SS);       { $P_{\alpha n}^v = P_\alpha^v - P_\beta^v = V_\alpha \cos(wt) - V_\beta \sin(wt)$ }
DM(PANH)=MR1;             {Save negative active power $P_{\alpha n}^v$ (result is 32 bits)}
DM(PANL)=MR0;
MR=MX1*MY0 (SS);          {$Q_\beta^v = V_\beta \cos(wt)$
MR=MR-MX0*MY1 (SS);       {$Q_{\alpha p}^v = Q_\alpha^v + Q_\beta^v = -V_\alpha \sin(wt) + V_\beta \cos(wt)$ }
DM(QAPH)=MR1;             {Save positive reactive power $Q_{\alpha p}^v$ (result is 32 bits)}}
DM(QAPL)=MR0;
MR=MX1*MY0 (SS);          { $Q_\alpha^v = V_\beta \cos(wt)$ }
MR=MR+MX0*MY1 (SS);       {$Q_{\alpha n}^v = Q_\alpha^v + Q_\beta^v = V_\beta \cos(wt) + V_\alpha \sin(wt)$
DIS M_MODE;               {Disable Integer Mode}
AR=NOT MR1;          {Take 2' complements of result because actual result is -$Q_{\alpha n}^v$ }
AX0=AR;
AR=NOT MR0;
AY0=AR;
```

```
            AR=AY0+1;
            DM(QANL)=AR;                  {Save negative reactive power Q^v_an (result is 32 bits)}}
            AR=AX0+C;
            DM(QANH)=AR;

{===        Find Summation (Average) Values of Powers              ===}

            CALL AVER;                          {Call addition subroutine}
            AX0=DM(Pp_AVGH);                    {Save Positive Active power}
            DM(PAPV_AVGH)=AX0;                  {Most significant bits of 32 bits result}
            AX0=DM(Pp_AVGL);
            DM(PAPV_AVGL)=AX0;                  {Least significant bits of 32 bits result}

            AR=DM(Pn_AVGL);               {Save Negative Active power after multiply by 4}}
            SR=LSHIFT AR BY 2 (LO);
            AR=DM(Pn_AVGH);
            SR=SR OR ASHIFT AR BY 2 (HI);
            DM(PANV_AVGH)=SR1;                  {Most significant bits of 32 bits result}
            DM(PANV_AVGL)=SR0;                  {Least significant bits of 32 bits result}

            AX0=DM(Qp_AVGH);                    {Save Positive Reactive power}
            DM(QAPV_AVGH)=AX0;                  {Most significant bits of 32 bits result}
            AX0=DM(Qp_AVGL);
            DM(QAPV_AVGL)=AX0;                  {Least significant bits of 32 bits result}

            AR=DM(Qn_AVGL);               {Save Negative reactive power after multiply by 4}
            SR=LSHIFT AR BY 2 (LO);
            AR=DM(Qn_AVGH);
            SR=SR OR ASHIFT AR BY 2 (HI);
            DM(QANV_AVGH)=SR1;                  {Most significant bits of 32 bits result}
            DM(QANV_AVGL)=SR0;                  {Least significant bits of 32 bits result}


            AY0=DM(CHECKV);                     {Check one period is complete or not}
            AR=PASS AY0;
            IF EQ JUMP VVOLT;                   {If one period is complete finish voltage reading }
            AR=AY0-1;                           {If not, decrease CHECKV by 1 and continue}
            DM(CHECKV)=AR;
            RTI;

VVOLT:      AX0=SMPL_CURR;                      {Set the flag for current sampling}
            DM(MUX)=AX0;                        {-Switch mux to current then-}
            DM(SMPL_FLAG)=AX0;                  {-Set the flag-}
            CALL CLEAR_SUB;                     {Clear all data buffers for current}
            AX0=0X0001;
            DM(V_FLAG)=AX0;
            AX0=0;
            DM(I_FLAG)=AX0;                     {Begin for current}
            RTI;


{===        Read Sampled Currents and Find The Powers Due to Currents         ===}

READ_CRENT: AY0=0XFFF0;
            AX0=DM(ADC_W);                      {Read phase-B current from A/D}
            AR=AX0 AND AY0,                     {Remove last four bits}
            SR=ASHIFT AR BY -5 (HI);            {Divide result by 32 result 16 bit integer}
```

```
        DM(IB_LOAD)=SR1;                {Save result to memory}
        AX0=DM(ADCAUX);                 {Read phase-C current from A/D}
        AR=AX0 AND AY0;                 {Remove last four bits }
        SR=ASHIFT AR BY -5 (HI);        {Divide result by 32 result 16 bit integer}
        DM(IC_LOAD)=SR1;                {Save result to memory}

{===        Do 3/2 Transformation        ===}

        AX0=DM(IB_LOAD);                { Load current $I_b$ }
        AY0=DM(IC_LOAD);                { Load current $I_c$ }
        AR=AX0+AY0;                     { $I_b+I_c$ }
        IF AV CALL OV_ADD;
        AR=-AR;                         {$-(I_b+I_c)$}
        DM(IX_LOAD)=AR;                 { Save current $I_\alpha$ }
        AR=AX0-AY0;                     { $I_b-I_c$ }
        IF AV CALL OV_ADD;
        MY0=COEFF1;                     {COEFF1=$1/\sqrt{3}$ }
        MR=AR*MY0 (RND);                { $1/\sqrt{3}(I_b - I_c)$ }
        DM(IY_LOAD)=MR1;                {Save current $I_\beta$ }

{===        Find Powers        ===}

        MY0=DM(COSA);                   {Load Reference voltage }
        MY1=DM(SINA);
        MX0=DM(IX_LOAD);                {Load Supply Currents in two dimensions}
        MX1=DM(IY_LOAD);
        ENA M_MODE;                     {Enable Integer Mode}
        MR=MX0*MY0 (SS);                { $P_\alpha^i = I_\alpha \cos(wt)$ }
        MR=MR+MX1*MY1 (SS);             { $P_{\alpha p}^i = P_\alpha^i + P_\beta^i = I_\alpha \cos(wt) + I_\beta \sin(wt)$ }
        DM(PAPH)=MR1;                   {Save positive active power $P_{\alpha p}^i$ (result is 32 bits)}
        DM(PAPL)=MR0;
        MR=MX0*MY0(SS);                 { $P_\alpha^i = I_\alpha \cos(wt)$ }
        MR=MR-MX1*MY1(SS);              { $P_{\alpha n}^i = P_\alpha^i - P_\beta^i = I_\alpha \cos(wt) - I_\beta \sin(wt)$ }
        DM(PANH)=MR1;                   {Save negative active power $P_{\alpha n}^i$ }
        DM(PANL)=MR0;
        MR=MX1*MY0(SS);                 { $Q_\beta^i = V_\beta \cos(wt)$ }
        MR=MR-MX0*MY1(SS);              { $Q_{\alpha p}^i = Q_\alpha^i + Q_\beta^i = -V_\alpha \sin(wt) + V_\beta \cos(wt)$ }
        DM(QAPH)=MR1;                   { Save positive reactive power $Q_{\alpha p}^i$ (result is 32 bits)}
        DM(QAPL)=MR0;
        MR=MX1*MY0(SS);                 { $Q_\beta^i = V_\beta \cos(wt)$ }
        MR=MR+MX0*MY1(SS);              { $Q_{\alpha p}^i = Q_\alpha^i + Q_\beta^i = V_\alpha \sin(wt) + V_\beta \cos(wt)$ }
        DIS M_MODE;                     {Disable Integer Mode}
        AR=NOT MR1;               {Take 2' complements of result because actual result is -$Q_{\alpha n}^i$ }
        AX0=AR;
        AR=NOT MR0;
        AY0=AR;
        AR=AY0+1;
```

```
        DM(QANL)=AR;                    {Save negative reactive power Q^i_an }
        AR=AX0+C;
        DM(QANH)=AR;

    {===      Find Summation (Average) Of Powers          ===}

        CALL AVER;                      {Call summation subroutine}
        AR=DM(Pp_AVGL);                 {Save Positive Active Power after multiply by 4}
        SR=LSHIFT AR BY 2 (LO);
        AR=DM(Pp_AVGH);
        SR=SR OR ASHIFT AR BY 2 (HI);
        DM(PAP_AVGH)=SR1;               {Positive active power}
        DM(PAP_AVGL)=SR0;

        AR=DM(Pn_AVGL);                 {Save Negative Active Power after multiply by
16}
        SR=LSHIFT AR BY 4 (LO);
        AR=DM(Pn_AVGH);
        SR=SR OR ASHIFT AR BY 4 (HI);
        DM(PAN_AVGH)=SR1;               {Negative active power}
        DM(PAN_AVGL)=SR0;

        AR=DM(Qp_AVGL);             {Save Positive Reactive Power after multiply by 4}
        SR=LSHIFT AR BY 2 (LO);
        AR=DM(Qp_AVGH);
        SR=SR OR ASHIFT AR BY 2 (HI);
        DM(QAP_AVGH)=SR1;               {Positive reactive power}
        DM(QAP_AVGL)=SR0;

        AR=DM(Qn_AVGL);             {Save Negative Reactive power after multiply by 16}
        SR=LSHIFT AR BY 4 (LO);
        AR=DM(Qn_AVGH);
        SR=SR OR ASHIFT AR BY 4 (HI);
        DM(QAN_AVGH)=SR1;               {Negative reactive power}
        DM(QAN_AVGL)=SR0;

        AY0=DM(CHECKI);                 {Check 1 period is complete or not}
        AR=PASS AY0;
        IF EQ JUMP CCURR;               {if yes finish reading current}
        AR=AY0-1;                       {if not, decrement CHECHI by 1 and continue}
        DM(CHECKI)=AR;
        RTI;

CCURR:  AX0=SMPL_VOLT;                  {Set flag for voltage sampling}
        DM(MUX)=AX0;                    {Switch mux to voltage then-}
        DM(SMPL_FLAG)=AX0;              {-Set the flaf-}
        AX0=0X0001;                     {Current sampling completed}
        DM(I_FLAG)=AX0;
        RTI;

{==========          TIMER INTERRUPT          ===========}

TIME:   CALL UPDATE;                {Call UPDATE subroutine for update data that to be}
                                    { send throughout the serial port to PC           }
        I2=^SP0_TRAN;               { I2 is used for auto-transmitting only }
        L2=%SP0_TRAN;
        M2=1;
```

```
            AX0=AUTOBUFT;                      {Load constant value for enable auto-transmitting }
            DM(SPORT0_AUTOBUF_CTRL)=AX0;       { enable auto transmitting }
            MX0=DM(I2,M2);
            TX0=MX0;                           {Start transmitting }
            RTI;

{==========          SPORT0T TRANSMIT INTERRUPT          ===========}

{ For 128 samples: CHECKV, CHECKVI=7f
 For  64 samples: CHECKV, CHECKVI =3f
 For  72 samples: CHECKV, CHECKVI =47}

SPORT0T:    AX0=AUTOBUFTS;                     {Load constant value for disable auto-transmitting}
            DM(SPORT0_AUTOBUF_CTRL)=AX0;
            CALL CLEAR_SUB;                    {Clear all Buffer after Transmitting data }

{===        Reload Control Loops Data          ===}
            AX0=0;
            DM(ANGLE)=AX0;
            DM(V_FLAG)=AX0;
            AX0=0X007F;
            DM(CHECKV)=AX0;                    {For one period (7f) to find voltage information}
            DM(CHECKI)=AX0;                    {For one period (7f) to find current information}
            AX0=0X0001;
            AX0=SMPL_VOLT;
            DM(MUX)=AX0;
            DM(SMPL_FLAG)=AX0;
            RTI;

{======     This a subroutine for Update data               =====}
{======     Data to be sent through the serial port         =====}
{======     are updated in this subroutine                  =====}

UPDATE:     I2=^SP0_TRAN;                      { I2 is use for auto-transmitting only }
            L2=%SP0_TRAN;
            M2=1;
            AX0=0X8000;                        {unchanged}
            AY0=DM(PAPV_AVGH);      CALL ENCODE;        {Send P^v_αp }
            AY0=DM(QAPV_AVGH);      CALL ENCODE;
            AY0=DM(PANV_AVGH);      CALL ENCODE;
            AY0=DM(QANV_AVGH);      CALL ENCODE;
            AY0=DM(PAPI_AVGH);      CALL ENCODE;
            AY0=DM(QAPI_AVGH);      CALL ENCODE;
            AY0=DM(PANI_AVGH);      CALL ENCODE;
            AY0=DM(QANI_AVGH);      CALL ENCODE;        {Send Q^i_αn }

            AX0=0;
            DM(ANGLE)=AX0;                     {Set Initial angle to zero again}
            RTS;

{======     This a subroutine for Encoding data                                  =====}
{======     First coming data is converted to unsigned number, then reverse it,  =====}
{======     Finally add stop-bit, start bit; and put result into circular buffer =====}
{======     Input: AY0                                                           =====}

ENCODE:     AR=AX0 XOR AY0, SI=AY0;            { Convert to unsigned number}
            SR=LSHIFT AR BY -8 (HI);           { Shift data down, get first bit ready to compare }
```

```
            AR=0;                                       { Clear result }
            CNTR=8;                                     { do 8 bit }
            SE=-1;                                      { do it upwards }
            AY0=1;                                      { the increase }

            DO ENCODE1 UNTIL CE;
            AF=SR1 AND AY0;                             { test the bit }
            IF NE AR=AR OR AY0;                         { if bit = 1 then AR+1 }
            SR=LSHIFT SR1 (HI), AY1=AR;                 { shift down one bits }
ENCODE1:    AR=AR+AY1;                                  { result shift up on bit }
            SR=LSHIFT AR BY 1 (HI);
            AY1=3;                                      { stop bit }
            AR=SR1 OR AY1, SR1=SI;
            DM(I2,M2)=AR;                               { save to buffer }
{===        lower byte          ===}
            AR=0;                                       { clear result }
            CNTR=8;                                     { do 8 bit }
            SE=-1;                                      { do it upwards }
            AY0=1;                                      { the increase }
            DO ENCODE2 UNTIL CE;
            AF=SR1 AND AY0;                             { test the bit }
            IF NE AR=AR OR AY0;                         { if bit = 1 then AR+1 }
            SR=LSHIFT SR1 (HI), AY1=AR;                 { shift down one bits }
ENCODE2:    AR=AR+AY1;                                  { result shift up on bit }
            SR=LSHIFT AR BY 1 (HI);
            AY1=3;                                      { stop bit }
            AR=SR1 OR AY1;
            DM(I2,M2)=AR;                               { save to buffer }
            RTS;


{======     This a subroutine for Initialisation                        =====}
{======     All register that are used in the program are set           =====}
{======     all initial values for control loops are loaded and         =====}
{======     all buffers and memories are cleared in this subroutine     =====}

START:      AX0=0X101F;         DM(SYS_CTRL_REG)=AX0;
            AX0=DM_WAIT;        DM(DM_WAIT_REG)=AX0;

      {===        Set up Timer Registers          ===}

            AX0=TSCALE;         DM(TSCALE_REG)=AX0;
            AX0=TPERIOD;        DM(TCOUNT_REG)=AX0;
            AX0=TPERIOD;        DM(TPERIOD_REG)=AX0;

      {===        Set Up Sport1 Registers         ===}
{===        Serial port1 is not used          ===}

            AX0=0;          DM(SPORT1_AUTOBUF_CTRL)=AX0;
            AX0=0;          DM(SPORT1_RFSDIV)=AX0;
            AX0=0;          DM(SPORT1_SCLKDIV)=AX0;
            AX0=0;          DM(SPORT1_CTRL_REG)=AX0;

      {=== Set Up Sport0 Registers for data transmitting to PC        ===}

            AX0=AUTOBUFT;       DM(SPORT0_AUTOBUF_CTRL)=AX0;
            AX0=0;              DM(SPORT0_RFSDIV)=AX0;
            AX0=SCLKDIV;        DM(SPORT0_SCLKDIV)=AX0;
```

```
        AX0=SP0CTRL;        DM(SPORT0_CTRL_REG)=AX0;
        AX0=0;              DM(SPORT0_TX_WORDS0)=AX0;
        AX0=0;              DM(SPORT0_TX_WORDS1)=AX0;
        AX0=0;              DM(SPORT0_RX_WORDS0)=AX0;
        AX0=0;              DM(SPORT0_RX_WORDS1)=AX0;


    {===        ADMC Initializations            ===}

        AX0=ADMCMODE;       DM(ADMC_CTRL)=AX0;
        AX0=PWMSF;          DM(PWM_TM)=AX0;
        AX0=PWMDT;          DM(PWM_DT)=AX0;
        AX0=PWMPD;          DM(PWM_PD)=AX0;

{=== Initialize all used memories, clear all used buffer and load control loops parameters ====}

{For 128 samples: CHECKV, CHECKVI=7f
 For  64 samples: CHECKV, CHECKVI =3f
 For  72 samples: CHECKV, CHECKVI =47}

        AX0=0;
        CALL CLEAR_SUB;                         {Clear all data buffers}
        AX0=0;                                  {Initial values}
        DM(ANGLE)=AX0;
        DM(V_FLAG)=AX0;
        DM(V_FLAG)=AX0;
        AX0=0X007F;                             {For one period}
        DM(CHECKV)=AX0;
        DM(CHECKI)=AX0;                         {For one period}
        AX0=0X0001;
        AX0=SMPL_VOLT,                          {Switch MUX to sample voltages}
        DM(MUX)=AX0;
        DM(SMPL_FLAG)=AX0;
        RTS;

{====     This is a subroutine for sin calculation              ====}
{====     Calling parameters AX0=X in Scaled 1.15 format        ====}
{====     Return Values AR=sin(x) in 1.15 format                ====}
{====     sin(x)=3.140625x + 0.02026367x^2 – 5.32519x^3 + 0.5446778x^4 + 1.800293x^5     ====}
SIN_SUB:  I4=^SIN_COEFF;
          M4=1;
          L4=0;
          AY0=0X4000;
          AR=AX0;
          AF=AX0 AND AY0;
          IF NE AR=-AX0;
          AY0=0X7FFF;
          AR=AR AND AY0;
          MY1=AR;
          MF=AR*MY1(RND),MX1=DM(I4,M4);
          MR=MX1*MY1(SS),MX1=DM(I4,M4);
          CNTR=3;
          DO APPROX1 UNTIL CE;
          MR=MR+MX1*MF(SS);
APPROX1:  MF=AR*MF(RND),MX1=DM(I4,M4);
          MR=MR+MX1*MF(SS);
          SR=ASHIFT MR1 BY 3(HI);
          SR=SR OR LSHIFT MR0 BY 3(LO);
```

```
            AR=PASS SR1;
            IF LT AR=PASS AY0;
            AF=PASS AX0;
            IF LT AR=-AR;
            RTS;

{====       This is a subroutine processing overflow for integer multiplication     ====}
{====       Calling parameters: MR1, MR0=the value that should be adjusted         ====}
{====       Return Values MR0=ajusted value                                         ====}
{====       Register used: AF, MR1, and MR0.                                        ====}

OV_MULT:AF=PASS MR1;
            IF GT JUMP POS_OV;
            IF EQ JUMP POS_CHEK;
            MR1=0XFFFF;
            AF=AF-MR1;
            IF NE JUMP NEG_OV;
            AF=PASS MR0;
            IF LT JUMP FIN_CHEK;
NEG_OV:     MR0=0X8001;
            JUMP FIN_CHEK;
POS_CHEK: AF=PASS MR0;
            IF GE JUMP FIN_CHEK;
POS_OV:     MR0=0X7FFF;
FIN_CHEK: RTS;

{====       This is a subroutine processing overflow for ADD or SUB      ====}
{====       Calling parameters: AR=the value to be adjusted.             ====}
{====       Return Values : AR=adjusted value                            ====}


OV_ADD:     AR=PASS AR;
            IF LT JUMP POS_ADD;
            AR=0X8001;
            JUMP FIN_ADD;
POS_ADD:    AR=0X7FFF;
FIN_ADD:    RTS;

{======     This is a subroutine for clearing all data buffers      ======}

CLEAR_SUB:  AX0=0,
            M0=1,
            L0=0;
            I0=^Pn_BUF;
            CNTR=%Pn_BUF;
            DO CLEAR1 UNTIL CE;
CLEAR1:     DM(I0,M0)=AX0;
            I0=^Qn_BUF;
            CNTR=%Qn_BUF;
            DO CLEAR2 UNTIL CE;
CLEAR2:     DM(I0,M0)=AX0;
            I0=^Pp_BUF;
            CNTR=%Pp_BUF;
            DO CLEAR3 UNTIL CE;
CLEAR3:     DM(I0,M0)=AX0;
            I0=^Qp_BUF;
            CNTR=%Qp_BUF;
            DO CLEAR4 UNTIL CE;
```

```
CLEAR4: DM(I0,M0)=AX0;
        I0=^Pn_BUF;                     {Data Buffter Address..}
        L0=%Pn_BUF;                     {Pointer Initializations for used buffers}
        M0=1;
        I1=^Qn_BUF;
        L1=%Qn_BUF;
        M1=1;
        I6=^Pp_BUF;
        L6=%Pp_BUF;
        M6=1;
        I7=^Qp_BUF;
        L7=%Qp_BUF;
        M7=1;
        AX0=0;
        DM(Pp_AVGH)=AX0;                {Clear Pp_AVG memory}
        DM(Pp_AVGL)=AX0;
        DM(Pn_AVGH)=AX0;                {Clear Pn_AVG memory}
        DM(Pn_AVGL)=AX0;
        DM(Qp_AVGH)=AX0;                {Clear Qp_AVG memory}
        DM(Qp_AVGL)=AX0;
        DM(Qn_AVGH)=AX0;                {Clear Qn_AVG memory}
        DM(Qn_AVGL)=AX0;
        RTS;

{====   This is a subroutine for finding summation (average) of powers    ====}

AVER:   AX1=DM(Pp_AVGH);
        AX0=DM(Pp_AVGL);
        AY1=DM(PAPH);
        AY0=DM(PAPL);
        AR=AX0+AY0;
        DM(Pp_AVGL)=AR;
        AR=AX1+AY1+C;
        DM(Pp_AVGH)=AR;

        AX1=DM(Qp_AVGH);
        AX0=DM(Qp_AVGL);
        AY1=DM(QAPH);
        AY0=DM(QAPL);
        AR=AX0+AY0;
        DM(Qp_AVGL)=AR;
        AR=AX1+AY1+C;
        DM(Qp_AVGH)=AR;

        AX1=DM(Pn_AVGH);
        AX0=DM(Pn_AVGL);
        AY1=DM(PANH);
        AY0=DM(PANL);
        AR=AX0+AY0;
        DM(Pn_AVGL)=AR;
        AR=AX1+AY1+C;
        DM(Pn_AVGH)=AR;

        AX1=DM(Qn_AVGH);
        AX0=DM(Qn_AVGL);
        AY1=DM(QANH);
        AY0=DM(QANL);
        AR=AX0+AY0;
```

```
        DM(Qn_AVGL)=AR;
        AR=AX1+AY1+C;
        DM(Qn_AVGH)=AR;
        RTS;

.ENDMOD;                        {End of program}

{==================          NVARIAB.H          ======================}
{===    This file contain all data buffers and data memories  that are used by program    =====}

.VAR/DM/RAM/STATIC/CIRC     SP0_TRAN[22];
.VAR/DM/RAM/STATIC          SIN_COEFF[5];
.VAR/DM/RAM/STATIC          ANGLE;
.VAR/DM/RAM/STATIC          ANGLE1;
.VAR/DM/RAM/STATIC          ANGLE2;
.VAR/DM/RAM/STATIC          RANDOM;
.VAR/DM/RAM/STATIC          NANGLE;
.VAR/DM/RAM/STATIC          INANGLE;
.VAR/DM/RAM/STATIC          SINA;
.VAR/DM/RAM/STATIC          COSA;
.VAR/DM/RAM/STATIC          IA_LOAD;
.VAR/DM/RAM/STATIC          IB_LOAD;
.VAR/DM/RAM/STATIC          IC_LOAD;
.VAR/DM/RAM/STATIC          IX_LOAD;
.VAR/DM/RAM/STATIC          IY_LOAD;
.VAR/DM/RAM/STATIC          PANH;
.VAR/DM/RAM/STATIC          PANL;
.VAR/DM/RAM/STATIC          PAPH;
.VAR/DM/RAM/STATIC          PAPL;
.VAR/DM/RAM/STATIC          QANH;
.VAR/DM/RAM/STATIC          QANL;
.VAR/DM/RAM/STATIC          QAPH;
.VAR/DM/RAM/STATIC          QAPL;
.VAR/DM/RAM/STATIC          PAP;
.VAR/DM/RAM/STATIC          PAN;
.VAR/DM/RAM/STATIC          QAP;
.VAR/DM/RAM/STATIC          QAN;
.VAR/DM/RAM/STATIC          Pn_AVGH;
.VAR/DM/RAM/STATIC          Pn_AVGL;
.VAR/DM/RAM/STATIC          Pp_AVGH;
.VAR/DM/RAM/STATIC          Pp_AVGL;
.VAR/DM/RAM/STATIC          Qn_AVGH;
.VAR/DM/RAM/STATIC          Qn_AVGL;
.VAR/DM/RAM/STATIC          Qp_AVGH;
.VAR/DM/RAM/STATIC          Qp_AVGL;
.VAR/DM/RAM/STATIC          PANV_AVGH;
.VAR/DM/RAM/STATIC          PAPV_AVGH;
.VAR/DM/RAM/STATIC          QANV_AVGH;
.VAR/DM/RAM/STATIC          QAPV_AVGH;
.VAR/DM/RAM/STATIC          PAN_AVGH;
.VAR/DM/RAM/STATIC          PAP_AVGH;
.VAR/DM/RAM/STATIC          QAN_AVGH;
.VAR/DM/RAM/STATIC          QAP_AVGH;
.VAR/DM/RAM/STATIC          PANV_AVGL;
.VAR/DM/RAM/STATIC          PAPV_AVGL;
.VAR/DM/RAM/STATIC          QANV_AVGL;
.VAR/DM/RAM/STATIC          QAPV_AVGL;
.VAR/DM/RAM/STATIC          PAN_AVGL;
```

```
.VAR/DM/RAM/STATIC              PAP_AVGL;
.VAR/DM/RAM/STATIC              QAN_AVGL;
.VAR/DM/RAM/STATIC              QAP_AVGL;
.VAR/DM/RAM/STATIC              U_LOAD;
.VAR/DM/RAM/STATIC              V_LOAD;
.VAR/DM/RAM/STATIC              W_LOAD;
.VAR/DM/RAM/STATIC              VX_LOAD;
.VAR/DM/RAM/STATIC              VY_LOAD;
.VAR/DM/RAM/STATIC              CONT;
.VAR/DM/RAM/STATIC              CONT1;
.VAR/DM/RAM/STATIC              SMPL_FLAG;
.VAR/DM/RAM/STATIC              CHECKV;
.VAR/DM/RAM/STATIC              CHECKI;
.VAR/DM/RAM/STATIC              V_FLAG;
.VAR/DM/RAM/STATIC              I_FLAG;
.INIT           SIN_COEFF:0X3240,0X0053,0XAACC,0X08B7,0X1CCE;


{===================            CONST.H            ==========================}
{=== This file contain all constant that are used by the program and some registers addresses    ====}

{====       Digital to Analogue Converter (DAC) Registers Addresses        ====}
.CONST  DAC1=0X1600;
.CONST  DAC2=0X1601;
.CONST  DAC3=0X1602;
.CONST  DAC4=0X1603;
.CONST  DAC_LD=0X2600;                  {DAC Data Transfer}

.CONST  MUX=0X3020;                     {The Address For Mux}
.CONST  ST_AD=0X3030;                   {The Address For Starting ADC}
.CONST  SMPL_CURR=0X0000;               {Flag for current sampling}
.CONST  SMPL_VOLT=0X0001;               {Flag for voltage sampling}

{====       System Control Constants            ====}
.CONST  SYSCLOCK=20;                    {MHz. Clock frequency}
.CONST  BAUDRATE=19200;
.CONST  SP0CTRL=0X4E0A;         {Internal Serial Clok=ISCKL,ITFS,TFSW,TFSR AND SLEN=10}
.CONST  AUTOBUFT=0X0502;                {TBUF Enable, I2 and M2 use for autobufer Transmit}
.CONST  AUTOBUFTS=0X0500;               {TBUF Disable, I2 and M2 use for autobufer Transmit}
.CONST  SCLKDIV=SYSCLOCK*500000/BAUDRATE-1;

{====       Set timer to send interrupts for serial communication          ====}
.CONST  TPERIOD=19999;                  {FOR 20MHz 1ms*TSCALE}
.CONST  TSCALE=124;                     {Every 125ms send data from Serial port1 for different }
{time TSCALE have to change}
.CONST  SYSCTRL=0X101F;                 {SPORT0 Enabled And SPORT1 Disabled}
.CONST  DM_WAIT=0X0280;                 {DWAIT3=1 DWAIT2=2}
.CONST  INT_CTRL=0X0007;                {Disable Nesting, Edge Sensitivity for IRQ1 and IRQ2}
.CONST  INT_MASK=0X0035;                {IRQ1,IRQ2,SPORT0 Transmit Enable}
.CONST  INT_CLR=0X0000;                 {Clear All Interrupt Bits}

{====            ADMC Control Constants            ====}

.CONST  PWMFREQ=6400;                   {Hz   PWM   Parameters   is   for   sampling
frequency=N/20ms}
.CONST  PWMCLOCK=10;                    {MHz After Divided by Two}
.CONST  PWMSF=16*PWMCLOCK*1000000/PWMFREQ;        {sampling interval}
.CONST  ADMCMODE=0X0B80;                {ADC_INT_EDGE,UVW CONVERSION}
```

```
.CONST  V_INT_STAT=0X0020;
.CONST  COEFF0=0X2AAA;              { 1/3 }
.CONST  COEFF1=0X49E6;              { 1/√3 }
.CONST  COEFF2=0X5555;              ( 2/3 }
.CONST  base=0X0D49, sqrt2=0X5A82;

{===================          REG.H          ==========================}
{========   This file contain all ADSP2101 Memory Mapped Register address   ========}

.CONST      SYS_CTRL_REG=0X3FFF;
.CONST      DM_WAIT_REG=0X3FFE;
.CONST      TPERIOD_REG=0X3FFD;
.CONST      TCOUNT_REG=0X3FFC;
.CONST      TSCALE_REG=0X3FFB;
.CONST      SPORT0_CTRL_REG=0X3FF6;
.CONST      SPORT0_SCLKDIV=0X3FF5;
.CONST      SPORT0_AUTOBUF_CTRL=0X3FF3;
.CONST      SPORT0_RX_WORDS1=0X3FFA;
.CONST      SPORT0_RX_WORDS0=0X3FF9;
.CONST      SPORT0_TX_WORDS1=0X3FF8;
.CONST      SPORT0_TX_WORDS0=0X3FF7;
.CONST      SPORT0_RFSDIV=0X3FF4;
.CONST      SPORT1_CTRL_REG=0X3FF2;
.CONST      SPORT1_SCLKDIV=0X3FF1;
.CONST      SPORT1_RFSDIV=0X3FF0;
.CONST      SPORT1_AUTOBUF_CTRL=0X3FEF;

{===================          PORT.H          ==========================}

.PORT     ADMC_CTRL;
.PORT     ADC_U;
.PORT     ADC_V;
.PORT     ADC_W;
.PORT     ADCAUX;
.PORT     ADMC_STAT;
```

E.2 Visual Basic Program

```
'DEFINE GLOBAL VARIABLE USED IN PROGRAM

Global receive$                                       'Variable for receiving data from DSP
Global kc, kc2, char, charXn, charlv, charr, charf    'Control variables
Global u_input(), p1(), p2(), km(), y_output
Global g_par1(), g_par2(), est, a_par1(), a_par2(), xp1(), ip1()

'MAIN PROGRAM

Sub main ()

Static y(8)
Dim Xpos, Ypos
```

```
'Set parameter for graphic display

ScaleMode = 4                           'Set Scalemode to pixels
DrawWidth = 8                           'Set DrawWidth, width of the line in pixels
DrawStyle = 0                           'Set DrawStyle to dot

kc2 = 1
pp = 0
receive$ = "                    "       'initially data are blank
pdtform.Show                            'Show the pdtform for graphic and numeric display


fnam1 = "d:\muslum\mdata\negaz3.dat"    'Define file names for saving data
Open fnam1 For Output As 1              'Open file


'INITIAL VALUES

Vnx0 = 0: Vny0 = 0: Inx0 = 0: Iny0 = 0: In0 = 0: kf = 0
AV = .8                                 'average factor
sqrt2 = Sqr(2)
charXn = ""
charlv = ""
charl = ""

'HEALTHY MOTOR PARAMETERS

'Generalized Machine
Rn1 = 3.17
Xn1 = 7.61


'INITIALISATION FOR RLS ALGORITHM
alpha = 400
'INITIAL VALUES FOR Xhn FIT
nx = 6
ReDim xp1(nx, nx), g_par1(nx, 1), g_par2(nx, 1)

For i = 0 To nx - 1
For j = 0 To nx - 1
If i = j Then
xp1(i, j) = alpha
Else
xp1(i, j) = 0
End If
Next j
Next i

'INITIAL PARAMETERS
intx = "1"                'intx is 1 or 0 for known values and unknown values parameters respectively
If intx = "1" Then
        g_parl(0,0) =0.14176:     g_parl(1,0) = -0.001122:  g_parl(2,0) = -0.002498:
g_parl(3,0) = -0.00127:  g_parl(4,0) = 0.0024065:  g_parl(5,0) = 0.00024281
elseIf intx = "0" Then
For i = 0 To nx - 1
g_par1(i, 0) = 0
Next i
End If
```

```
‘INITIAL VALUES FOR Imnlv FIT
ni = 5
ReDim ip1(ni, ni), a_par1(ni, 1), a_par2(ni, 1)

For i = 0 To ni - 1
For j = 0 To ni - 1
If i = j Then
ip1(i, j) = alpha
Else
ip1(i, j) = 0
End If
Next j
Next i

'INITIAL PARAMETERS
inti = ”1”                    ‘inti is 1 or 0 for known values and unknown values parameters respectively
if inti = ”1” Then
        a_par1(0,0)= 0.2372:  a_par1(1,0)= 0.0132:  a_par1(2,0)= -0.0028
a_par1(3,0)= 0.1773:  a_par1(4,0)= .0329
elseIf inti = “0” Then
For i = 0 To ni - 1
a_par1(i, 0) = 0
Next i
End If

'SHIFTING FACTOR OF VOLTAGE AND CURRENT
NVp = 0 - 16            'for shift voltage 6 bits right (16 bits right was done in DSP)
NVn = 2 - 16            'for shift negative voltage 8 bits right (14 bits right was done in DSP)
NIp = 2 - 16            'for shift current 8 bits right (14 bits right was done in DSP)
NIn = 4 - 16            'for shift negative current 10 bits right (12 bits right was done in DSP)

'AMPLIDUTE SCALLING FACTOR FOR VOLTAGE,CURRENT AND IMPEDANCE
Ns = 128   'number of samples in per period
KNs = 1 / (Ns * sqrt2)
KVp = KNs * 403 / 2 ^ (22 + NVp)    'multiply by scalling factor and shift factor
KVn = KNs * 403 / 2 ^ (22 + NVn)
KIp = KNs *  16.97 / 2 ^ (21 + NIp)
KIn = KNs * 16.97 / 2 ^ (21 + NIn)



'PROGRAM BEGIN FROM HERE


Do While DoEvents()             ‘Go into loop until interrupt

        If kc1 <> kc Then               'Check for new data

'GET DATA FROM DSP

                For j = 1 To 8
                        i = j * 2
                        b$ = Mid(receive$, i - 1, i - 1): c$ = Mid(receive$, i, i)
                        y(j - 1) = (Asc(b$) - 128) * 256 + Asc(c$)
                Next
```

```
'DIFINE NEGATIVE AND POSITIVE SEQUENCE VOLTAGE AND CURRENT FROM RECIEVING DATA

        Vpx =KVp * y(0)
        Vpy = KVp *  y(1)
        Vnx = KVn *  y(2) - Vnx0
        Vny = KVn *  y(3) - Vny0
        Ipx = KIp *  y(4)
        Ipy = KIp *  y(5)
        Inx = KIn *  y(6) - Inx0
        Iny = KIn *  y(7) - Iny0

'FIND AVERAGE VALUES FOR VOLTAGE AND CURRENT
        Vnx_av = Vnx_av + (Vnx - Vnx_av) * AV
        Vny_av = Vny_av + (Vny - Vny_av) * AV
        Inx_av = Inx_av + (Inx - Inx_av) * AV
        Iny_av = Iny_av + (Iny - Iny_av) * AV
        Vpx_av = Vpx_av + (Vpx - Vpx_av) * AV
        Vpy_av = Vpy_av + (Vpy - Vpy_av) * AV
        Ipx_av = Ipx_av + (Ipx - Ipx_av) * AV
        Ipy_av = Ipy_av + (Ipy - Ipy_av) * AV

'FIND POSITIVE AND NEGATIVE; VOLTAGES AND CURRENT VALUES FROM AVERAGING VALUES
        Vp_av = Sqr(Vpx_av ^ 2 + Vpy_av ^ 2)
        Vn_av = Sqr(Vnx_av ^ 2 + Vny_av ^ 2)
        Ip_av = Sqr(Ipx_av ^ 2 + Ipy_av ^ 2)
        In_av = Sqr(Inx_av ^ 2 + Iny_av ^ 2)


'FIND NEGATIVE SEQUENCE IMPEDANCE
        Zn1 = Vnx_av * Inx_av + Vny_av * Iny_av
        Zn2 = Vny_av * Inx_av - Vnx_av * Iny_av
        Zn4 = Inx_av ^ 2 + Iny_av ^ 2

        If Z4 <> 0 Then Rn = Zn1 / Zn4: Xn = Zn2 / Zn4

'FIND THE NEGATIVE SEQUENCE IMPEDANCE ANGLE
        angle Rn, Xn, Zn_ang

'FIND THE NEGATIVE SEQUENCE VOLTAGE ANGLE
        angle Vnx_av, Vny_av, Vn_ang =

'ADJUST AV FOR AVEARGING BY PRESSING KEYS + OR -
        If char = "+" Then AV = AV + .01: char = ""
        If char = "-" Then AV = AV - .01: char = ""

'Call RLS sub program for calculate curve fitting parameters
        if charXn = "1" Then
            rlsx Ipx_av, Ipy_av, Vn_av, Vn_ang, Xn, xp1(), g_par1(), nx, g_par2()
        End if

        If  charlv = "1" Then
            rlsi Ipx_av, Ipy_av, Imn_av, ip1(), a_par1(), ni, a_par2()
        End If

'FIND ESTIMATED Xhn
        uxn(0,0) = 1:  uxn(0,1) = Vn_an:  uxn(0,2) = Sin(2*Vn_ang)
```

```
            uxn(0,3) = Cos(2*Vn_ang):  uxn(0,4) = Ipx_av:  uxn(0,4) = Ipy_av^2
            multmat uxn(), g_par1(), 1, nx, 1, matxn()

            Xhn = 1 / matxn(0,0)

'FIND NEGATIVE SQUENCE CURRENT THAT COME FROM VOLTAGE and LOAD VARIATION
            If charl = "1" Then
                  ui(0,0) = 1:  ui(0,1) = Ipx_av:  ui(0,2) = Ipx_av^2:  ui(0,3) = Ipy_av:  ui(0,4) = Ipy_av^2
                  multmat ui(), a_par1(), 1, ni, 1, matui()
                  Imnlv=matui(0,0)
            Else
                  Imnlv = 0: Inmznlv = 0
            End If


'*******************************************************************************
'********************* MOTOR FAULT CURRENT BY USING Zhn ***********************
'*******************************************************************************

'FIND SUPPLY NEGATIVE SEQUENCE CURRENT

            Zhn = Rn1 ^ 2 + Xn1 ^ 2
            Isnx = (Vnx_av * Rn1 + Vny_av * Xn1) / Zhn
            Isny = (Vny_av * Rn1 - Vnx_av * Xn1) / Zhn
            Isn = Sqr(Isnx ^ 2 + Isny ^ 2)



'FIND MOTOR NEGATIVE SEQUENCE CURRENT
            Imnx = Inx_av - Isnx
            Imny = Iny_av - Isny
            Imnx_av = Imnx_av + (Imnx - Imnx_av) * AV
            Imny_av = Imny_av + (Imnx - Imny_av) * AV
            Imnzn_av = Sqr(Imnx_av ^ 2 + Imny_av ^ 2)


'*******************************************************************************
'********************* MOTOR FAULT CURRENT BY USING Xhn ***********************
'*******************************************************************************


'FIND SUPPLY NEGATIVE SEQUENCE CURRENT FROM Xhn
            Iss = Vn_av * Sin(Zn_ang) / Xhn
            Imn = In_av – Iss – Imnlv
            Imn_av = Imn_av + (Imn - Imn_av) * AV

'CHECK THAT FOR DIVISION BY ZERO
'negative to positive current ratio
            If Ip_av <> 0 Then Inp_av = 100 * In_av / Ip_av
'negative to positive voltage ratio
            If Vp_av <> 0 Then Vnp_av = 100 * Vn_av / Vp_av


'find residual unbalance in healthy motor
            If charr = "1" Then
                  Vnx0 = Vnx_av:    Vny0 = Vny_av:
                  Inx0 = Inx_av:              Iny0 = Iny_av
                  charr = ""
```

```
                End If
'make residual unbalance values zero
                If charr = "0" Then
                        Vnx0 = 0:    Vny0 = 0:
                        Inx0 = 0:              Iny0 = 0
                        charr = ""
                End If

'DRAW REFERENCE LINES ON SCREEN FOR GRAPHIC DISPLAY
                If kc2 = 1 Then
                        pdtform.Line (20, 25)-(200, 25), QBColor(7)
                        pdtform.Line (20, 35 + 2)-(200, 35 + 2), QBColor(7)
                        pdtform.Line (20, 45 + 4)-(200, 45 + 4), QBColor(7)
                        pdtform.Line (20, 55 + 6)-(200, 55 + 6), QBColor(7)
                        pdtform.Line (20, 65 + 8)-(200, 65 + 8), QBColor(7)
                        pdtform.Line (20, 75 + 10)-(200, 75 + 10), QBColor(7)
                        pdtform.Line (20, 85 + 12)-(200, 85 + 12), QBColor(7)
                        pdtform.Line (20, 95 + 14)-(200, 95 + 14), QBColor(7)
                        pdtform.Line (20, 105 + 16)-(200, 105 + 16), QBColor(7)
                End If

'Display calculated information in graphic mode
                Xpos = .2 * kc2 + 43                                        'X axis value

                Ypos = -Vp_av / 50 + 25 'ScaleHight / 2                     'Y axis value for Vp_av
                pdtform.PSet (Xpos, Ypos), QBColor(7)     'Black'Put point on screen for a given cordinate

                Ypos = -4 * Ip_av + 37'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(11)    'light magenta

                Ypos = -2 * Vn_av + 49 'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(13)    'light blue

                Ypos = -10 * In_av + 61 'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(10)    'light green

                Ypos = -10 * Iss + 73 'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(14)    'light yellow

                Ypos = -100 * Imn_av + 85 'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(11)    'light cyan

                Ypos = -5 * Rn + 97'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(9)     'light blue

                Ypos = -5 * Xn + 109'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(10)    'light green

                Ypos = -100 * Imnzn_av + 121 'ScaleHight / 2
                pdtform.PSet (Xpos, Ypos), QBColor(12)    'light red


'Display calculated values in numeric format
                pdtform.x(0) = Format(Vp_av, "0.000")          'Vp positive voltage
                pdtform.x(1) = Format(Vn_av, "0.000")          'Vn negative voltage
                pdtform.x(2) = Format(Ip_av, "0.000")                  'Ip positive current
                pdtform.x(3) = Format(In_av, "0.000")                  'In total negative current
                pdtform.x(4) = Format(Iss, "0.000")            'Isn total negative supply current
```

```
            pdtform.x(5) = Format(Imn_av, "0.000")          'Imn motor fault current by using Xhn
            pdtform.x(6) = Format(Imnzn_av, "0.000")        'Imnzn motor fault current by using Zhn
            pdtform.x(7) = Format(Imnlv, "0.000") 'Negative sequence current for load and voltage variation
            pdtform.x(8) = Format( Rn, "0.000")             ' real part of negative impedance
            pdtform.x(9) = Format( Xn, "0.000")             'imaginary part of negative impedance
            pdtform.x(10) = Format(Xhn, "0.00")             'Estimated  negative sequence reactance
            pdtform.x(11) = Format(Zn , "0.000")            'Zn negative impedance
            pdtform.x(12) = Format(Vnp_av, "0.00")     'Vnp negative to positive voltage ratio in part 100
            pdtform.x(13) = Format(Inp_av, "0.00")          'Inp negative to positive current ratio in part 100
            pdtform.x(14) = Format(Zn_ang, "0.000")         'Angle of negative sequence impedance
            pdtform.x(15) = Format(Ipy_av, "0.000")         'Magnetization current
            pdtform.x(16) = Format(Ipx_av, "0.000")         'Load current
            pdtform.x(17) = Format(kf, "0.00")              'number of data that store to file


'save raw data to a file
            If char = "f" Then
                  Print #1, Vpx_av, Vpy_av, Vnx_av, Vny_av, Ipx_av, Ipy_av, Inx_av, Iny_av
                  kf = kf + 1 'number of data
            End If

            kc1 = kc
            kc2 = kc2 + 1
            If kc2 = 785 Then
                  kc2 = 1
                  pdtform.Cls
            End If
      End If
Loop                                        'Program finish here

End Sub



'*********************************************************************************
'******************** SUBROUTINE USED IN THE PROGRAM *************************
'*********************************************************************************

'RECEIVING DATA FROM DSP VIA SERIAL PORT
Sub Com_OnComm ()

If com.CommEvent <> 2 Then GoTo ok
      k = com.InBufferCount: receiving = k: a$ = com.Input
      If k = 16 Then receive$ = a$
      c = c + 1: Counter = c
      kc = kc + 1
ok:

End Sub

'FIND ANGLE IN RANGE 0 TO 360 DEGREES
Sub angle (real, imag, angd)

PI = 3.141592654
DEG = 180 / PI

      If real > 0 And imag > 0 Then angr = Atn(imag / real)
```

```
        If real > 0 And imag < 0 Then angr = Atn(imag / real) + 2 * PI
        If real < 0 Then angr = Atn(imag / real) + PI
        If real = 0 And imag > 0 Then angr = PI / 2
        If real = 0 And imag < 0 Then angr = 3 * PI / 2
        angd = angr * DEG                                   'convert radian to degree

End Sub

'This subprogram find summation or subtraction of matrixes
Sub mat_ad (am(), bm(), na, ma, sbad, cm())

        For i = 0 To na - 1
                For j = 0 To ma - 1
                        cm(i, j) = am(i, j) + sbad * bm(i, j)
                Next j
        Next i

End Sub


'This subprogram find the multiplication of two matrix
'aa=[ns ms] and bb=[ms ls]

Sub multmat (aa(), bb(), ns, ms, ls, cc())

        For i = 0 To ns - 1
                For j = 0 To ls - 1
                        For k = 0 To ms - 1
                                multab = aa(i, k) * bb(k, j)
                                cc(i, j) = cc(i, j) + multab
                        Next k
                Next j
        Next i

End Sub

'This subprogram find the parameters on-line for define curve
Sub rlsi (Ipxas, Ipyas, y_output, p1si(), a_par1s(), nn, a_par2s())
ReDim p1ui(nn, 1), uip1ui(1, 1)
ReDim u_input(nn, 1), u_inputtr(1, nn), p1ui(nn, 1), uip1ui(1, 1)
ReDim uitp1(1, nn), k2m(nn, 1), puup(nn, nn), p2(nn, nn)
ReDim est(1, 1), b22(nn, 1), km2(nn, 1)

'Step 2
'define input
        u_input(0, 0) = 1
        For i = 1 To nn - 3
                u_input(i, 0) = Ipxas ^ i
                u_input(i + 2, 0) = Ipyas ^ i
        Next i

'Step 3
'find transpose of input matrix
        For i = 0 To nn - 1
                u_inputtr(0, i) = u_input(i, 0)
        Next i

'find p1*u_input
```

```
        multmat p1s(), u_input(), nn, nn, 1, p1u()

'find u_inputtr*p1u
        multmat u_inputtr(), p1u(), 1, nn, 1, up1u()
‘find 1/a
        a = 1 / (1 + up1u(0, 0))

‘STEP 4
'find u_inputtr*p1
        multmat u_inputtr(), p1s(), 1, nn, nn, utrp1()

‘STEP 5
'find k=a*p1u
        For i = 0 To nn - 1
                k(i, 0) = a * p1u(i, 0)
        Next i

'find p1*u * utr*p1/a
        multmat k(), utrp1(), nn, 1, nn, p1uutrp1()

'find p2=p1-p1uutrp1
        sb_ad = -1  'for subtraction
        mat_ad p1s(), puup(), nn, nn, sb_ad, p2()

‘STEP 6
'find u_inputtr*b_par1
        multmat u_inputtr(), b_par1s(), 1, nn, 1, est()
        est2 = y_output - est(0, 0)

‘STEP 7
'find k * est2
        For i = 0 To nn - 1
                b22(i, 0) = est2 * k(i, 0)
        Next i

'find b1-b22
        sb_ad = 1  'for summation
        mat_ad b_par1s(), b22(), nn, 1, sb_ad, b_par2s()

‘STEP 8
'pass new value to old value for next computation
        For i = 0 To nn - 1
                b_par1s(i, 0) = b_par2s(i, 0)
        Next i
        For i = 0 To nn - 1
                For j = 0 To nn - 1
                        p1si(i, j) = p2(i, j)
                Next j
        Next i

End Sub



'This subprogram find the parameters on-line for define curve
Sub rls (Ipxas, Ipyas, Vn_av, Vn_ang, y_output, p1sx(), g_par1s(), nn, g_par2s())
ReDim p1ui(nn, 1), uip1ui(1, 1)
ReDim u_input(nn, 1), u_inputtr(1, nn), p1ui(nn, 1), uip1ui(1, 1)
```

```
ReDim uitp1(1, nn), k2m(nn, 1), puup(nn, nn), p2(nn, nn)
ReDim est(1, 1), b22(nn, 1), km2(nn, 1)

'Step 2
'define input
        u_input(0, 0) = 1
        u_input(1, 0) = Vn_av
        u_input(2, 0) = Sin( Vn_ang)
        u_input(3, 0) = Cos( Vn_ang)
        u_input(4, 0) = Ipxas
        u_input(5, 0) = Ipyas ^ 2

'Step 3
'find transpose of input matrix
        For i = 0 To nn - 1
                u_inputtr(0, i) = u_input(i, 0)
        Next i

'find p1*u_input
        multmat p1s(), u_input(), nn, nn, 1, p1u()

'find u_inputtr*p1u
        multmat u_inputtr(), p1u(), 1, nn, 1, up1u()
'find 1/a
        a = 1 / (1 + up1u(0, 0))

'STEP 4
'find u_inputtr*p1
        multmat u_inputtr(), p1s(), 1, nn, nn, utrp1()

'STEP 5
'find k=a*p1u
        For i = 0 To nn - 1
                k(i, 0) = a * p1u(i, 0)
        Next i

'find p1*u * utr*p1/a
        multmat k(), utrp1(), nn, 1, nn, p1uutrp1()

'find p2=p1-p1uutrp1
        sb_ad = -1  'for subtraction
        mat_ad p1s(), puup(), nn, nn, sb_ad, p2()

'STEP 6
'find u_inputtr*b_par1
        multmat u_inputtr(), b_par1s(), 1, nn, 1, est()
        est2 = y_output - est(0, 0)

'STEP 7
'find k * est2
        For i = 0 To nn - 1
                b22(i, 0) = est2 * k(i, 0)
        Next i

'find b1-b22
        sb_ad = 1  'for summation
        mat_ad b_par1s(), b22(), nn, 1, sb_ad, b_par2s()
```

```
'STEP 8
'pass new value to old value for next computation
        For i = 0 To nn - 1
                b_par1s(i, 0) = b_par2s(i, 0)
        Next i
        For i = 0 To nn - 1
                For j = 0 To nn - 1
                        p1s(i, j) = p2(i, j)
                Next j
        Next i

End Sub


Sub Form_KeyPress (keyascii As Integer)
char = Chr(keyascii)

        If char = "c" Then          'clear graphic display
                Cls
                kc2 = 1
        End If

        If char = "f" Then          'save data to file
                charf = char
        ElseIf char = "s" Then
                charf=""
        End If

        If char = "r" Then          'remove residual unbalance
                charr = "1"
        ElseIf char = "t" Then
                charr = "0"
        End If

        If char = "x" Then          'Xn parameter fit
                charXn = "1"
        ElseIf char = "z" Then
                charXn = ""
        End If

        If char = "l" Then          'load and voltage variation parameter fit
                charlv = "1"
        ElseIf char = "k" Then
                charlv = ""
        End If

        If char = "e" Then          'exit form program
                Unload pdtform
                chare = "e"
                chexit = "e"
        Else
                chare = ""
        End If

End Sub
```