Synchronized Rotor Angle Measurement of Synchronous Machines

David Christopher Mazur

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
In
Electrical Engineering

Jaime De La Ree
Virgilio Centeno
William T. Baumann

April 13, 2012
Blacksburg, Virginia

Keywords: Torque Angle, Synchronous Machine, Precision Time Protocol,
Programmable Automation Controller, Synchronized Actuation, Time Synchronization

Copyright 2012

Synchronized Rotor Angle Measurement of Synchronous Machines

David Christopher Mazur

ABSTRACT

A key input parameter to governor feedback control and stability protection of generators is the angle of the induced voltage internal to the generator. Current practice is to estimate this value using measurements from the terminals of the generator and mathematical models. This project aims to develop a system that would directly measure the internal angle of the generator using a rotary encoder on the shaft of the machine. This document describes the theory and experimental setup of this proposed system and outlines the test procedure of experimentation.

DEDICATION

This work is dedicated to my family. Thank you for always being there and believing in me. You have taught me over the years that I am capable of anything that I put my mind to. This document is proof that anything is possible and that the North American engineering spirit is truly alive and well. Thank you!

I hope that this document may serve as a concrete reference for machinery and machinery control.

Ce travail est dédié à ma famille. Je vous remercie d'être toujours là et de croire en moi. Vous m'avez appris au fil des ans que je suis capable de tout ce que je mets mon esprit à. Ce document est la preuve que tout est possible et que l'Amérique du Nord d'ingénierie esprit est bien vivant et bien. Je vous remercie!

ACKNOWLEDGEMENT

It should be noted that this project would not have been possible without the support of Rockwell Automation Incorporated. I would like to take the time to give a special thank you to Mr. Frank Kulaszewicz and Mr. Blake Moret for financing this project. Additionally, I would like to thank Mr. Fred Discenzo and Mr. David Mayewski for helping the throughout this process. Most importantly, I would like to thank my advisor Dr. Jaime De La Ree for standing by my side and for his intricate guidance on this project.

TABLE OF CONTENTS

Contents

ABSTRACT	ii
Dedication	iii
Acknowledgement	iv
Table of Contents	v
Table of Equations	x
Table of Figures	xii
1. Chapter 1 Introduction	1
A. Introduction	2
2. Chapter 2 Previous Attempts at Measuring Rotor Angle	3
A. Introduction	4
B. V.A. Kinitsky	4
1) Solution Overview	5
2) Pros of Solution	9
3) Cons of Solution	9
C. D.G. Lewis and W. E. Austin	10
1) Solution Overview	10
2) Pros of Solution	11
3) Cons of Solution	12
D. Virginia Tech Approach	12
3. Chapter 3 Machine Models	14
A. Introduction	15
B. Steady State Model Cylindrical Rotor	15
1) Rotor Self-Inductance	18
2) Stator-to-Rotor Mutual Inductances	19
3) Stator Inductances	20
C. Steady State Model Salient Pole Rotor	24
1) D-Q-0 Reference Frame	25
2) Direct and Quadrature Axis Quantities	27
3) Equivalent Circuit	28
D. Seven Variable Dynamic State Space Model	29
1) Definitions	29

2) Resistance Matrices	30
3) Flux Linkage Equations	31
4) Overall Flux Linkage Matrix.....	31
5) Voltage Equations in Matrix Form	33
6) Torque.....	35
7) Acceleration Equation	35
8) Solving State Space Equations	35
E. Conclusions	36
4. Chapter 4 Plant Componentes That Utilize Torque Angle.....	38
A. Introduction	39
B. Active Power and Frequency Control	39
1) Speed Governor Fundamentals	40
2) Isosynchronous Governor	41
C. Exciter Systems	42
1) V/Hz Limiter	43
D. Power System Stabilizer	44
E. Applications	44
1) Local Area Applications.....	44
2) Wide Area Applications	46
F. Conclusions	46
5. Chapter 5 Time Synchronization	47
A. Introduction.....	48
B. Network Time Protocol.....	49
C. Global Positioning System (GPS)	50
1) Definition.....	50
2) Pulse-Per-Second (PPS)	50
3) IRIG-B	50
D. Precision Time Protocol.....	51
1) Definition.....	51
2) Master-Slave Relationship.....	52
3) The Use of Tunable Clocks	54
4) Clock Type	56
E. Common Industrial Protocol.....	58
F. Quality of Service (QoS)	59

G. Conclusion	61
6. Chapter 6 Experimental Setup and Component Justification	62
A. Introduction	63
B. Mechanical Components.....	63
1) Minimum Encoder Specifications	63
2) Mechanical Couplings	65
C. Embedded Solution vs. Automation Controller	66
D. Programmable Logic Controller vs. Programmable Automation Controller for Power Applications	66
1) Local Control Logix Rack	68
2) Remote I/O Rack	69
3) Network Hardware	72
E. Network Topologies and Architectures.....	76
1) Linear Topology	76
2) Device Level Ring	77
3) Star Topology	77
F. Chosen Network Topology.....	78
G. Software Algorithm	81
H. Conclusions.....	84
7. Chapter 7 Experimental Trials and Analysis.....	86
A. Introduction.....	87
B. Infinite Bus Connection	87
1) Synchroscope.....	88
2) Infinite Bus No Time Synchronization.....	90
3) Infinite Bus Time Synchronized via PTP	91
C. Constant Frequency Source	96
D. Dynamometer Loading.....	98
E. Conclusions	102
8. Chapter 8 Conclusions and Future Work	103
A. Overall Conclusions	104
B. Future Work	108
9. Works Cited.....	110
10. Appendix A Three Phase Machinery Primer	112
A. Windings	113
B. Saliency.....	113

C. Electromagnetic Field Interactions	113
D. Definition of Synchronism.....	115
E. Functionality-- Single Phase Machines	115
F. Three Phase Machines	117
G. Torque.....	118
11. Appendix B Seven State Matrix Reduction	121
1) Ls Matrix.....	122
2) Lsr Matrix	122
3) Lr Matrix.....	122
4) Magnetization Inductances	123
5) Mutual to Magnetizing Inductance Relationship	123
6) Reduction of the Lsr Matrix	124
7) Reduction of the Lr Matrix.....	124
8) Flux Linkage Matrix Equation.....	124
9) Revised Voltage Matrix Equations	124
10) Transformation of the Ls Matrix	125
11) Transformation of the Lr Matrix.....	125
12) Transformation of the Lsr Matrix	125
13) (Lsr)T transformed.....	126
14) Ls Transformed	126
15) Lr Transformed	126
16) Speed Voltages	126
12. Appendix C Precision Time Protocol.....	128
A. Best Master Clock Algorithm	129
B. CIP Sync Object.....	131
13. Appendix D Component Selection.....	133
A. Local Rack Component Descriptions	134
B. Remote Rack Components	138
14. Appendix E Infinite Bus Test Data	140
A. No Time Synchronization Data.....	141
B. Varying Frequency with Time Synchronization Data	148
C. Fluke Metering Test Data.....	173
15. Appendix F Dynamometer Testing Results.....	198
A. 7.5 HP Machine	199

1) Raw Data	199
2) Plots	203
B. 10 HP Machine	205
1) Raw Data	205
2) Plots	211
16. Appendix G MATLAB Code	213

TABLE OF EQUATIONS

Equation 2.1	5
Equation 2.2	6
Equation 2.3	7
Equation 2.4	7
Equation 2.5	11
Equation 3.1	15
Equation 3.2	18
Equation 3.3	18
Equation 3.4	19
Equation 3.5	19
Equation 3.6	20
Equation 3.7	20
Equation 3.8	20
Equation 3.9	20
Equation 3.10	21
Equation 3.11	21
Equation 3.12	21
Equation 3.13	21
Equation 3.14	22
Equation 3.15	22
Equation 3.16	22
Equation 3.17	22
Equation 3.18	22
Equation 3.19	23
Equation 3.20	23
Equation 3.21	23
Equation 3.22	23
Equation 3.23	24
Equation 3.24	27
Equation 3.25	27
Equation 3.26	30
Equation 3.27	30
Equation 3.28	30
Equation 3.29	31
Equation 3.30	31
Equation 3.31	32
Equation 3.32	32
Equation 3.33	32
Equation 3.34	33
Equation 3.35	33
Equation 3.36	35
Equation 3.37	35
Equation 3.38	35
Equation 3.39	36

Equation 3.40	36
Equation 3.41	36
Equation 4.1	40
Equation 6.1	64
Equation 6.2	64
Equation 7.1	95
Equation 7.2	95
Equation 7.3	95
Equation 7.4	95
Equation 10.1	115
Equation 10.2	115
Equation 10.3	116
Equation 10.4	116
Equation 10.5	116
Equation 10.6	116
Equation 10.7	116
Equation 10.8	117
Equation 10.9	117
Equation 10.10	117
Equation 10.11	117
Equation 10.12	117
Equation 10.13	118
Equation 10.14	118
Equation 10.15	118
Equation 10.16	118
Equation 10.17	118
Equation 10.18	119
Equation 10.19	119
Equation 10.20	119
Equation 10.21	119
Equation 10.22	120
Equation 10.23	120
Equation 10.24	120
Equation 10.25	120
Equation 10.26	120

TABLE OF FIGURES

Figure 1 Short Circuit Characteristic	6
Figure 2 Vector Diagram	6
Figure 3 Kinitzky Experimental Setup.....	8
Figure 4 Lewis and Austin Experimental Setup	11
Figure 5 Cylindrical Rotor Synchronous Machine Model.....	16
Figure 6 Inductance Seen By Two Pole Cylindrical Rotor Machine.....	17
Figure 7 Steady State Model Synchronous Motor.....	24
Figure 8 Steady State Model Synchronous Generator.....	24
Figure 9 Salient Pole Machine Inductance	26
Figure 10 Salient Pole Machine.....	29
Figure 11 Seven Variable Model	34
Figure 12 Stator and Rotor Circuits	34
Figure 13 Governor Principles.....	40
Figure 14 Speed Governor Block Diagram	42
Figure 15 Exciter System.....	42
Figure 16 Paralleled Machine Setup	45
Figure 17 Synchronized Timing	49
Figure 18 System of Clocks	53
Figure 19 Master and Slave Open Loop	54
Figure 20 Periodic Re-sync.....	55
Figure 21 Tunable Crystal	56
Figure 22 Transparent Clock Operation	58
Figure 23 QoS Example.....	60
Figure 24 Control Logix Rack	68
Figure 25 Experimental PAC Layout	69
Figure 26 Point I/O Implementation	71
Figure 27 Managed Switch	73
Figure 28 Cut through Definition	75
Figure 29 Network Topologies	76
Figure 30 Network Topology.....	78
Figure 31 Software Algorithm.....	83
Figure 32 Infinite Bus Connection.....	87
Figure 33 Synchronization Circuit.....	89
Figure 34 Control Schematic	90
Figure 35 Non-Time Synchronized System.....	91
Figure 36 Time Synchronized Samples	92
Figure 37 Test Setup Frequency Test	93
Figure 38 Time Synchronized Encoder Trial Frequency Test.....	94
Figure 39 Frequency Controlled Source Setup	96
Figure 40 Frequency Controlled Source	97
Figure 41 Dynamometer Testing Setup	98
Figure 42 Dynamometer Testing 7.5 HP	99
Figure 43 40 N-m Test 7.5 HP Machine.....	100
Figure 44 Dynamometer Loading 10 HP Machine.....	100

Figure 45 30 N-m Test of 10 HP Machine 101

Figure 46 Synchronization Messages (latency) 130

Figure 47 CIP Sync Object Model..... 131

CHAPTER 1 INTRODUCTION

A. Introduction

Synchronous generators or alternators are electric machines which convert mechanical energy into AC electrical energy. The power industry has for many years, and continues today, to utilize synchronous machines as the main form of producing electrical power [1]. It is well known that the angular stability of a power system is closely related to the stability of its generator units. In order to assess the stability and/or stability margins of a power system, the angular displacement or torque angle of each machine must be evaluated. Input information provided by load-flow studies, state-estimation studies, and models of the machines must be used in conjunction to estimate the internal torque angle of each individual machine.

Today with the advent of precise timing protocols, fast speed encoders, and data acquisition systems, it is possible to measure directly the angular shaft displacement of a synchronous machine. This measurement would/could alleviate time delays and inaccuracies due to machine models' parameter errors and approximations. At the same time, if GPS timing systems are incorporated into the measurement system, it is possible to convert the resulting measured information to a stream of data similar to that provided by Phasor Measurement Units (PMUs) to be used at a central location for advanced control, monitoring, and/or protection systems.

In order to understand the theory and applications presented in this document the reader must have a basic understanding of the operation of three phase electrical machinery. A basic three phase machinery review can be seen in Appendix A of this document.

CHAPTER 2 PREVIOUS ATTEMPTS AT MEASURING ROTOR ANGLE

A. Introduction

The torque angle is a key parameter to assess the stability and to provide control of synchronous machines. Many previous attempts have been made to measure the torque angle of synchronous machines, but limitations in technology and timing issues have prevented prior researchers from obtaining reasonable results. Results from previous studies could be used to marginally improve the feedback control of machines. Many of these studies performed on synchronous machines were conducted during the 1950s and 1960s. Research has shown that relatively little advancement in this field has evolved over the last 50 years.

This chapter will discuss the works of V.A. Kinitsky, D.G. Lewis, W.E. Austin, as well as the proposed Virginia Tech solution to this problem. The chapter will outline each contributor's work as well as discuss the pros and cons to each contribution to the art of measuring the rotor angle of synchronous machines.

B. V.A. Kinitsky

V.A. Kinitsky laid the founding work for the theory of measuring the rotor angle of a synchronous machine. Kinitsky, a ranking AIEE member, first begins by stating that the rotor angle plays a key role in the operation of synchronous machines. He goes further into detail outlining that power output and the stability of the synchronous machine depend very much on this key parameter. Kinitsky continues to elaborate on the importance of the internal angle of the synchronous machine by stating that changes in load or excitation will alter the rotor displacement angle, and [2] he acknowledges the fact that at the time, there are two arts to measuring the rotor angle of the machine which include:

Measuring the torque angle from an auxiliary synchronous machine with the same number of poles as the main machine fixed on its shaft [3].

Measure the torque angle via the stroboscopic effect. The Virginia Tech solution presented in this document takes the concept behind the analog stroboscope, and with the use of a high speed encoder creates a digital stroboscope.

Kinitsky makes a key point in stating that these two methods of measuring the torque angle of a synchronous machine were only applicable during testing , not normal operation. The solution proposed by Kinitsky only addresses cylindrical rotor machines.

1) Solution Overview

The first solution that Kinitsky specifies is for a cylindrical rotor machine. He first discusses that the power output from the synchronous machine is described by the following equation where m is the number of phases, E_t is the terminal voltage line to neutral, I_d is the steady state short circuit current, and δ is the rotor displacement angle:

$$P = mE_tI_d\sin\delta [W]$$

Equation 2.1

Using this equation, the steady state stability limit of a cylindrical rotor synchronous machine with negligible armature resistance can be found when delta is equal to 90 degrees. He continues by stating that the no load and steady state short circuit characteristics of a synchronous machine are described by Figure 1, where I_d is the steady state short circuit current, I_f is the field current, and α is the angle between the short circuit characteristic and abscissa (x-axis). From this figure, equation VIII.2 is given as the following.

Figure 1 Short Circuit Characteristic

$$I_d = I_f \tan \alpha$$

Equation 2.2

Figure 2 Vector Diagram

Since this approximates to almost a straight line, this equation reduces to the following:

$$I_d = KI_f$$

Equation 2.3

By performing substitutions on these equations the sine of the torque angle is given below, where I is the armature current, and φ is the phase angle between the terminal voltage and armature current (the power factor angle):

$$\sin\delta = \frac{E_t I \cos\varphi}{KE_t I_f}$$

Equation 2.4

Kinitsky uses the previous equation, also seen in vector form in Figure 2, for his experimental setup for measuring rotor angle. He states that the measurement device should consist of two watt meters attached to the same shaft. The deflection of the watt meters should be proportional to the power output as well as the steady state stability limit [2]. Additionally, Kinitsky uses two dynamometers, A and B in order to measure the rotor angle of a cylindrical rotor machine. The connections can be seen in the Figure 3 below.

Figure 3 Kinitsky Experimental Setup

The deflecting torque of the dynamometer is dependent upon the sine of the angle between the moving and fixed coils. The maximum deflection will occur when the coils are at right angles to each other while minimum deflection will occur when the coils are in the same plane. The fixed coils of both the A and B dynamometers are displaced in the same plane, while the moving coils are fixed to the shaft of the machine at right angles to each other [2]. As a result, when the torque of the upper dynamometer equals the torque of the lower dynamometer, the instrument will be at rest.

The coils of the A dynamometer are connected to the armature windings of the synchronous machine and as a result the deflecting torque is proportional to the power output of the machine multiplied by the cosine of the shaft deflection angle. The torque read by the B dynamometer is proportional to the terminal voltage multiplied by the field current and the sine of the shaft deflection angle. These results are proportional to the sine of the rotor displacement angle [2]. As a result, by taking the ratio of these two quantities the sine of the rotor angle is calculated for the cylindrical rotor machine.

2) Pros of Solution

Up to this point in time, the community only had two ways to measure the rotor angle of a synchronous machine: measuring off an auxiliary synchronous machine, or using the stroboscopic effect. By utilizing two dynamometers mounted to the shaft of the synchronous machine and pulling signals from the armature and field respectively the ratio of the two dynamometer readings will yield the rotor angle measurement. This method allows for a true measurement during normal operation of a synchronous machine. Accuracy of this system was tested to have errors ranging from zero degrees to 0.5 degrees [2]. The method of rotor angle measurement used by Kinitsky can easily be implemented into field rheostat control for synchronous machines.

3) Cons of Solution

While the work by Kinitsky provided the community with a breakthrough in the development of a device that could accurately measure the rotor angle of a synchronous

machine, there were still flaws to his methods. This solution presented by Kinitzky only discusses rotor angle displacement based on steady state operating conditions[2]. Theory for transient results would need to be developed in order to model the machine throughout various scenarios to which it may be subjected. Due to mechanical vibrations, the equipment used by Kinitzky would require careful periodic calibrations. Finally, there was no way for this method to provide trending or storage of any information recorded useful for post mortem analysis. Additionally, this method only was valid for cylindrical rotor machines, and would not work properly for salient pole machines due to machine model differences.

C. D.G. Lewis and W. E. Austin

The next major contribution to rotor angle measurement would not come for another eleven years (1969). D.G. Lewis and W.E. Austin developed a device that would accurately record the rotor load angle of a synchronous machine during transients. They state that devices previously used to measure rotor angle implemented various filters, and as a result, caused unwanted phase shifts as well as restricted the response times of the measuring devices[4].

1) Solution Overview

The device that Lewis and Austin developed relies on a pulse-width/pulse-height conversion technique using a staircase converter. The pulse-height output is immediately gated after being converted from an ultra violet recorder galvanometer. The galvanometer deflects a distance proportional to the peak current flowing in the system. In essence, the work by Lewis and Austin essentially produced the first analog encoder. By gating the ultraviolet light, based on the value of current in the system, they would produce a set of discrete values that would range from zero to a finite maximum proportion of the max current. This can be viewed as the bit of an encoder ranging from the values shown in equation 2.5, where n is the number of bits of the encoder.

$$0 \rightarrow 2^{n-1} - 1$$

Equation 2.5

Figure 4 Lewis and Austin Experimental Setup

Figure 4 and shows the experimental setup by Lewis and Austin. The results are shown on chart paper with two degrees per centimeter accuracy. During testing of their device, Lewis and Austin state that the accuracy of their device is within one degree on the most sensitive settings.

2) Pros of Solution

The solution developed by Lewis and Austin provides an accurate way to visually display the rotor angle of a synchronous machine. By implementing staircase quantizers, they eliminated previous problems due to filter phase shifts and restricted device response times. The rotor angle recorder is a simple, yet effective way to measure the torque angle of a synchronous machine. The calibrated output of the recorder is interpreted off an easy to read chart, and the precision of the device can be easily adjusted. One final advantage of the device is that it does not need to be calibrated after its initial calibration. Finally, this solution can easily be integrated into a multi-machine system.

3) Cons of Solution

While the device by Lewis and Austin was easy to implement, it had one fatal flaw, it could not store any data. The only task that this device would perform was to provide the rotor angle of the machine in graphical form. While this is great for testing purposes, it does not provide a signal that can be used as feedback to control synchronous machines. Another flaw of this system is its accuracy. The accuracy of this system is actually worse than the device developed by Kinitsky eleven years earlier. The system is based on recording data from an AND gate off a 1MHz clock. The accuracy of this clock may not be precise enough from drift to provide accurate results for wide area applications.

D. Virginia Tech Approach

The main setback to the solution provided by Kinitsky was the required frequent calibration that needed to be provided to his equipment setup. Additionally, he lacked a way to measure rotor angle under transient conditions. Lewis and Austin, on the other hand, lacked a tunable precise timing device. Finally, both solutions lacked some form of data storage.

The approach used by the author for this application to measure the rotor angle of synchronous machines leverages current technologies developed for the industrial marketplace. If the rotor position is measured at (time-tagged) equal time intervals, the information collected yields data that allows researchers to perform post mortem analysis on transient events that will affect the performance of the synchronous machine or to develop applications for control.

The approach used in this document utilizes the accuracy of GPS time distributed over a variable latency packet switched network in order to record the rotor angle position of the synchronous machines. A single turn absolute encoder will be axially mounted to the shaft of the machine. With careful consideration of how the encoder is installed, the encoder value read will yield the physical delta angle of the synchronous machine with

respect to the electrical phase A axis. Additionally, a data logging system was implemented in order to create a database of information where accurate timestamps and angle measurements could be stored for future applications (real-time or offline).

As there are no analog filters implemented on this system, the values read would help represent transient events that the synchronous machine may be subjected. The key importance to this system is that precise timing had to be maintained over a variable latency packets switched Ethernet based network. Chapters five and six discuss how timing was maintained and the components chosen for this project.

CHAPTER 3 MACHINE MODELS

A. Introduction

When discussing machine models for synchronous alternators, three main models are generally used. These models include a source behind a reactance, a source behind a sub-transient reactance, and a seven state variable model. Each of these models has their pros and cons. As more intricacy is added to the model, a more accurate result will be generated, but at the cost of processing time.

This chapter will discuss each of the models in detail, deriving their existence and explaining their significance in the world today. In addition, pros and cons of each model will be noted throughout this chapter.

B. Steady State Model Cylindrical Rotor

The first model discussed in this paper is the steady state model of a synchronous alternator. Synchronous machines, just like all AC machinery, operate on the principle of the interaction of rotating magnetic fields to produce induced voltages [5]. Equation 3.1 defines the relationship between induced stator voltages and fluxes produced by interacting magnetic fields.

$$V_{as} = r_s i_{as} + \frac{d\lambda_{as}}{dt} [\text{Volts}]$$

Equation 3.1

Taking this equation into account, the value of lambda is determined by the inductance of each coil and their interaction with the other coils distributed throughout the rotor and stator.

For this derivation the simple Figure 5 below will be utilized to derive the equations for the steady state analysis model.

Figure 5 Cylindrical Rotor Synchronous Machine Model

Figure 6 shows the cross-sectional area of a three phase synchronous machine. Although Figure 6 shows a two pole machine, the same principle applies to a N pole machine. The coils of as-as', bs-bs', and cs-cs' are representative of the distributed windings throughout the stator which will produce the required magneto-motive forces (MMFs). The winding fd-fd' represents the distributed winding on the rotor that produces the stationary unidirectional magnetic field for the synchronous machine. The kd-kd' and kq-kq' windings represent the damper (amortisseur) windings used to dampen oscillations in the machine (discussed in seven state variable section).

Figure 6 Inductance Seen By Two Pole Cylindrical Rotor Machine

Figure 6 will be used to illustrate the inductance produced by a two pole cylindrical rotor synchronous machine. As the rotor rotates within the stator, the inductance seen between the rotor and phase A remains constant. This is because the distributed effect of the field windings in conjunction with the geometry of the rotor produce a constant inductance no matter the rotor orientation. Although this is a two pole machine the same principle applies for multi-pole machines.

The first step in computing the parameters of the steady state model are to define the flux linkages generated by the above figure in terms of each phase as well as the field winding. By use of Faraday's law, the following equations in 3.2 express the flux linkages from each of the windings:

$$\begin{aligned}\lambda_a &= L_{aa}i_a + L_{ab}i_b + L_{ac}i_c + L_{af}i_f \\ \lambda_b &= L_{ba}i_a + L_{bb}i_b + L_{bc}i_c + L_{bf}i_f \\ \lambda_c &= L_{ca}i_a + L_{cb}i_b + L_{cc}i_c + L_{cf}i_f \\ \lambda_f &= L_{fa}i_a + L_{fb}i_b + L_{fc}i_c + L_{ff}i_f\end{aligned}$$

Equation 3.2

In the equations 3.2, subscripts of matching letter represent the self-inductance of the corresponding winding, while miss-matched subscripts correspond to the mutual inductance between the two coils. It should be noted that each of these inductances can be calculated from manufacturers' design data or simple tests (i.e. open circuit and blocked rotor). There are three main types of inductance in any synchronous machine: rotor self-inductance, rotor-to-stator mutual inductance, and stator inductances.

1) *Rotor Self-Inductance*

In cylindrical rotor synchronous machines, the self-inductance of the rotor is independent of the angular rotor position due to geometry [1]. This assumption also neglects the harmonic effects as well as the slot openings of the stator. As a result of these assumptions the rotor self-inductance can be written as the following:

$$L_{ff} = L_{ff0} + L_{fl} \text{ [Henry]}$$

Equation 3.3

The first term of L_{ff0} accounts for the fundamental component of the air-gap flux created between the rotor and stator. The second term L_{fl} accounts for the leakage flux in the field winding that does not link any other coil. It is important to note that these

assumptions can only be made under steady state conditions. If the machine is operating in a transient condition, the flux linkages created in the field will vary with time, and as a result, the voltages generated in the rotor circuit (field and damping windings) will play a large role in the synchronous machines' performance [5].

2) *Stator-to-Rotor Mutual Inductances*

Unlike the rotor self-inductance, the stator-to-rotor inductances vary with angular rotor position θ . The angle θ is the angular difference between the magnetic axis of the rotor field winding and the armature winding of phase A. This quantity is also depicted in Figure 4. Assuming a sinusoidal space-mmF and air gap flux distribution, the mutual inductances between the field and stator inductance of phase A will vary by a cosine function[5]. The mathematical expression relating this mutual inductance is below.

$$L_{af} = L_{fa} = L_{af} \cos \theta \text{ [Henry]}$$

Equation 3.4

It should be noted that the mutual inductances between the rotor and B and C phase of the stator windings also vary by the cosine function with the only difference being θ is replaced by $\theta - 120$ and $\theta + 120$ respectively. When the rotor is rotating at synchronous speed ω , the angular rotor position can be calculated by the following equation.

$$\theta = \omega t + \theta_0 \text{ [radian]}$$

Equation 3.5

The value of θ_0 is the angular position of the rotor at time equal zero. When a DC current of magnitude I_f is applied to the rotor, the flux linkages created between phase A and the rotor is given by the equation 3.6:

$$\lambda_{af} = L_{af} I_f \cos(\omega t + \theta_0) \left[\frac{A - T}{m} \right]$$

Equation 3.6

3) *Stator Inductances*

Assuming a cylindrical rotor machine, and neglecting the effect of rotor slots, the air-gap between the rotor and stator are independent of angular position. As a result, the stator self-inductances are all equal in magnitude and constant values. The stator self-inductances can be expressed as the following:

$$L_{aa} = L_{bb} = L_{cc} = L_{aa} = L_{aa0} + L_{al} \text{ [Henry]}$$

Equation 3.7

where L_{aa0} is the self-inductance component due to the air-gap and L_{al} is the component of leakage flux of the armature. Another important fact to note is that the phase-to-phase mutual inductances are solely based on the air-gap flux. Since the stator windings are physically displaced by 120 degrees the mutual phase-to-phase inductances can be represented as the following:

$$L_{ab} = L_{ba} = L_{ac} = L_{ca} = L_{bc} = L_{cb} = -\frac{1}{2} L_{aa0} \text{ [Henry]}$$

Equation 3.8

The flux linkages on phase A can then be calculated as the following:

$$\lambda_a = (L_{aa0} + L_{al}) i_a - \frac{1}{2} L_{aa0} (i_b + i_c) + \lambda_{af} \left[\frac{A - T}{m} \right]$$

Equation 3.9

Assuming a balanced three phase set of currents, the following identities hold true:

$$i_a + i_b + i_c = 0 [A] \text{ or } i_b + i_c = -i_a [A]$$

Equation 3.10

Substituting this identity into equation 3.9 yields the following:

$$\begin{aligned}\lambda_a &= (L_{aa0} + L_{al})i_a + \frac{1}{2}L_{aa0}i_a + \lambda_{af} \left[\frac{A-T}{m} \right] \\ \lambda_a &= \left(\frac{3}{2}L_{aa0} + L_{al} \right) i_a + \lambda_{af} \left[\frac{A-T}{m} \right]\end{aligned}$$

Equation 3.11

At this point, it is important to define a synchronous machine quantity known as the synchronous inductance, L_s . The synchronous inductance is defined by the following equation:

$$L_s = \frac{3}{2}L_{aa0} + L_{al} [Henry]$$

Equation 3.12

Applying this new term to equation 3.11, the flux linkages due to phase A yields the following result:

$$\lambda_a = L_s i_a + \lambda_{af} \left[\frac{A-T}{m} \right]$$

Equation 3.13

The synchronous inductance is defined as the inductance seen by phase A under normal and balanced three phase operating conditions. The synchronous inductance is calculated by three components: self-inductance, leakage, and mutual inductance between coils. The first component is created due to the self-flux linkages due to the air gap between the rotor and stator. The second component accounts for the leakage between the phase coil and the armature. The final component results from the flux linkages produced from currents in phase B and phase C respectively.

Now that the flux linkage equations for each phase have been developed, the overall goal of this derivation is to develop the terminal voltage of the synchronous generator. This can be achieved by applying the flux linkages to equation 3.14:

$$V_T = r_s i_{as} + \frac{d\lambda_{as}}{dt} \text{ [Volts]}$$

Equation 3.14

This equation can also be re-written as:

$$V_T = r_s i_{as} + L_s \frac{di_{as}}{dt} + \frac{d\lambda_{af}}{dt} \text{ [Volts]}$$

Equation 3.15

Recalling that:

$$E_{af} = \frac{d\lambda_{af}}{dt} = -\omega L_{af} I_f \sin(\omega t + \theta_0) \text{ [Volts]}$$

Equation 3.16

This voltage, E_{af} , is defined as the excitation voltage, and is created from the flux produced by the rotating rotor field winding. By applying the following the trigonometric identity, the excitation voltage can be re-written as equation 3.18:

$$\sin \alpha = -\cos \left(\alpha + \frac{\pi}{2} \right)$$

Equation 3.17

$$E_{af} = +\omega L_{af} I_f \cos \left(\omega t + \theta_0 + \frac{\pi}{2} \right) \text{ [Volts]}$$

Equation 3.18

It can be seen that the excitation voltage leads the flux linkage by 90 degrees, which makes sense as the mathematical time derivative is a 90 degree phase shift in time. The root mean square value of the excitation voltage is given by the equation 3.19:

$$E_{af} = \frac{\omega L_{af} I_f}{\sqrt{2}} [Volts]$$

Equation 3.19

It should be noted at this point that the excitation voltage is usually related to the rotor field current by a constant k . It can be shown that the constant k is defined as the following:

$$k = \frac{\omega L_{af}}{\sqrt{2}}$$

Equation 3.20

The equation relating the terminal voltage to the excitation voltage is now given by equation 3.21:

$$V_T = R_a I_a + L_s \frac{di_a}{dt} + E_{af} [Volts]$$

Equation 3.21

Taking the steady state sinusoidal solution to this equation yields the following:

$$\overrightarrow{V_T} = R_a \overrightarrow{I_a} + jX_s \overrightarrow{I_a} + \overrightarrow{E_{af}} [Volts]$$

Equation 3.22

Equation 3.22 is for the terminal voltage of a synchronous motor as the armature current is defined as traveling from the terminals of the machine to the induced voltage. The diagram of this machine can be seen below in Figure 7.

Figure 7 Steady State Model Synchronous Motor

When the synchronous machine operates as a generator, rather than a motor, the armature current flows from the induced voltage to the terminals of the machine thus yielding the following equation:

$$\vec{V}_T = -R_a \vec{I}_a - jX_s \vec{I}_a + \vec{E}_{af} [Volts]$$

Equation 3.23

The equivalent circuit of this equation is shown below in Figure 8.

Figure 8 Steady State Model Synchronous Generator

C. Steady State Model Salient Pole Rotor

It was previously defined that a salient pole rotor is a rotor that is not cylindrical in nature, but has protrusions as the windings are wrapped around the poles of the rotor themselves. In addition, it was noted that under steady state for a cylindrical rotor synchronous machine there are four main elements in the equivalent circuit:

1. Excitation voltage E_A
2. Armature reaction of the synchronous generator
3. The stator winding's self-inductance
4. The stator winding's resistance

It should be noted that bullet points one, three, and four remained unchanged for salient pole theory. Point two must be modified in salient synchronous machine theory, as flux is easier to establish in some directions rather than others due to the geometry of the rotor. In order to better understand this concept a new transformation will be introduced to simplify calculations. This axis transformation is known as the D,Q,0 reference frame.

1) D-Q-0 Reference Frame

The synchronous machine armature quantities can be resolved into two rotating components. The first component is one that is aligned with the field winding axis known as the direct axis component, and one quadrature to the field winding axis known as the quadrature axis [1]. This conceptual transformation is very useful due to the fact that although each of the stator phases sees a position varying inductance as a result of the saliency of the rotor, the transformed quantities will rotate with the rotor. This concept can be seen in Figure 9. As a result, the transformed equivalent inductances see a constant magnetic path.

Figure 9 Salient Pole Machine Inductance

Due to the saliency of the rotor, the inductance seen by phase A coils can be seen in the graphs above. This inductance is a position dependent quantity, in that while the rotor changes position in time, so too does the inductance seen by the magnetic axis of phase A.

This transformation is quite useful when analyzing the interaction of the rotor and stator fluxes as well as their corresponding mmf waves. When one transforms the stator quantities into equivalent rotating quantities in steady state, the results obtained are constant mmf and flux waves separated by a spacial angle [1].

The corresponding transformation and inverse transformations are shown below in equations 3.24 and 3.25:

$$\begin{bmatrix} S_d \\ S_q \\ S_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos \theta_{me} & \cos(\theta_{me} - 120) & \cos(\theta_{me} + 120) \\ -\sin \theta_{me} & -\sin(\theta_{me} - 120) & -\sin(\theta_{me} + 120) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix}$$

Equation 3.24

$$\begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix} = \begin{bmatrix} \cos \theta_{me} & -\sin \theta_{me} & 1 \\ \cos(\theta_{me} - 120) & -\sin(\theta_{me} - 120) & 1 \\ \cos(\theta_{me} + 120) & -\sin(\theta_{me} + 120) & 1 \end{bmatrix} \begin{bmatrix} S_d \\ S_q \\ S_0 \end{bmatrix}$$

Equation 3.25

The quantity of S refers to the quantity to be transformed. The subscripts d, q, and 0 correspond to the direct axis, quadrature axis, and zero components respectively of the quantity being transformed. The zero component quantity represents the components of the armature current which does not generate any net air-gap flux, thus not linking the rotor circuit. Under balanced three phase conditions, this quantity is equal to zero. It should also be noted that this transformation requires instantaneous values of the quantities being transformed, not the RMS values.

2) *Direct and Quadrature Axis Quantities*

It was previously stated that the air-gap flux produced in a cylindrical rotor synchronous machine is independent of rotor position. Due to the protruding poles of a salient rotor, a preferred direction of magnetization is created. The reluctance along the direct axis is much less than that of the quadrature axis. In synchronous machines, the armature flux wave will lag the field flux wave by a spacial angle of $90 + \phi_{lag}$, where ϕ_{lag} is the phase angle by which the armature current pointing in the direction of the EMF lags the excitation EMF [5]. If the armature current lags the excitation voltage by 90 degrees, the armature flux wave is directly opposite the field poles and thus the opposite direction of the field flux. If the armature current is in phase with the excitation voltage, the armature flux wave becomes heavily polluted with third order harmonics due to saturation [5].

It should be stated that there is a high reluctance of the air-gap between poles. As a result, the fundamental armature flux when the field poles are in quadrature with the armature poles is much less than the fundamental armature flux when the armature is aligned directly aligned with the field poles [1]. Also, the previous statement can be rephrased as the following: the magnetizing reactance is less when the armature current is in time phase with the excitation voltage than when it is in time quadrature with respect to the excitation voltage.

The effect of saliency is taken into account by resolving the armature current into two parts: $\vec{I}_d$ and $\vec{I}_q$. The first component known as the direct axis armature current will produce a fundamental armature flux $\vec{\Phi}_{ad}$. The second component known as the quadrature axis armature current will produce a corresponding armature flux $\vec{\Phi}_{aq}$. The subscripts of d and q accordingly represent the direct and quadrature quantities that refer to the space phase of the armature fluxes rather than the time phase of the currents which produce the fluxes. In general any direct axis quantity is defined as one whose magnetic field effect is centered on the axes of the field poles. Also any quadrature axis quantity is defined as one whose magnetic field effect is centered on the inter-polar space.

3) Equivalent Circuit

Utilizing the D-Q-0 transform, Figure 10 shows vector diagram for a salient pole synchronous machine. It should be noted that the net flux, B, is not the same as the cylindrical rotor machine, due to geometry. As a result, the net per-phase terminal voltage will be the vector sum of the induced voltage with the direct and quadrature axis components of induced voltage.

Figure 10 Salient Pole Machine

D. Seven Variable Dynamic State Space Model

The previous two models will only address the steady state solutions for cylindrical rotor and as well as salient pole machines. In order to most accurately predict the response of a synchronous machine to various input conditions, the next few subsections will define the seven state variable model and describe how it is developed and why it is useful. The following derivation was developed using a synchronous machine that contains two damper windings, one on the direct axis, and one on the quadrature axis.

1) Definitions

The voltage equations for a three phase salient pole synchronous machine can be represented in the following form:

$$\begin{aligned}
v_{as} &= r_s i_{as} + \frac{d\lambda_{as}}{dt} \\
v_{bs} &= r_s i_{bs} + \frac{d\lambda_{bs}}{dt} \\
v_{cs} &= r_s i_{cs} + \frac{d\lambda_{cs}}{dt} \\
v_{kq} &= r_s i_{kq} + \frac{d\lambda_{kq}}{dt} \\
v_{fd} &= r_s i_{fd} + \frac{d\lambda_{fd}}{dt} \\
v_{kd} &= r_s i_{kd} + \frac{d\lambda_{kd}}{dt}
\end{aligned}$$

Equation 3.26

These equations can be reduced to equations in matrix form as seen below. Note that the subscripts of s and r represent stator and rotor quantities respectively.

$$\begin{aligned}
v_{abcs} &= r_s i_{abcs} + p\lambda_{abcs} \\
v_{qdr} &= r_r i_{qdr} + p\lambda_{qdr}
\end{aligned}$$

Equation 3.27

2) Resistance Matrices

The resistance of any synchronous machine is broken down into two categories, the stator resistance, and the rotor resistance. The stator resistance is represented as three equal resistances as the windings of the stator are symmetrical. The resistance of the rotor is represented by the resistance of each winding, field, and two damper windings respectively. These quantities are represented in the following two matrix equations.

$$\begin{aligned}
r_s &= \begin{bmatrix} r_s & 0 & 0 \\ 0 & r_s & 0 \\ 0 & 0 & r_s \end{bmatrix} \\
r_r &= \begin{bmatrix} r_{kq} & 0 & 0 \\ 0 & r_{fd} & 0 \\ 0 & 0 & r_{kd} \end{bmatrix}
\end{aligned}$$

Equation 3.28

3) Flux Linkage Equations

Equation 3.29 represents the flux linkages created between all windings of the synchronous machine.

$$\begin{aligned}
\lambda_{as} &= L_{asas}i_{as} + L_{asbs}i_{bs} + L_{ascs}i_{cs} + L_{askq}i_{kq} + L_{asfd}i_{fd} + L_{askd}i_{kd} \\
\lambda_{bs} &= L_{bsas}i_{as} + L_{bsbs}i_{bs} + L_{bscs}i_{cs} + L_{bskq}i_{kq} + L_{bsfd}i_{fd} + L_{bskd}i_{kd} \\
\lambda_{cs} &= L_{csas}i_{as} + L_{csbs}i_{bs} + L_{cscs}i_{cs} + L_{cskq}i_{kq} + L_{csfd}i_{fd} + L_{cskd}i_{kd} \\
\lambda_{kq} &= L_{kqas}i_{as} + L_{kqbs}i_{bs} + L_{kqcs}i_{cs} + L_{kqkq}i_{kq} + L_{kqfd}i_{fd} + L_{kqkd}i_{kd} \\
\lambda_{fd} &= L_{fdas}i_{as} + L_{fdbs}i_{bs} + L_{fdcsc}i_{cs} + L_{fdkq}i_{kq} + L_{fdfd}i_{fd} + L_{fdkd}i_{kd} \\
\lambda_{kd} &= L_{kdas}i_{as} + L_{kdbbs}i_{bs} + L_{kdcs}i_{cs} + L_{kdqk}i_{kq} + L_{kdfd}i_{fd} + L_{kdkd}i_{kd}
\end{aligned}$$

Equation 3.29

In matrix form these equations become the following:

$$\begin{bmatrix} \lambda_{abcs} \\ \lambda_{qdr} \end{bmatrix} = \begin{bmatrix} L_s & L_{sr} \\ (L_{sr})^T & L_r \end{bmatrix} \begin{bmatrix} i_{abcs} \\ i_{qdr} \end{bmatrix}$$

Equation 3.30

The L_s matrix represents the stator self-inductance matrix, L_{sr} , the stator to rotor inductance, L_{sr} transpose as the rotor to stator inductance, and L_r as the rotor self-inductance. Each of these matrices was subjected to the D-Q-0 transform to remove time dependency and move to a spacial domain. The analysis for this transformation can be seen in Appendix B of this document.

4) Overall Flux Linkage Matrix

Appendix B includes transformations and matrix reductions on equation 3.30 to reach equation 3.31.

$$\begin{bmatrix} \lambda_{qs} \\ \lambda_{ds} \\ \lambda_{0s} \\ \lambda_{kq} \\ \lambda_{fd} \\ \lambda_{kd} \end{bmatrix} = \begin{bmatrix} L_q & \dots & 0 & L_{mq} & \dots & \dots \\ \dots & L_d & 0 & \dots & L_{md} & L_{md} \\ 0 & 0 & L_{ls} & 0 & 0 & 0 \\ L_{mq} & 0 & 0 & L_{kq} & 0 & 0 \\ 0 & L_{md} & 0 & 0 & L_{fd} & L_{md} \\ 0 & L_{md} & 0 & 0 & L_{md} & L_{kd} \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \\ i_{kq} \\ i_{fd} \\ i_{kd} \end{bmatrix}$$

Equation 3.31

The ellipsis of the flux linkage matrix represents where quantities used to describe the “speed voltages”. This topic will be explained in Appendix B of this document. These ellipsis for the flux linkage equation are zero thus producing the following matrix:

$$\begin{bmatrix} \lambda_{qs} \\ \lambda_{ds} \\ \lambda_{0s} \\ \lambda_{kq} \\ \lambda_{fd} \\ \lambda_{kd} \end{bmatrix} = \begin{bmatrix} L_q & 0 & 0 & L_{mq} & 0 & 0 \\ 0 & L_d & 0 & 0 & L_{md} & L_{md} \\ 0 & 0 & L_{ls} & 0 & 0 & 0 \\ L_{mq} & 0 & 0 & L_{kq} & 0 & 0 \\ 0 & L_{md} & 0 & 0 & L_{fd} & L_{md} \\ 0 & L_{md} & 0 & 0 & L_{md} & L_{kd} \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \\ i_{kq} \\ i_{fd} \\ i_{kd} \end{bmatrix}$$

Equation 3.32

$$\begin{bmatrix} v_{qs} \\ v_{ds} \\ v_{0s} \\ v_{kq} \\ v_{fd} \\ v_{kd} \end{bmatrix} = \begin{bmatrix} pL_q & \omega L_d & 0 & pL_{mq} & \omega L_{md} & \omega L_{md} \\ -\omega L_q & pL_d & 0 & -\omega L_{mq} & pL_{md} & pL_{md} \\ 0 & 0 & pL_{ls} & 0 & 0 & 0 \\ pL_{mq} & 0 & 0 & pL_{kq} & 0 & 0 \\ 0 & pL_{md} & 0 & 0 & pL_{fd} & pL_{md} \\ 0 & pL_{md} & 0 & 0 & pL_{md} & pL_{kd} \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \\ i_{kq} \\ i_{fd} \\ i_{kd} \end{bmatrix}$$

Equation 3.33

5) Voltage Equations in Matrix Form

Accounting for the resistive elements yields the final matrix for voltage equations:

$$\begin{bmatrix} v_{qs} \\ v_{ds} \\ v_{0s} \\ v_{kq} \\ v_{fd} \\ v_{kd} \end{bmatrix} = \begin{bmatrix} r_s + pL_q & \omega L_d & 0 & pL_{mq} & \omega L_{md} & \omega L_{md} \\ -\omega L_q & r_s + pL_d & 0 & -\omega L_{mq} & pL_{md} & pL_{md} \\ 0 & 0 & r_s + pL_{ls} & 0 & 0 & 0 \\ pL_{mq} & 0 & 0 & r'_{kq} + pL'_{kq} & 0 & 0 \\ 0 & pL_{md} & 0 & 0 & r'_{fd} + pL'_{fd} & pL_{md} \\ 0 & pL_{md} & 0 & 0 & pL_{md} & r'_{kd} + pL'_{kd} \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \\ i_{kq} \\ i_{fd} \\ i_{kd} \end{bmatrix}$$

Equation 3.34

The model for the seven variable state space equations can be seen in Figure 11. This model represents the matrix equation 3.34. These circuits are magnetically coupled equations, where torque and speed are controlled by q-axis variables, while voltage is controlled by d-axis variables. In addition, Figure 12 depicts the electrical diagrams of the stator and rotor axes for a synchronous machine. The leftmost subfigure in Figure 12 shows the wye connected circuit of the stator, while the rightmost shows the equivalent direct and quadrature axis circuits of the rotor. The field and direct axis damper winding are represented on the same axis, while the q-axis damper winding is shown on the quadrature axis. In order to solve for the rotor angle, the equation 3.35 must be used and the integral of the difference between mechanical and synchronous speed taken:

$$\theta = \int (\omega_m - \omega_s) dt$$

Equation 3.35

Figure 11 Seven Variable Model

Figure 12 Stator and Rotor Circuits

6) *Torque*

The torque equation is given as the following:

$$T_e = \frac{3}{2} * \left(\frac{poles}{2} \right) (\lambda_{ds} i_{qs} - \lambda_{qs} i_{ds}) [N - m]$$

Equation 3.36

Equation 3.36 is a representation of the true work that this machine can provide electrically to a load.

7) *Acceleration Equation*

Equation 3.37 is the final equation necessary to solve this state space system of equations:

$$T_{mech} - T_{electrical} = J \frac{d\omega}{dt}$$

Equation 3.37

This equation sets the dynamics of any electromechanical system. This is a form of Newton's second law, where the forces are represented by the difference in torques, mass is represented by an inertia constant, and acceleration is the derivative of angular velocity. The difference between the mechanical torque applied to the shaft of a generator and electrical torque applied by a load will result in acceleration or deceleration of the machine until a new steady state operating point is reached.

8) *Solving State Space Equations*

The following state space equation method was used to solve the state space model previously described:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ V &= ZI \\ Z &= C + pD \end{aligned}$$

Equation 3.38

$$C = \begin{bmatrix} r_s & \omega L_d & 0 & 0 & \omega L_{md} & \omega L_{md} \\ -\omega L_q & r_s & 0 & -\omega L_{mq} & 0 & 0 \\ 0 & 0 & r_s & 0 & 0 & 0 \\ 0 & 0 & 0 & r'_{kq} & 0 & 0 \\ 0 & 0 & 0 & 0 & r'_{fd} & 0 \\ 0 & 0 & 0 & 0 & 0 & r'_{kd} \end{bmatrix}$$

Equation 3.39

$$D = p \begin{bmatrix} L_q & 0 & 0 & L_{mq} & 0 & 0 \\ 0 & L_d & 0 & 0 & L_{md} & L_{md} \\ 0 & 0 & L_{ls} & 0 & 0 & 0 \\ L_{mq} & 0 & 0 & L'_{kq} & 0 & 0 \\ 0 & L_{md} & 0 & 0 & L'_{fd} & L_{md} \\ 0 & L_{md} & 0 & 0 & L_{md} & L'_{kd} \end{bmatrix}$$

Equation 3.40

$$V = (C + pD)I$$

$$V = CI + pDI$$

$$V = CI + D\dot{I}$$

$$\dot{I} = D^{-1}V - D^{-1}CI$$

Equation 3.41

Solving this equation yields the rate of change in each of the currents with respect to time. If a small time increment is considered, the new state currents can be resolved. It can be seen that this model is computationally intensive. This model will yield the most accurate results when analyzing a machine, but at the cost of the high computation time.

E. Conclusions

This chapter has explored three different models for synchronous machines: steady state model cylindrical machine, steady state model salient pole machine, and seven variable state space models. As the model increases in complexity, it is a better representation of the machine's performance under various conditions. The downside to the increase in

accuracy is also an increase in computing time in order to find a solution to the model. For the remainder of this document, the seven variable model was used to validate the experimental setup described in chapter 6.

CHAPTER 4 PLANT COMPONENTS THAT UTILIZE TORQUE ANGLE

A. Introduction

It was previously stated that the torque angle of the synchronous machine is a key parameter for generation stability as well as power output. This chapter will explore the various control systems that utilize the torque angle of the synchronous machine either directly or indirectly.

When referring to the power network stability, one must look at the control of active and reactive power. These two network parameters are independent of each other, and as such, are governed by separate control algorithms. The torque angle of the synchronous machine is used as a feedback or input parameter in many different power system control applications. This section will explore both active and reactive power control schemes utilized by present day generation plant components.

B. Active Power and Frequency Control

Electrical frequency is a key parameter that should maintain relatively constant for proper power system performance. By precisely controlling frequency, one can ensure the relatively constant speed of both induction and synchronous motors that are run across the line. A considerable drop in frequency could mean magnetizing currents of damaging magnitudes to both induction machines as well as transformers [6]. At one point in time, frequency was important to the point that clocks directly connected to outlet used the frequency of the power system to meter time.

Kundur states that the electrical frequency of a system is dependent upon active power balance across the network [6]. Since frequency is a shared parameter across the system, a change in active power at one point in the system is often exhibited as a change in frequency at a subsequent location. There are many generators that are suppliers of active power to the network, and as a result, there must be some means to allocate the change in demand of the generators. The speed governor at each generation station provides the primary speed control function [1].

1) Speed Governor Fundamentals

A load change on the power system is immediately reflected as a change in electrical torque output of the generator. This change in electrical torque causes a difference between the supplied mechanical torque applied to the shaft of the generator and the output electrical torque. As a result, there will either be acceleration or deceleration of the synchronous machine as described by equation 4.1.

$$T_e - T_m = J \frac{d\omega}{dt}$$

Equation 4.1

Since there is a change in angular velocity, there is in turn a change in electrical frequency away from the nominal value. Equation 4.1 is also displayed in a visual format in Figure 13.

Figure 13 Governor Principles

When a load is switched on or off in the power system network, this change is reflected all the way back at the generation unit. It can be seen from both equation 4.1 and Figure 11, that the mechanical torque applied to the shaft of the machine and the electrical torque produced at the terminals of the machine are in counteracting directions. When a load is switched on or off in the network, the system is no longer in steady state, and the machine will either accelerate or decelerate until a new steady state operating point is reached, if that point exists. If no speed governor is implemented in the system, the time that the system takes to reach a new operating point, is dependent upon the mechanical

inertia and damping constants. The speed governor will introduce an additional feedback signal aiding in the response time of the system to reach a new steady state operating condition. For the purpose of simplicity, the model of the speed governor examined in this chapter will be the isochronous governor.

2) Isosynchronous Governor

Figure 14 depicts the workings of the isosynchronous speed governor. The governor works by measuring the mechanical speed of the machine and comparing it against a set mechanical speed (related to the nominal system frequency) and integrates this value. The integral of the mechanical speed difference yields a signal proportional to rotor angle deviation. In essence, the governor is trying to match the mechanical speed of the shaft of the machine to the synchronous speed reference of the system. By changing the valve position, more steam will generate more mechanical torque thus creating an acceleration of the system reaches a new steady state operating point.

The proposed system would improve the calculation of a new valve position by eliminating an integrator block, which takes time, and replacing it with a robust measured quantity. In essence, the summer and integrator could be eliminated from this speed governor setup, thus improving the response time of the system. The principle on how the governor would function remains the same. The error between the commanded position and the actual position would be reduced to zero. The only reference change that would need to be made would be an adjustment to the proportional gain, K , in order to scale the value sent to the controlled steam valve.

Figure 14 Speed Governor Block Diagram

C. Exciter Systems

The function of exciter systems is to provide a DC current to the field winding of a synchronous machine. This DC current establishes an induced voltage as described by the model. By controlling the magnitude of the induced voltage, it is possible to control the flow of reactive power [6]. The exciter system also performs protection of the synchronous machine in that; it will not exceed its capability limits.

Figure 15 Exciter System

Figure 15 depicts a high level diagram of a pilot exciter system. It can be seen that there are three components of the exciter system, which can be thought of as three separate machines. The first of three machines is the pilot exciter. The pilot exciter rotor is embedded with permanent magnets in order to establish a constant magnetic field. The pilot exciter armature windings feed a three phase rectifier and the output of the rectifier to the exciter field of the second machine. The second machine, or main exciter, is an inside-out machine in that the three phase windings are on the rotor, while the field is applied to the armature. The exciter armature is then rectified and the output sent to the main synchronous generator field to establish an internal voltage.

In order to ensure that the synchronous machine stays within its designed operating limits, limiters and protective circuits were developed for protection purposes. These circuits were devised in order to ensure that the exciter as shown in Figure 15 maintains a stable operating point within the capability limit of the machine. Examples of these protective circuits can be seen in field current limiters, volts-per-hertz (V/Hz) regulators, and under-excitation limiters.

1) V/Hz Limiter

The volts-per-hertz limiters are used to protect both generators and their corresponding transformer units from damage due to low frequency or overvoltage. Both of these conditions are caused by excessive magnetic flux, and if sustained for long periods of time will damage both the generating unit and transformer. The V/Hz limiter will control the field voltage to limit the terminal voltage when the V/Hz value exceeds a preset constant. The corresponding protection device will trip the generating unit when the V/Hz value is exceeded for an extended period of time [6].

Both the limiter and protective device measure terminal voltage and electrical frequency. The Virginia Tech solution will easily measures the angular displacement, but the mechanical frequency can be derived by taking the derivative of the position signal. The mechanical frequency can be related to the electrical frequency via the number of poles of the synchronous machine.

D. Power System Stabilizer

Power system stabilizers (PSSs) utilize external signals in order to control the excitation system to improve dynamic system performance [6]. PSSs were originally developed in order to dampen power system oscillations resulting from constant changes to automatic voltage regulators [7]. Key input parameters to PSSs include mechanical shaft speed, terminal frequency, and power [7]. The proposed system presented in this document will provide angular position as well as mechanical shaft velocity.

E. Applications

1) Local Area Applications

By being able to measure the rotor angle of a synchronous machine directly, the following four local area control applications can be realized: speed governor control, exciter system control, local power system stabilizer control, and damping of paralleled inter-machine oscillations. The first three applications have already been discussed in this chapter. The fourth item identified will now be discussed.

a) Paralleled Machine Connection

Most generating stations parallel multiple turbine-generator setups to the electrical bus to which they provide power. As a result, they can utilize the steam created multiple times to drive multiple turbines, increasing process yield. The down side to this setup is inter-machine oscillations. Take a simple example where two generators are paralleled on the system. The first turbine will be a high pressure turbine and the second a low pressure turbine. The steam is passed through the high pressure turbine and exhausted to the low pressure turbine. The steam is then passed through the low pressure turbine, and then exhausted or recycled depending on the system. This setup can be seen in Figure 16.

Figure 16 Paralleled Machine Setup

Current practice is to place a Phasor Measurement Unit (PMU) at the electrical bus of the system. As a result, only the conglomerate effect of the system is shown. As the steam is passed through the high pressure turbine first, that machine will be brought to synchronous speed first, and tend to pull G2 up to speed. G2 will overshoot and attempt to slow down. Consequently the two machines will oscillate against each other as they attempt to operate at a common synchronous speed. The proposed solution in this document will measure each machine's contribution of the oscillation, rather than the total effect, so that control can be coordinated to the proper machine in order to quickly dampen these oscillations.

2) Wide Area Applications

Furthermore, synchronized rotor angle measurements can provide substantial benefit to the following wide area applications: center of angles, angle coherency, and coordinated PSS control. Center of angles is a reduction technique where a geometric centroid will represent an equivalent machine. A weighted average will be conducted based upon machine inertia and an equivalent location then calculated. The proposed method directly measures rotor angle, so machine models will not need to be run in order to estimate the internal angle of the machine. Angle coherency is a way to evaluate how machines will group with each other during stressed power system conditions. Again, the angle of the machine can be directly measured and can replace model estimates. Finally, with a precise time tag applied to rotor angle measurements, wide area coordinated PSS control can be implemented since all rotor angle measurements are on the same time reference.

F. Conclusions

As described in the prior sections, there are a variety of plant components that can utilize the rotor angle measurement either directly or indirectly. These components include speed governors, exciter systems, and power system stabilizers. The proposed system will allow for benefits in both local and wide area control for power systems. Locally, improvements can be realized in speed governor control, exciter systems, local power system stabilizers, and local inter-machine oscillations. Wide area applications can also be realized for center of angles calculations, angle coherency, and coordinated control of power system stabilizers.

CHAPTER 5 TIME SYNCHRONIZATION

A. Introduction

In today's society, industry is relying more heavily on time synchronization for the administration of their everyday processes. Almost everything that we do in our lives relies on a clock or reference to time and as such it is important that our clocks are synchronized. Examples of time synchronization in industry today can be seen in input time stamping, sequence of events recording, and motion control.

Let's first take the example shown below in Figure 17 with a local controller shown to the left and a distributed input/output I/O device connected by a network media shown in blue. The input device shown here is a normally open switch. Prior to advancements in precise timing technology when a normally open contact was closed, only information containing data was sent back to the controller.

With today's technology to synchronize clocks, it is now possible to send both the data and a meaningful timestamp back to the controller, providing more information. Currently, many industries utilize time synchronization to coordinate sequence of events measurements, time stamped data logging, as well as coordination with GPS time. Since the clocks of the controller and the I/O device are time synchronized, scheduled outputs, and synchronized actuation can be achieved. Common ways to synchronize systems are Network Time Protocol, Global Positioning System, and Precision Time Protocol.

Figure 17 Synchronized Timing

B. Network Time Protocol

Network time protocol (NTP) is a protocol for synchronizing clocks of computers using a data network, such as the intranet or a wide area network (WAN). This protocol was developed at the University of Delaware in 1980 and it is the first protocol to address time synchronization over variable latency packet switched networks [8]. This protocol provides accuracies that depend on the setup of the network between each device and the performance of the computers' operating systems [9]. Ideally, the connections in the network should be as short as possible, but this protocol does include methods to estimate and account for round-trip path delay. Overall, the accuracy of this protocol is in the low tens of milliseconds over wide area networks (WANs) and better than a millisecond over local area networks (LANs) [8].

C. Global Positioning System (GPS)

1) Definition

Global Positioning System (GPS) is not only a navigation system, but has evolved to become the world's primary mean of distributing precise time and frequency. GPS was developed by the United States Department of Defense (DoD) in 1978, and is still maintained by the organization today [10]. Currently, 31 satellites orbit the earth and provide accurate time within ± 10 ns to GPS receivers [8]. The US DoD dictates the accuracy of the system, and has the right to limit this accuracy. Additionally, GPS systems can be costly to deploy over a distributed I/O network as a GPS receiver is required at each node, and the antenna must have a clear view of the sky to attain a locked signal.

While the accuracy the GPS provides is great for time synchronization, there are a few drawbacks to implementing GPS in industrial applications. The first drawback is that the location of the plant or process must have a clear line of sight of at least four satellites for time and position, but just three satellites for time. Also the cost of implementing a GPS receiver at every node within a network can become costly. It should also be noted, that GPS technology is also rapidly decreasing in cost, thus becoming a more viable option for time synchronization applications.

2) Pulse-Per-Second (PPS)

With respect to power system operation and control, GPS provides a very useful trigger signal for many devices and tests known as the Pulse-per-Second or PPS. The PPS is defined by its name, a square wave pulse every second of time on the second. This is used to resynchronize the clock of the receiver at any location.

3) IRIG-B

In 1952, the commanders of the U.S. guided missile test ranges formed the Inter-Range Instrumentation (IRIG) group as part of the Range Commanders Council (RCC) of the U.S. Army in order to share information about range instrumentation [11]. Today, the

Steering Committee and ten Technical Working Groups including the Telecommunications and Timing Group (TTG) control the IRIG time code standards. The IRIG standard was last updated in September, 2004 and is titled IRIG Serial Time Code Formats [12]. The most well-known and utilized is the IRIG-B time code from this standard. Despite multiple choices of IRIG code formats, IRIG-B is the mostly widely used in the power industry. In fact, more than ninety percent of substations use the IRIG-B format, which is typically better than ± 1 ms [13].

There exist two different forms of the IRIG-B time code: modulated and un-modulated. Modulated IRIG-B is transmitted on a carrier frequency sinusoid (similar to AM radio), and as a result, must be demodulated at the receiving end in order to interpret the time data. With advancements in phase locked loop (PLL) technology, modulated IRIG-B schemes can obtain accuracies within ± 10 μ s [14]. Installations with modulated IRIG-B must be isolated through some form of transformer in order to prevent ground loops and signal degradation. Un-modulated IRIG-B, or level shifted IRIG, is the alternative to the traditional modulated approach. This form of time code transmission uses digital level shifting to achieve data transfer. As no demodulation is necessary at the receiving end, time synchronization accuracy improves to ± 1 μ s [14]. The experimental setup described in this paper utilizes both modulated and un-modulated IRIG-B.

D. Precision Time Protocol

1) Definition

Precision Time Protocol, also known as PTP, is an IEEE standard that defines how to synchronize clocks across a network of distributed devices. IEEE 1588, PTP, was designed to fill a niche neither served by NTP or GPS. PTP provides very high accuracy, within 100ns time synchronization across a distributed network of devices [15]. This protocol is a great way to obtain precise timing where a GPS signal is not feasible or where full GPS deployment is not achievable [16].

PTP is fully entitled Precision Clock Synchronization Protocol for Networked Measurement and Control Systems. This protocol was designed for, but not limited to, LAN networks such as Ethernet, DeviceNet, and ControlNet. This protocol was originally released in 2002 defining many of the regulations and specifications, and later revised to its current version in 2008 and released as IEEE 1588v2. The original version of this standard defined ordinary and boundary clocks while version 2 defines transparent and hybrid clocks; all of these clock types will be addressed in this chapter [17].

PTP has many advantages to its predecessor of NTP. These advantages include: minimal installation and administration requirements, allowance for a wide spectrum of clock accuracy, and a low cost implementation of time synchronization that results in a high degree of accuracy and precision locally. For a wide area system such as the power system, the ultimate source of time should be tied to a wide area reference such as GPS.

2) Master-Slave Relationship

In its simplest form, PTP was intended to be an administration-free protocol. The clocks within a network will communicate with each other across a network media and establish multiple master-slave relationships. These master-slave relationships form what is known as a hierarchy of clocks [17]. The overall intent of PTP is that distributed I/O devices will manage the synchronization of clocks automatically, thus requiring little if any network administrator input. Not all devices within a network require the same level of time synchronization accuracy; as a result, PTP allows for the support of a wide spectrum of clock accuracies to support the needs of the end device or process [18].

For example, when dealing with protective relays, accuracy within a millisecond is acceptable due to mechanical component tolerances, but when dealing with PMUs, sub-microsecond accuracy is needed to meet the IEEE C37.118-2005 standard [19]. PTP can be configured to meet the needs of both of these applications. Additionally, PTP can use multiple sources of time as an ultimate time reference including, but not limited to, Global Positioning System (GPS), IRIG, Network Time Protocol (NTP), or another PTP clock.

As previously stated, PTP functions on a master-slave relationship to distribute precise time throughout the network of locally distributed devices (i.e. power plant/substation). Figure 18, below, depicts an example network consisting of devices with various types of clocks that will be discussed later in this chapter. The bubbles labeled *M* define a master and bubbles *S* define the slave in the master-slave pair.

Figure 18 System of Clocks

PTP defines that the slave in the master-slave relationship will tune its clock to that of the master. The overall best clock in the system is identified as the grandmaster clock to which all remaining clocks within the local network are ultimately synched to. It should be noted in this example that there are four different types of PTP clocks in any given network: boundary clock, ordinary clock, transparent clock, and hybrid clock. Each of these different clock types will be furthermore discussed in this chapter. The algorithms for how PTP determines a hierarchy of clocks can be seen in Appendix C.

3) *The Use of Tunable Clocks*

It has been previously discussed that PTP uses a set of requests and slave responses to determine the offset in time between master and slave. This section will provide an overview on how PTP clocks are adjusted with the offset time previously calculated. It should first be stated, that no two clocks will meter time at exactly the same rate. Due to crystal imperfections and varying frequencies, clocks will drift away from each other. This can be seen in Figure 19.

Figure 19 Master and Slave Open Loop

Protocols such as GPS use a traditional form of correcting this issue, by synchronizing the slave clock to that of master clock periodically. This only brings the two clocks into synchronization for an instant of time, and they will again begin to drift away from each other until the next synchronization[20]. This phenomenon is depicted in Figure 20.

Figure 20 Periodic Re-sync

PTP, on the other hand, does not perform time adjustments only at synchronization points. By definition in the standard, all devices which meet IEEE 1588 shall contain a tunable crystal so that it can be adjusted to the metered rate of that of its master [15]. With the implementation of a tunable clock, the slave clock can be adjusted for both value and frequency which allows for the clock to meter time and the same rate as the master clock. This explanation can be seen in Figure 21. At the first adjustment, the frequency of the slave clock is adjusted, while at the second adjustment point, the value of the slave clock is adjusted to that of the master. As a result, both clocks now meter time at the same rate.

Figure 21 Tunable Crystal

4) Clock Type

The PTP network architecture is built upon multiple types of clocks [7]. These clocks include boundary, transparent, hybrid, and ordinary clocks. Each of these types of clocks is embedded in various network PTP devices and function a little differently under IEEE 1588.

a) Ordinary Clock

The most basic type of clock used in PTP networks is the ordinary clock. The ordinary clock contains one PTP port where it will either serve as a slave or the grandmaster based on the best master clock algorithm. Examples of ordinary clocks are GPS receivers, logic controllers, and computers. Ordinary clocks are usually found as the extreme nodes of any network [16].

b) Boundary Clock

The second type of clock that will be addressed in this paper is that of the boundary clock. Boundary clocks are typically employed in network switches and routers. These clocks act as interfaces or boundaries between network segments as the name suggests. By definition a boundary clock consists of multiple PTP ports. Typically one port will serve as a slave, while the remaining ports, masters. PTP messages sent from the upstream master will be collected by the slave PTP port in the boundary clock. New PTP messages will be disseminated by each master port in the boundary clock [16].

Due to the segmented design of the boundary clock, it is not necessary to compensate for communication delays as time is synchronized from one clock to the next. As a boundary clock is no more than a cascaded clock-to-clock time synchronization technique, ordinary, or end node clocks should not be placed more than a few cascaded network hops from the grandmaster [16]. This means that boundary clocks are not a good choice to implement in long linear topologies.

c) Transparent Clock

In order to address linear network topologies, transparent clocks were introduced. Transparent clocks operate by compensating for propagation time delays through the network. As a result, the transparent clocks appear as a non-existent interface between master and slave clocks. Transparent clocks typically appear in managed network switches as well as motion devices where they will typically be combined with an ordinary clock to form a hybrid clock. An example of how transparent clocks operate can be seen below in Figure 22.

Figure 22 Transparent Clock Operation

A transparent clock operates by measuring the residence time or transit time between when a message is received and then transmitted. Transparent clocks contain a free running clock which is used to record timestamp information as PTP event messages enter or leave the switch. As the PTP message first enters the ingress port, a timestamp is recorded. Also when the PTP message leaves the egress port, another timestamp is recorded. These two timestamps are used to calculate the residence time. This residence time is then added to the correction field of the PTP packet [15]. This correction field is used by downstream switches to properly compensate for switch latency.

E. Common Industrial Protocol

The Common Industrial Protocol (CIP) is an industrial protocol suite that contains message and service instructions for automation applications pertaining to control, safety, synchronization, and motion [21]. This protocol is currently managed by the Open DeviceNet Vendors Association (ODVA) and is currently an open development network supported by hundreds of industrial vendors [22]. CIP allows for these applications to be implemented on enterprise-level Ethernet networks. Benefits of implementing CIP networks include: seamless integration of I/O control and data collection, information flow across multiple networks, and implementable multilayer networks without the need for the implementation of network bridges [23].

CIP can be defined as an object-oriented connection based protocol that supports both explicit and implicit messaging. Explicit message connections provide generic,

multipurpose communications paths between devices. These messages provide typical request/response type network communication [21]. Explicit messages are used by CIP for configurations, monitoring, and troubleshooting. Implicit messages, also known as I/O connections, provide special purpose communications paths between producing and consuming agents within a network. This I/O data is often exchanged cyclically or at a requested packet interval (RPI) [21].

The principle behind how EtherNet/IP synchronizes time across a network of distributed I/O can be attributed to the CIP Sync Object [20]. This object provides an interface to CIP Sync that allows devices such as logic controllers to access the synchronization mechanism. CIP Sync defines an offset clock model that addresses the requirements for various control applications. This model is necessary as PTP defines a mechanism for distributing and synchronizing time, but fails to define a mechanism for compensating for step changes in time that may occur at the grandmaster source [8]. The CIP Sync Object is further described in Appendix C of this document.

F. Quality of Service (QoS)

CIP defines a quality of service (QoS) that prioritizes the order in which network traffic is processed through network switches and routers. QoS helps to minimize network delay. QoS is important as PTP are time critical and need to be transmitted with minimal delay. An example of the implementation of QoS is provided in the figure below in Figure 23.

Figure 23 QoS Example

Figure 23 depicts a power systems application when QoS would be useful. QoS determines a hierarchy of packets based upon the packet's differential services code point (DSCP) field on the IP header [20, 24]. The higher the value in the DSCP field, the higher the priority of the packet. In the example above, the system contains four types of messages: PTP packets, PMU packets, IED packets, and various distributed I/O. If each of these packets reached the managed switch at the same time, according to the DSCP hierarchy, PTP packets would be processed first followed by PMU, IED, and distributed I/O packets.

The main concept of QoS is simple, whenever possible, process the highest priority data first. As a result, this will ensure time critical data is delivered in a timely manner. In the case of traditional power systems applications, the time synchronization algorithm packets must be given highest priority in order to preserve the quality of the network system time.

Based on system conditions, the QoS can be dynamically adjusted. For example if there was a fault on the system, timing packets may not have the highest priority, so one might re-define the QoS to make IED information have the highest priority.

G. Conclusion

In conclusion, time synchronization of measurements provides more meaningful information to control processes. Various forms of time synchronization are currently utilized today each with their own accuracies, advantages and disadvantages. Examples of time synchronization protocols are NTP, GPS, and PTP. The latter two protocols will be used for the experimentation presented in this paper. PTP can be implemented over a wide area system, such as the power grid with GPS serving as the grandmaster reference to time.

CHAPTER 6 EXPERIMENTAL SETUP AND COMPONENET JUSTIFICATION

A. Introduction

This chapter will discuss the components chosen for the experimental setup for the tests carried out in this document. In order to implement this system, a digital computing platform must be chosen. The two most practical solutions are a microcontroller based embedded system, or an industrial controls system consisting primarily of a Programmable Automation Controller (PAC). Each approach has its advantages and disadvantages. It should be reminded that the end result of the proposed system is to develop a way to robustly measure the torque angle of a synchronous machine. This measurement will then be used to implement a more efficient form of control for voltage excitation systems as well as power system stabilizers. Before developing an electrical system, the mechanical constraints of the system must first be defined.

B. Mechanical Components

Mechanical components that needed to be chosen for this application were an encoder, generator, and associated shaft couplings. These components will be discussed in this section.

1) Minimum Encoder Specifications

The encoder chosen for this application was a single turn absolute encoder. A single turn absolute encoder was chosen as it would provide position between 0 and $2^n - 1$ where n is the number of bits of the encoder. Due to the accuracy required in the shaft angle measurement, the encoder must adhere to the following set of specifications. A single turn absolute encoder must be used to measure the rotational position of the shaft. The encoder must output an absolute reading, as opposed to incremental, because the shaft angle must be determined instantaneously without reference to historical data.

With careful consideration of how the encoder is aligned to the shaft of the synchronous machine, the rotor position with respect to phase A will always be known. Since, at the point of synchronization, the field winding axis and magnetic axis of phase A are aligned, the rotor position with respect to A will be known if the zero of the encoder is aligned as

previously described. This is because the encoder is a mechanical device that is fixed to the shaft with a non-volatile zero reference.

To measure the angular frequency of the shaft (ω_m), the time difference between consecutive zero marker pulses will be used. In order to obtain the desired resolution of one tenth (0.1) of a degree, the encoder must have a resolution of 12 bits or higher. The resolution is found using Equation 6.1.

$$Angular\ Resolution = \frac{360}{2^{Bits}}$$

Equation 6.1

The encoder must be able to “sample and hold”, or “latch”, so the value will not change when read. Additionally, the time it takes for the encoder to respond to a latch pulse must less than 8 microseconds for this specific application. For purposes of this paper, the frequency response of the encoder will be defined as the maximum rate at which the encoder output changes. The minimum frequency response value is determined by the least significant bit, and the formula to calculate this value is below in Equation 6.2.

$$Frequency\ Response = \frac{ShaftRPM}{60} * 2^{Bits}$$

Equation 6.2

The 4-pole synchronous machine used for testing will rotate at 30 revolutions per second. Assuming a 12 bit encoder is used, the frequency response must exceed 122.8 kHz. A Sick|Stegmann encoder with a frequency response of 256 kHz was chosen to satisfy the constraints.

To reduce erroneous measurements caused by the switching of multiple bits at one time, the encoder should output in binary grey code. In order to communicate with the PAC, the encoder will output the latched grey code in Synchronous Serial Interface (SSI)

format when the latch command is sent. A SSI module will receive the information and transmit data over the Ethernet/IP network.

The encoder communicated over a Synchronous Serial Interface (SSI) back to the remote I/O chassis. SSI was chosen for this application as it provided a high speed data transfer from the encoder back to the PAC. In addition, SSI provided three cyclic redundancy (CRC) bits in order to check for transfer errors or bad data [32]. The encoder was triggered to be read by sinking 24 VDC to the encoder so it would latch the current value and return it to the PAC.

2) Mechanical Couplings

The last components that needed to be chosen were the couplings in order to couple the prime mover to the generating unit in the experimental setup. Three main types of couplings were considered for this application: solid metal dove-tail coupling, bellows coupling, and rubber spider coupling. A solid metal dove-tail coupling has the least amount of play in alignment of the two shafts of the machines. In addition, an improper alignment could mean that there would be too much torque placed on the coupling as well as the encoder which would damage the equipment. In other words, the margin of error and degrees of freedom for alignment with dove-tail couplings is very minimal. Bellows couplings provide more error for horizontal misalignment in that the bellows will flex to a certain extent absorbing the shock and torque due to the misalignment. The issue with this form of coupling is that it is often difficult to find two different size bores from coupling manufacturers. Finally, the spider coupling provides the most amount of play, and introduces the most amount of error into the system as a result. Spider couplings function by mounting two different couplings, one on each shaft, and properly torquing them until they will no longer rotate on the shaft axis. A piece of rubber or hard plastic is then inserted between the two couplings in order to interlock them. This piece of rubber absorbs all shock and torque pulses due to misalignment, thus directing them away from metering devices. For this project, spider couplings were chosen due to their cost as well as ease of mounting and alignment.

C. Embedded Solution vs. Automation Controller

Both approaches are capable of capturing the encoder output with sufficient accuracy and speed. However, a microcontroller would require significant software and hardware development. A PAC based system would be much quicker to implement, and would allow for easy expansion, however it should be noted that a PAC system excessive as this due to the PAC's potential processing power. For the purpose of experimentation and research, a PAC based system is a more appropriate choice. If this system were to be turned into an industrial product, where lengthy development time is expected, a microcontroller based system may be the best solution. The exception to this is a case where a power plant has an existing PAC based system to manage the generator, and wants to add shaft measurement functionality to the system. In the end, a PAC based system was used for implementation.

D. Programmable Logic Controller vs. Programmable Automation Controller for Power Applications

The first controllers implemented in power system applications were known as programmable logic controller (PLCs). PLCs are synchronous devices in nature in that they operate in the following manner: read inputs from modules, solve programmed logic, and update outputs based upon this logic [25]. PLCs were utilized by substation engineers as they were often programmed in ladder logic, a system very similar to hard wired relay logic. As a result, engineers had a small learning curve based on their previous experience of relay logic. The power industry saw multiple benefits of implementing PLCs including: hardware and software monitoring of system health, reduced wiring, fewer devices to implement, and the ability to change logic rapidly [25].

Prior to the 1990's, PLCs were only suitable for applications that were not time critical. For example, PLCs were used for automatic reclosers and carrier checkback schemes. PLCs also lacked the ability to interface with 125 VDC batteries typically used in substations [25]. If PLCs were to be used for protection and tripping applications, they would have to become faster machines that could communicate over a wide range of

protocols.

With growing demand for a device that would control complex processes in real-time, manufacturers began to develop programmable automation controllers (PACs). PLCs were well suited for complete plant automation, cell, and level processes, but lacked in managing complex algorithms to control large distributed processes [26]. There was a demand to run complex control algorithms in conjunction with the rest of the plant processes in real-time. PACs were developed to integrate these complex algorithms with a wide variety of inputs to control processes more quickly than PLCs [27]. PACs also have the distinct advantage of being able to interface with a wide variety of devices over industry established protocols.

PACs that are asynchronous in nature provide another distinct advantage to power system applications. Asynchronous machines utilize two different sets of memory, program and I/O. These two memory banks are in separate locations and can be written to independently. This means that I/O information can get collected separately from the executing instructions in program memory [27]. I/O memory is populated via the use of a backplane circuit and a Requested Packet Interval (RPI). Multiple data words can be simultaneously shared between modules connected to the backplane as well as written to I/O mapped memory locations [28]. This means that PACs can achieve true multitasking and can be interrupted by any real world input. This works well for protection and data acquisition applications as discrete inputs could interrupt a lower priority process so that action can be taken as quickly as possible.

The proposed solution consists of several standard Rockwell Automation components. It is believed that by using standard components, as compared to unique microcontrollers, that this system would ultimately be more attractive to various electric power utilities. The basic design of this experimental system can be broken down into two main components: local rack hardware and remote I/O hardware. Both of these components will be reviewed in this section.

1) Local Control Logix Rack

The Rockwell Automation Control Logix platform was chosen as the backbone for this project. This class of PAC allows for a wide variety of performance as well as flexibility as it is a rack based system. Simply speaking, the Control Logix platform consists of a chassis known as a rack, which contains slots numbered from zero to $n-1$, where n is the number of slots in the chassis. Modules, or PAC cards, are then placed into the slots to form a system. These cards perform a wide variety of operations ranging from discrete, analog, and communication type functions. A diagram of a Control Logix system is shown in Figure 24 below.

Figure 24 Control Logix Rack

Figure 25 shows rack consisting of 10 slots number zero through nine. The device on the left is the power supply which, as its name implies supplies power to the backplane. The modules are then seated into the backplane as shown in slot zero. The green printed circuit board shown in the diagram above is known as the backplane.

This PAC executes processes asynchronously. Most modern controllers run synchronously, meaning that they will scan the input table, execute a logic program, and then update the output table. Asynchronous processors on the other hand update their I/O table asynchronously from the program or logic scan. Communication between modules is achieved via the backplane itself. The modules are able to share information with each other across the backplane at a Requested Packet Interval or RPI. The RPI is essentially how often the I/O table is updated for that module. As the Control Logix system is asynchronous, it is capable of true multitasking as well as addressing various types of interrupts.

A diagram of the Control Logix rack and components that were used in this experimentation are depicted in Figure 25 below.

Figure 25 Experimental PAC Layout

The experimental setup used in this project consisted of the PAC components shown above. The list of components can be seen in the Table 1 below. For a complete description of each component and justification see Appendix D of this document.

Table 1 List of Components

Slot	Part Number	Description
0	1756-L73	Processor
1	1756-L73	Processor
2	1756-EN2T	EtherNet/IP Connection
3	1756HP-TIME	GPS Card
4	1756-IB16SOE	Sequence of Events Input Module
5	1756-OB16D	24VDC Sourcing Output Module
6	1756-OF6CI	6 point 4-20mA Isolated Output Module
7	1756-EN2T	EtherNet/IP Connection
8	1756-HIST2G	2GB solid state storage device

2) Remote I/O Rack

To this point, the automation and control system was defined from the local rack point of perspective. In an electrical generation station, not all of the various components are

physically located next to each other. This calls for two different approaches to solve this problem.

The first possible solution would be to place a rack based controller at every site and connect them all via a network such as Ethernet. There are a few major flaws with this implementation. Many processors are not meant to be operating in an environment where they are susceptible to large swings in temperature, or caustic, dirty environments. Most rack based PACs are kept in a control room that is environmentally controlled with climate control [29]. Placing multiple rack based processors in the same room may be very expensive to implement. The second downfall to this implementation is the wiring issue. With all of the PAC equipment located in one room, wire ways and conduit must be run so that field devices can be connected to the automation controller. Often long wire runs causes concerns with noise, cross talk, and signal degradation. All of these concerns are valid and can usually be fixed with shielded cable and repeater or amplification devices, but all updates come at a cost [28].

The second alternative solution to the central control cabinet issue would be to use distributed I/O. This method takes into account that all plants, when originally constructed, did not take into account the implementation of complex control and automation systems. Distributed I/O implements a solution with minimal wiring at a reasonable cost to the end user. The theory behind distributed I/O is simple: separate the I/O needed for the control system to where the equipment is located. This means that specific quantities of I/O can be distributed across a plant to meet the plant's needs. Wire lengths will be short as the I/O will be placed essentially next to the equipment to be monitored, which eliminates the need for amplifiers and reduces the risk of crosstalk [28]. The last problem is how to obtain the data collected by the remote I/O. Since most PACs already sit on communication networks, the quick answer is to sit the remote I/O on the same network.

For this application, the Rockwell Automation point I/O family of products was chosen. Point I/O can be thought of as I/O by the slice. Many of the modules manufactured for

the Control Logix Rack based family are also manufactured for point I/O. Point I/O is very similar to that of the Control Logix PAC as it is rack based. Each point I/O cluster contains a power supply, a network communications module, and various quantities of I/O modules. The power supply used for the point I/O cluster is a 24 VDC power supply. The communications module implemented for this application was EtherNet/IP, the same as the Control Logix rack. Figure 26 below depicts the point I/O utilized for this application.

Figure 26 Point I/O Implementation

Figure 26 above depicts the point I/O components utilized for this project. The item on the left is the power supply that is used to provide power to the system shown to the right. Moving from left to right, the first module is the EtherNet/IP module, sinking output module, and SSI module. The EtherNet/IP module is used to link the point I/O cluster to the rest of the network on which the main PAC resides. Power for all modules in the point I/O cluster is provided via the rack into which module is seated. The listing of all the modules used in the point I/O cluster can be seen in Table 2. For additional description of each of the elements listed in Table 2 see Appendix D of this document.

Table 2 Remote I/O Modules

Slot	Part Number	Description
0	1734-AENTR	EtherNet/IP adapter
1	1734-OV4E	24VDC Sinking Output
2	1734-SSI	Synchronous Serial Interface Module

3) Network Hardware

Up to this point, the control architecture of the rotor angle measurement process has been defined. The last major component of any control system is the network communication between devices. It was previously discussed that EtherNet/IP was chosen as the primary communications protocol for this system. It has also been discussed that the control for this project would be divided into two major sections: a local Control Logix rack, and a point I/O cluster that must be time synchronized via CIP Sync.

CIP Sync is a very powerful object developed by ODVA in order to synchronize a distributed network of industrial components via PTP. With the selection of certain network topologies and hardware network components, the time synchronization network can achieve and maintain a very high level of accuracy and precision of time. One of the most powerful components that can be implemented in any PTP network is the managed network switch. The switch that was chosen for the job was the Stratix managed Ethernet switch. An image of this switch is shown below in Figure 27.

Figure 27 Managed Switch

a) *Managed vs. Unmanaged Switches*

There are fundamental differences between traditional unmanaged switches and industrial managed switches. Managed switches allow the network administrator more control of network traffic flow by presenting the option for configurable ports on the switch. Unmanaged switches on the other hand usually come pre-fabricated with firmware and do not allow changes to their configurations [24]. Managed switches in short provide greater control of how data travels over the network and who has access to the network.

Managed switches are built upon protocols such as Simple Network Management Protocol (SNMP). SNMP is a protocol that provides for the exchange of information between network devices to determine the health of various network nodes [24]. Managed switches provide multiple services such as virtual local area networks (VLANs), redundancy, port mirroring, and most importantly Quality of Service (QoS) (discussed in Chapter 5).

Virtual LANs (VLANs) is a term that defines when a managed switch logically groups network devices together to isolate network traffic, even when these devices are located on the same physical switch. The segmentation of the two virtual networks provides packet isolation and thus reduces unnecessary network traffic [24].

Redundancy refers to the managed switch's ability to provide an alternate path of network communication should the primary path be compromised. Via the use of Spanning Tree Protocol (STP), redundant paths are provided, while preventing loops created by multiple active links between network devices [24]. STP allows for only one active path between two network devices at a time thus preventing loops. Protection schemes, for example, usually exhibit at least one critical link that cannot be compromised in order for network traffic to flow correctly [30]. The redundancy feature of managed switches will aid in the dependability of any packet switched network.

Port mirroring is a feature of managed switches that allows for troubleshooting network problems without taking down the network. The port mirroring feature, as its name implies, examines the data on one network port and replicates the data on another port of the network switch [24]. With the use of a network analyzer such as Wireshark, an analyst can troubleshoot the network without taking nodes out of service.

When it comes to time synchronization, managed switches provide an additional benefit to make sure that precise timing is maintained throughout the network known as Quality of Service (QoS). QoS prioritizes the order in which network traffic is processed at the switch in order to minimize network delay [31]. QoS is of the utmost importance in time synchronized packet switched networks as PTP messages must be transmitted with minimal delay.

Each of these features of managed switches was the main justification for utilizing a managed Ethernet switch in this application. The troubleshooting features would come in very handily during debugging of the system during initial testing. The dynamic QoS that could be provided was a key feature in how the network architecture could maintain precise timing throughout the system.

A final important feature of the managed switch is cut through. Cut through is a feature that starts delivering a packet out the egress port of a switch as soon as it is processed through the ingress port. Normally, a packet is fully delivered in the ingress port, stored in a first in first out (FIFO), and then the packet is sent out the egress port. Figure 28 below shows a visual definition of cut through versus normal switch packet operation. The top diagram shows how a normal unmanaged switch functions. When the TCP/IP packet enters the ingress port, the whole packet is then received, stored in a FIFO, and then sent out the egress port. The bottom diagram shows the cut through feature of the Stratix managed switch. The TCP/IP packet enters the ingress port and is transferred out the egress port, bit by bit to reduce residence time.

Figure 28 Cut through Definition

Since time synchronization and accuracy were a large concern for this project, measures were taken to eliminate network latency as much as possible. The alternative to reduce network latency is design a network architecture that reduces network latency.

E. Network Topologies and Architectures

When designing any automation and control system, network architecture must be taken into account to make sure network latency is not an issue. There exist many different network architectures used in automation today. The major network topologies used in many systems today include, star, linear, and device level ring. Each of these topologies has their own advantages and disadvantages depending on the overall goal of the network and process. Figure 29 below shows various network topologies practically used today.

Figure 29 Network Topologies

1) Linear Topology

In the linear topology, every device is directly connected to its neighbor as shown in the Figure 29 above. This topology allows for a simple implementation with minimal wiring as one device is directly connected to the next. There are two main downfalls with this topology for this application. The first downfall is that if a segment is broken, the network becomes segmented and there is no data path between the segmented parts of the network. The second downfall to the linear topology is that every message must physically be routed through all other devices until the packet reaches its destination. This introduces variable network delays which, in precise timing applications, cannot be easily compensated for or predicted.

2) Device Level Ring

In order to provide a more redundant system, compared to the linear network, the device level ring was introduced. The device level ring, as its name implies, forms a ring shape by taking the linear topology and closing in on itself forming a circle or ring of devices. This topology is more robust to network faults as it will take at least two node link failures to segment the network.

Although this network is more robust, it still has the downfall that all packets must pass through other devices until they reach their destination. The only advantage is now the packets have two different paths that they can take to reach their intended recipients. With regard to network latency, this topology is not the best choice as packets must be routed through more than one device to reach their destination. Each one of the hops between devices takes time and adds to the overall latency of the system.

3) Star Topology

In order to address network latency the star topology was developed. This topology consists of all network devices, i.e. nodes, placed to the outside of the topology and all connected to a central hub or distribution point to form a star. The center node of the star is often a layer two or three network device such as a managed switch or router. In the case of this setup, the Stratix 8000 managed switch is the central node for this topology. The topology has the advantage that it reduces network latency. By having all data collecting nodes to the extremities of the topology network traffic is directed to the central node and back out to the intended packet recipient.

The managed switch in the middle, by being a transparent clock adjusts for the residency time of the packet and puts a correction time field on the header of every data packet so that this delay is accounted for. The downfall to this form of network is the question of what happens to data that is received at the switch at the same time. The answer to this question is the QoS defined by the network administrator will give priority to certain packet types over others so that packets received at the switch at the same time are

queued by priority and sent out the egress port of the switch. Another downfall to this network topology is that if a link is lost between a node and the central hub, the network becomes segmented. This can be corrected by placing a second, or redundant, connection between the end node device and the managed switch for industrial applications. This setup is known as the redundant star topology.

F. Chosen Network Topology

For this application, the star topology was implemented. This network topology was chosen as it provides the best resistance to network latency. Also this connection is the most commonly used amongst all the network topologies for high speed motion and control applications. Figure 30 below depicts the network diagram and architecture utilized for this project.

Figure 30 Network Topology

Figure 30 above shows the network topology for the systems used for experimentation in this project. As previously mentioned, the network is a star topology based network with a managed switch bridging the network. The star contains five different nodes, three permanent and two non-permanent, that attach to the managed switch. These five nodes are detailed in the Table 3 below.

Table 3 Network Topology

Node	IP Address	Description
PAC	192.168.1.101	Local Control Logix Rack
Point/IO	192.168.1.106	Point I/O Cluster (SSI)
Remote Terminal	192.168.1.200	Remote programming computer (NP)
Human Machine Interface	192.168.1.200	Visual display screen (NP)
Database Server	192.168.1.150	Storage device for data

The local PAC rack contains multiple devices which are IP addressable. These devices and their IP addresses are listed below in the Table 4.

Table 4 Local PAC Addresses

Node	IP Address	Description
EN2T (Primary)	192.168.1.101	Primary network connection to PAC
EN2T (Redundant Backup)	192.168.1.102	Redundant network connection to PAC
Time Module	192.168.1.55	PTP address port of ordinary clock
Historian Output Port	192.168.1.66	Correlation port between module and server

In the network architecture, the remote terminal is a computer that is used to connect to the processor through the managed switch. This is typically a laptop computer equipped with software to program the PAC. The IP address of this computer is set to a static IP address to match the network, in this case 192.168.1.200.

The human machine interface or HMI is a visual display of data within the processor. A HMI is typically an industrial computer or touch screen that displays certain tags, or

variables of information that are currently in the processor. For example, the HMI for this project depicted the current time, the time stamp given to a data point, the encoder count for the data point taken, and the corresponding mechanical angular position of the synchronous machine.

The server mentioned in Table 3 serves the purpose to correlate with the historian module in the local rack of the PAC. The server is just a normal desktop computer that is equipped with a SQL database that will correlate with the with the historian module. At every correlation point the server will request the following information from the historian module: the angular position of the rotor and the corresponding UTC time tag. The SQL database also contains the same two fields. The first column is the UTC time tag and the second column of data is the mechanical angle of the machine. Table 5 below shows example output from the database.

Table 5 Database Sample Output

UTC Time Tag	Angular Position [deg]
11.08.2011.10.07.01.342145	23

The UTC time tag shown above is written in the following format as shown in the Table 6 below.

Table 6 Time Tag Definition

UTC Time Tag [aa.bb.cccc.dd.ee.ff.gggggg]	Field Description
aa	Month of year [00-12]
bb	Day of month [00-31]
cccc	Year [0000-9999]
dd	Hour of Day [00-23]
ee	Minute of Hour [00-59]
ff	Second of Minute [00-59]
gggggg	Microsecond of second [000000-999999]

This concludes the definition and scope of the automation and control equipment for the project. In summary, the automation and control equipment utilized for this project consisted of both a local control rack and a remote point I/O cluster connected via Ethernet in a star topology. These pieces of control equipment interface with remote PCs for programming, HMIs for displaying data, and servers for archiving large amounts of data in a SQL database.

As this is a high speed control application, precise timing is needed to ensure accuracy of measurements. Precise time will be kept by a local GPS clock and distributed throughout the network via the PTP protocol. Tunable clocks were investigated and chosen as a better solution over the GPS PPS as there will be less variance in how the clock meters time if they are tuned to one another rather than just brought into alignment at one instance in time.

Various switches and routers were investigated to route data packets and a managed switch was ultimately chosen for its ability to define QoS for every data packet and its ability to function as a transparent clock in the PTP architecture of this system. Ultimately, the star network topology was chosen as this application is a high speed motion application, and the star topology provides the least amount of network latency between all network nodes.

G. Software Algorithm

A software algorithm needed to be written in order for the PAC to gather encoder samples. This program was constructed in ladder logic and downloaded to the PAC processor. The basic outline of how the program worked is shown in Figure 31.

The program began and would wait for a valid time synchronization signal. In this case, the program would look for a valid pulse-per-second from the GPS source. When the PAC received a valid time synchronization signal, a valid sample period and start time would be entered into the program. The sample period would define how often the

encoder would be latched, and the start time would define when in Universal Time Coordinated (UTC) the encoder would begin to be sampled. If a sample time has not been reached, the PAC will wait until the next sample is ready to be taken.

When the sample is ready to be taken, the encoder is latched by sinking 24 VDC from the remote I/O module. The data received by the PAC is then checked for CRC bit errors that might have been incurred during data transfer. If the data is not valid, the data point is rejected and the PAC waits for the next valid data point. On the other hand, if there are no CRC errors, the data is time stamped with a UTC timestamp and stored in a local array buffer. The local database module will then retrieve this data word and save it to the SQL database. This process will occur until the last sample is taken, or the process is interrupted by the operator. The main reason that this algorithm of synchronized sampling works properly is that all devices chosen for this system are PTP compliant and time synchronized over PTP.

Figure 31 Software Algorithm

The program written for this application was based on the principle of synchronized actuation. Take for example a local controller with distributed I/O (i.e. Local Rack and Point I/O cluster) each with their own clock. If these clocks are synchronized, and meter time at the same rate, the controller can then send information, both time and data across the network to the distributed device. The I/O (encoder latch), can then actuate its output (latch) at its scheduled time. As both clocks are synchronized and meter time at the same rate, the resulting function at the distributed I/O device is known as a scheduled output or synchronized actuation. The only requirement for this principle to work is that the data and time of execution must reach the I/O device prior to the scheduled execution time.

Synchronized actuation is the fundamental principle of motion control with respect to servo drives.

H. Conclusions

This chapter has justified the components used for both the electrical and mechanical setups for this project. A microprocessor based solution as well as an automation controller solution was considered, and an automation controller solution based system was selected for ease of prototype implementation. After the PAC selection was made, a determination on whether to use a Programmable Logic Controller (PLC) or Programmable Automation Controller (PAC) was to be used. After determining that an asynchronous processor would work better for this process, a PAC was chosen for the project.

The automation hardware was divided into two categories, local and remote distributed I/O. The local I/O consisted of the PAC communicating to a variety of devices such as HMIs, databases, and remote terminals. The distributed I/O consisted of an SSI module to interface to the encoder as well as a 24 VDC module to latch the encoder value. A system consisting of both local and remote I/O because it represents a real world system where distributed I/O would be placed near equipment to be monitored and information would be brought back to a central location.

A single turn absolute encoder was chosen for this application as opposed to an incremental encoder, as absolute position must be known at all times. By aligning the zero marker of the encoder with the magnetic axis of phase A, absolute position will be known at all points in time. Synchronous Serial Interface (SSI) was chosen as a communications algorithm to interface back to the automation system. SSI allows for communications up to 2 MHz and also provides noise immunity over traditional parallel connections of encoders. Additionally, the SSI encoder chosen for this application as it provided eight error codes to represent bad data during data collection or transfer.

The software algorithm written for this process was based upon synchronized actuation, a fundamental principle of servo drive motion control. The program was written in ladder logic and utilized the time synchronized network to latch the encoder values at set intervals. This value was then stored in a buffer array and transferred the historian module in the background while other tasks were executing.

This concludes the chapter on experimental setup and component justification. The following chapters will discuss the experimental setup and experimental results obtained.

CHAPTER 7 EXPERIMENTAL TRIALS AND ANALYSIS

A. Introduction

This chapter focuses on describing the various tests conducted on the machine model setup for this experimental setup. Three main tests were conducted, each with their own various trials. These tests include: infinite bus connection, constant frequency source connection, and dynamometer loading test. Each test will be discussed in detail explaining the procedure and showing the results in graphical form with analysis.

B. Infinite Bus Connection

The first setup tested was the infinite bus connection. In this setup, the synchronous machine was run as a generating unit driven from a fractional horsepower permanent magnet DC machine. This connection can be seen in Figure 32, below.

Figure 32 Infinite Bus Connection

In this schematic, the blue DC bus represents a lab bench DC power supply in order to feed the DC armature with approximately 90 VDC to drive the machine at 1800 RPM (synchronous speed of the synchronous motor). The green AC bus represents the

equivalent power system to which the small synchronous generator was connected. Since the four pole synchronous machine's capacity of 0.25 HP is much less than the capacity of the equivalent system (eastern power interconnection), the bus looks "infinite" to the synchronous machine.

1) Synchroscope

Before any synchronous generator can be connected to the grid, it must be properly electrically synchronized to the system to which it connects. The previous statement holds true for all generators. If the machine is not properly synchronized to the grid, it runs the risk of damaging the machine and associate transformer unit.

In order to properly synchronize a machine to the grid, the following criteria must be met:

1. Same electrical frequency
2. Same voltage magnitude
3. Same electrical sequence (positive or negative)
4. Same electrical phase angle

At the point of electrical synchronization, there is no power transferred to or from the electrical bus at which the machine is connected. In order to test for proper synchronization, a synchronization circuit was developed and implemented between the generator terminals and the electrical bus. The diagram of the synchronization circuit can be seen in Figure 33.

Figure 33 Synchronization Circuit

In Figure 33, connections J1-1 and J1-2 are banana plug outlets connected to the DC power supply. Contactor K2 represents the power switch on the DC power supply breaking both legs of the DC connection. The dashed line between the two machines represents a mechanical coupling between the two machines, and the solid line connected to the bottom of the synchronous generator is the field winding connection. J2-1, J2-2, and J2-3 represent the electrical phase connections to the three phase infinite bus to which the generator was connected. Contactor K1 was the synchronizing contactor. When the four criteria were met in order to synchronize a machine, this contactor was closed, thus completing the circuit between the generator and the electrical bus.

In order to check for proper electrical phase and sequence, the six incandescent pilot lights shown in Figure 34 were used. In the connection shown above the six lights would cycle between dim and bright luminosities at the same time.

Figure 34 Control Schematic

Figure 34 depicts the control diagram for the synchroscope developed for this investigation. SS1 represents the normally open pushbutton used to control the synchronization contactor. After the machine is properly synchronized to the system, benchmark testing can be performed.

2) Infinite Bus No Time Synchronization

In order to test the theory that time synchronization was an important application for this process, a simple test was conducted removing all PTP configurations from the setup. The principle behind the test was simple; if the synchronous generator was tied to an infinite bus, and the bus was at an electrical frequency of 60 Hz, the encoder value read every 100 milliseconds would be the same value. This is due to the fact that the four pole synchronous generator would make three full revolutions every 100 milliseconds running at 1800 RPM or 60 Hertz. Figure 35 depicts the results of the test where no PTP was implemented in the distributed system.

Figure 35 Non-Time Synchronized System

It can be seen from Figure 35, that there appears to be no pattern to the data when the data is collected at 100 millisecond intervals, but in a non-time synchronized distributed system. Further analysis of the data shown, the angle of the machine can be fit to the polynomial equation shown in Figure 36. The results from this trial show that time synchronization must be implemented in this system to have meaning in a wide area system.

3) Infinite Bus Time Synchronized via PTP

To further explore the system, Precision Time Protocol was then enabled in the distributed system. The master clock chosen for application was the GPS time module in the local rack of the controller. As previously described, this time would be distributed over EtherNet/IP to the remaining distributed devices on the network. Figure 36 depicts the results first obtained from these trials.

Figure 36 Time Synchronized Samples

Figure 36 shows the results from a time synchronized trial conducted with the infinite bus connection previously described. At first glance, one would assume that the system developed to measure rotor angle, failed at its main intention. This statement would be true if the data represented in Figure 36 depicted a truly 60 Hz power system. The actual power system does not operate at exactly 60 Hz at every instance in time, but averages 60 Hz over the long run. The data depicted in Figure 36 was actually taken during a time when the system varied between 60.02, 60.01, 60.00, and 59.99 Hz respectively.

Multiple trials were conducted in order to attempt to find a test case where the frequency would remain relatively constant for a test trial. In order to perform this task the following setup as seen in Figure 37 was constructed.

Figure 37 Test Setup Frequency Test

Figure 37 shows the test setup used in order to measure the frequency of the system in order to obtain a case where the electrical frequency remained relatively constant. Since this test was conducted in an academic lab environment, the electrical system consisted of the building in which the lab was housed. This meant that cooling, overhauling loads such as elevators, lighting, as well as a variety of other events would affect the frequency of the bus to which the generator was connected.

The frequency meters depicted in Figure 37 are representative of Fluke 45 multimeters which were calibrated to $\pm 0.05\%$ in their current mode [33]. The oscilloscope shown in the figure represents a LeCroy 1 GHz digital oscilloscope. The oscilloscope was configured to measure the electrical frequency between phases A and B in the current setup.

The testing for this setup was performed during late hours of the evening or during early hours of the morning, so that the building environment would have as little effect as possible on the outcome of this trial. The results from a successful test are shown in Table 7 and Figure 38 respectively.

Table 7 Frequency Test Results

Metering Device	Frequency [Hz]
Freq. Meter 1	60.01
Freq. Meter 2	60.01
Freq. Meter 3	60.02
Scope	60.0157

Figure 38 Time Synchronized Encoder Trial Frequency Test

It can be seen from Figure 38 that the linear regression line is a very good fit, as its Pearson coefficient is approximately 1.0 or perfect linear fit. The slope of the line is approximately 3.31 encoder counts per sample taken. The frequency meters as well as oscilloscope measurements remained close to constant throughout the trial. It can be seen from Table 7 that the metering devices provided similar measurements throughout the test. In order to see if the system is properly calibrated, mathematical formula needed to be implemented in order to validate that collection process. These calculations can be seen in the following equations. It will then be determined if the above described system will function properly in the dynamic power system environment.

$$\frac{3.3083 \text{ counts}}{\text{sample}} * \frac{1 \text{ sample}}{100 \times 10^{-3} \text{ s}} * \frac{360 \text{ degrees}}{4096 \text{ counts}} * \frac{\pi \text{ radians}}{180 \text{ degrees}} = 0.0507 \frac{\text{mech radians}}{\text{sec}}$$

Equation 7.1

Equation 7.1 uses simple conversion factors in that the program took one sample every 100 milliseconds as well as the basic specifications of the encoder to determine that during this test the encoder was reading 0.0507 mechanical radians per second. This then needed to be converted to electrical radians and eventually deviation from electrical frequency. This calculation is shown in equation 7.2 and 7.3 respectively.

$$\frac{0.0507 \text{ mech radians}}{\text{sec}} * \frac{2 \text{ electrical radians}}{1 \text{ mech radian}} = 0.10149 \frac{\text{electrical radians}}{\text{s}}$$

Equation 7.2

$$0.10149 \frac{\text{electrical radians}}{\text{sec}} * \frac{1 \text{ Hz}}{2\pi \text{ radians}} = 0.01615 \text{ Hz}$$

Equation 7.3

It can be seen from equations 7.2 and 7.3 that the deviation from nominal is approximately 0.10149 electrical radians or 0.01615 Hz respectively. Taking the difference between the oscilloscope's measured frequency and the encoder measured frequency yield the result shown in equation 7.4.

$$\Delta f = f_{\text{encoder}} - f_{\text{oscilloscope}}$$

$$\Delta f = 60.01615 - 60.0157 = 0.00045 \text{ Hz}$$

Equation 7.4

The difference between the measured value from the oscilloscope and the experimentally measured encoder value was 0.00045 Hz. This corresponds to a percentage error of 7.49 E -4, which is well within the acceptable error range for this project as it is within the error tolerance of the mechanical components. This test has shown that the test setup will

function properly within a model power system. In order to test the accuracy of repeatability of measurements, a constant frequency source must be utilized. The data used to plot the above figures in this chapter can be found in Appendix E.

C. Constant Frequency Source

In order to further validate the accuracy of this device, a second setup was created by which the synchronous machine would be driven as a motor by a constant frequency source. By being able to control frequency, the accuracy of the encoder sampling process from latch to reported value could be investigated. Furthermore, providing a constant frequency source eliminates one more experimental degree of freedom from the testing scheme. The setup for this testing scheme can be seen in Figure 39 below.

Figure 39 Frequency Controlled Source Setup

As previously mentioned, Figure 39 depicts that the synchronous machine utilized in the setup would be driven as a motor rather than a generator. This meant that rated field current would need to be applied field windings and the speed of the machine would be controlled by the AC source frequency. The adjustable frequency source, which was much larger in capacity than the motor, looked like an infinite bus to the motor during the test. Based on the theory presented previously in this chapter, if the encoder was sampled every 100 milliseconds and the electrical frequency was precisely 60.00 Hz, the same encoder value would be latched on every sample interval. By taking into consideration how the encoder was aligned to the shaft of the machine (aligning it with the magnetic axis of phase A), the value read by neglecting mechanical losses of the machine should be the zero count.

Additionally, a phasor measurement unit was placed on the electrical bus between the source and the motor in order to evaluate the voltage of phase A as a vector. In theory, during normal positive sequence operation the phase angle of electrical phase A should be zero degrees as a balanced three phase set.

The results of this test are shown in Figure 40.

Figure 40 Frequency Controlled Source

It can be seen from Figure 40 that the encoder value remains constant except for two points on the graph over the 1000 samples taken. In addition, the angle of electrical phase A also remained the same over the time trial. The encoder value was not exactly zero, but approximately -0.24 degrees, which corresponds to three encoder counts. This is due to the fact that the synchronous machine has losses in its windings and friction in its bearings. The author feels that this test validates the sampling of the encoder.

D. Dynamometer Loading

In order to test the dynamics of the system, a dynamometer was attached to the synchronous machine. By changing the loading on the dynamometer, the synchronous motor would experience different levels of dynamics and the rotor angle could be measured. The results of the captured data were then compared against a MATLAB simulation where a seven variable state model of the machine was run. The parameters for the machine were obtained from the manufacturer. The results of the MATLAB simulation were exported to Excel where the rotor angle measurements could be plotted on the same graph.

Figure 41 Dynamometer Testing Setup

Figure 41 depicts the dynamometer setup for the tests conducted in this trial. The dynamometer applied a step change in torque to the synchronous machine to represent a change in load of the power system. Although the machine is being driven as a motor, the principle is the same. The loading on the dynamometer will be increased in steps of 10 N-m until the machine loses synchronism.

Figure 42 Dynamometer Testing 7.5 HP

Figure 42 depicts the load curves for a 7.5 HP synchronous machine used for testing in the first dynamometer trials. The y-axis represents the rotor angle in degrees and the x-axis represents sample number. For this test samples were taken every 10 milliseconds and a step change in torque was applied at the tenth sample. The results can be seen in the figure above. It should be noted, that when neglecting the resistance of the machine, the steady state stability limit is a rotor angle of -90 degrees for a motor. When this value is exceeded the machine is considered out-of-step. It can be seen from Figure 43 that the machine loses synchronization at 120 N-m and is definitely out-of-step when 130 N-m of torque is applied. Additionally, for this machine a plot of 40 N-m is shown in greater detail as the first few curves are closely bunched together. This plot can be seen in Figure 43. Additional plots and data can be seen in Appendix F of this document.

Figure 43 40 N-m Test 7.5 HP Machine

It can be seen from Figure 43, that the dynamics of the machine are not too visible. This is because the machine is so small, i.e. 7.5 HP, and the rotor possesses a small inertia. In order to attempt to test the dynamics of the machine, a 10 HP machine with different manufacturer's parameters were used. The results of these tests can be seen in Figure 44.

Figure 44 Dynamometer Loading 10 HP Machine

It can be seen from Figure 45, that the dynamics of this second machine tested are different than that of the first. For example, this machine loses synchronism when 120 N-m is applied as a step change at the shaft. Also since the inertia is larger on this machine as the shaft was larger in diameter as well as length, there are small oscillations seen when the step change is applied. This can be seen in more detail in Figure 46.

Figure 45 30 N-m Test of 10 HP Machine

In order to validate the test results depicted in Figures 42-45, a MATLAB program was written to simulate the seven state variable model of the synchronous machine. The parameters we entered by manufacturer's specification sheets and the results simulated and compared against the results shown above. For all stable cases, the maximum percentage error difference between simulated and actual measurement of rotor angle was 0.63%, an acceptable value of error for this application. The MATLAB program written for this application can be seen in Appendix G of this document.

E. Conclusions

This chapter has outlined the experimental tests and procedures conducted to validate the designed prototype for this process. Three different tests were conducted including: the infinite bus test, the constant frequency source test, and the dynamometer loading test. Each of these tests was conducted to validate a different portion of the experimental design. In all, the equipment under tests performed as expected and the experimental error was kept to a minimum throughout all tests conducted.

CHAPTER 8 CONCLUSIONS AND FUTURE WORK

A. Overall Conclusions

The torque angle is a key parameter to assess the stability and to provide control of synchronous machines. Many previous attempts have been made to measure the torque angle of synchronous machines, but limitations in technology and timing issues have prevented prior researchers from obtaining reasonable results. Results from previous studies could be used to marginally improve the feedback control of machines. Many of these studies performed on synchronous machines were conducted during the 1950s and 1960s. Research has shown that relatively little advancement in this field has evolved over the last 50 years. The two main approaches discussed in this document were by V. A. Kinitzky, D.G. Lewis, and A. Austin.

The main setback to the solution provided by Kinitzky was the required frequent calibration that needed to be provided to his equipment setup. Additionally, he lacked a way to measure rotor angle under transient conditions. Lewis and Austin, on the other hand, lacked a tunable precise timing device. Finally, both solutions lacked some form of data storage.

The approach used by the author for this application to measure the rotor angle of synchronous machines leverages current technologies developed for the industrial marketplace. If the rotor position is measured at (time-tagged) equal time intervals, the information collected yields data that allows researchers to perform post mortem analysis on transient events that will affect the performance of the synchronous machine or to develop applications for control.

When discussing machine models for synchronous alternators, three main models are generally used. These models include a source behind a reactance, a source behind a sub-transient reactance, and a seven state variable model. Each of these models has their pros and cons. As more intricacy is added to the model, a more accurate result will be generated, but at the cost of processing time.

This document has explored three different models for synchronous machines: steady state model cylindrical machine, steady state model salient pole machine, and seven variable state space models. As the model increases in complexity, it is a better representation of the machine's performance under various conditions. The downside to the increase in accuracy is also an increase in computing time in order to find a solution to the model.

It was previously stated that the torque angle of the synchronous machine is a key parameter for generation stability as well as power output. This chapter will explore the various control systems that utilize the torque angle of the synchronous machine either directly or indirectly.

When referring to the power network stability, one must look at the control of active and reactive power. These two network parameters are independent of each other, and as such, are governed by separate control algorithms. The torque angle of the synchronous machine is used as a feedback or input parameter in many different power system control applications.

As described in the prior sections, there are a variety of plant components that can utilize the rotor angle measurement either directly or indirectly. These components include speed governors, exciter systems, and power system stabilizers. The proposed system will allow for benefits in both local and wide area control for power systems. Locally, improvements can be realized in speed governor control, exciter systems, local power system stabilizers, and local inter-machine oscillations. Wide area applications can also be realized for center of angles calculations, angle coherency, and coordinated control of power system stabilizers.

In today's society, industry is relying more heavily on time synchronization for the administration of their everyday processes. Almost everything that we do in our lives relies on a clock or reference to time and as such it is important that our clocks are

synchronized. Examples of time synchronization in industry today can be seen in input time stamping, sequence of events recording, and motion control.

With today's technology to synchronize clocks, it is now possible to send both the data and a meaningful timestamp back to the controller, providing more information. Currently, many industries utilize time synchronization to coordinate sequence of events measurements, time stamped data logging, as well as coordination with GPS time. Since the clocks of the controller and the I/O device are time synchronized, scheduled outputs, and synchronized actuation can be achieved. Common ways to synchronize systems are Network Time Protocol, Global Positioning System, and Precision Time Protocol.

Time synchronization of measurements provides more meaningful information to control processes. Various forms of time synchronization are currently utilized today each with their own accuracies, advantages and disadvantages. Examples of time synchronization protocols are NTP, GPS, and PTP. The latter two protocols will be used for the experimentation presented in this paper. PTP can be implemented over a wide area system, such as the power grid with GPS serving as the grandmaster reference to time.

In order to implement this system, a digital computing platform must be chosen. The two most practical solutions are a microcontroller based embedded system, or an industrial controls system consisting primarily of a Programmable Automation Controller (PAC). Each approach has its advantages and disadvantages. It should be reminded that the end result of the proposed system is to develop a way to robustly measure the torque angle of a synchronous alternator. This measurement will then be used to implement a more efficient form of control for voltage excitation systems as well as power system stabilizers.

A microprocessor based solution as well as an automation controller solution was considered, and an automation controller solution based system was selected for ease of prototype implementation. After the PAC selection was made, a determination on whether to use a Programmable Logic Controller (PLC) or Programmable Automation

Controller (PAC) was to be used. After determining that an asynchronous processor would work better for this process, a PAC was chosen for the project.

The automation hardware was divided into two categories, local and remote distributed I/O. The local I/O consisted of the PAC communicating to a variety of devices such as HMIs, databases, and remote terminals. The distributed I/O consisted of an SSI module to interface to the encoder as well as a 24 VDC module to latch the encoder value. A system consisting of both local and remote I/O because it represents a real world system where distributed I/O would be placed near equipment to be monitored and information would be brought back to a central location.

A single turn absolute encoder was chosen for this application as opposed to an incremental encoder, as absolute position must be known at all times. By aligning the zero marker of the encoder with the magnetic axis of phase A, absolute position will be known at all points in time. Synchronous Serial Interface (SSI) was chosen as a communications algorithm to interface back to the automation system. SSI allows for communications up to 2 MHz and also provides noise immunity over traditional parallel connections of encoders. Additionally, the SSI encoder chosen for this application as it provided eight error codes to represent bad data during data collection or transfer.

The software algorithm written for this process was based upon synchronized actuation, a fundamental principle of servo drive motion control. The program was written in ladder logic and utilized the time synchronized network to latch the encoder values at set intervals. This value was then stored in a buffer array and transferred the historian module in the background while other tasks were executing.

Three main tests were conducted, each with their own various trials. These tests include: infinite bus connection, constant frequency source connection, and dynamometer loading test. Each test will be discussed in detail explaining the procedure and showing the results in graphical form with analysis.

Each of these tests was conducted to validate a different portion of the experimental design. In all, the equipment under tests performed as expected and the experimental error was kept to a minimum throughout all tests conducted.

B. Future Work

The future work for this project can be categorized into three phases. These phases include: integration with PMUs at the electrical bus, producing a valid C37.118 word, and implementing this system on a paralleled machine setup.

The first task for future work on this project would be to implement a PMU at the electrical bus of the system. By placing a PMU at the electrical bus, the internal angle of the synchronous machine measured by the encoder can be compared to the electrical bus angle at the terminals of the machine. In a wide area system, all PMUs are referenced to one location or machine in the system. Since this reference is most likely not at the local bus, a simple subtraction must be performed between the local PMU and encoder value to reference the internal angle of the machine to that of the reference. In order for this to work correctly, the encoder must be sampled at the same time interval as which the PMU produces and electrical phasor value.

The second task for future work consists of producing a valid IEEE C37.118 word. IEEE C37.118 is the governing standard for phasor measurement units. This standard specifies minimum mechanical and electrical specifications in order for a device to be classified as a PMU. Additionally, C37.118 specifies a format by which data must be packaged, time-tagged, and sent throughout the network. By making this proposed system meet IEEE C37.118 compliance for communications, the information about internal machine angle can be shared with command centers across the network, and the information may be useful for wide area control schemes as discussed in chapter four of this document.

The final portion of future work identified in this document will be to implement the test setup in a real system. This means that modifications to the mechanical setup would need

to be made in order to incorporate this system into a generation station. The main component that would need to be replaced would be the encoder. For the small laboratory tests, an axial mount, or hollow shaft encoder could have been selected as the shaft diameter is small, and a stub shaft was available for mounting. The optimal solution to this problem would be to place a linear tape encoder on the shaft of the turbine-generating unit. As a result, some form of SSI sensor would need to be affixed to the shaft as well in order to read the values. The optimal solution for this issue would be to utilize an optical sensor. There may be a location within the Mexican system that this system may be able to be implemented.

WORKS CITED

- [1] P. C. Krause and O. Wasynczuk, *Electromechanical motion devices*: McGraw-Hill, 1989.
- [2] V. A. Kinitzky, "Measurement of Rotor Displacement Angle on Synchronous Machines," *Power Apparatus and Systems, Part III. Transactions of the American Institute of Electrical Engineers*, vol. 77, pp. 349-352, 1958.
- [3] V. Biron, and Springer, *Starkstrom-Messtechnik*, 1933.
- [4] D. G. Lewis and W. E. Austin, "Measurement of rotor load angle of synchronous machines," *Electronics Letters*, vol. 5, pp. 113-114, 1969.
- [5] A. E. Fitzgerald, C. Kingsley, and S. D. Umans, *Electric machinery*: McGraw-Hill, 2002.
- [6] P. Kundar, *Power system stability and control*: McGraw-Hill Education (India) Pvt Ltd, 1994.
- [7] Basler. *Power System Stabilizers*. Available:
<http://www.basler.com/html/html/powersysstabilizer.htm>
- [8] R. A. Inc., "Precision Time Protocol Over EtherNet/IP," in *Motion Over EtherNet/IP Learning Series*, R. A. Inc., Ed., ed. Milwaukee, WI: Rockwell Automation Inc., 2011.
- [9] D. L. Mills, "Internet time synchronization: the network time protocol," *Communications, IEEE Transactions on*, vol. 39, pp. 1482-1493, 1991.
- [10] L. M. N. M. A. Lombardi, A. N. Novick, V. S. Zhang, "Time and Frequency Measurements Using the Global Positioning System," *Cal. Lab. Int. J. Metrology*, pp. 26-33, 2001.
- [11] B. W. Pike, "IRIG, Inter-Range Instrumentation Group---History, Functions and Status, 1959," *Space Electronics and Telemetry, IRE Transactions on*, vol. SET-6, pp. 59-61, 1960.
- [12] "IRIG Serial Time Code Formats," in *IRIG Standard 200-04* ed. U.S. Army White Sands Missile Range, New Mexico: Range Commanders Council, Sept. 2004.
- [13] B. Dickerson, "Time in the Power Industry: How and Why We Use It," *Arbiter Systems, Inc.*
- [14] I. P. PSRC, "IRIG-B Time Code Connection Requirements."
- [15] "IEC/IEEE Precision Clock Synchronization Protocol for Networked Measurement and Control Systems (Adoption of IEEE Std 1588-2008)," *IEC 61588 First edition 2004-09; IEEE 1588*, pp. 0_1-156, 2004.
- [16] R. Automation, "CIP Sync and Precision Time Protocol," in *Integration Motion on EtherNet/IP*, R. Automation, Ed., ed. Milwaukee, Wisconsin: Rockwell Automation, 2011.
- [17] "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," *IEEE Std 1588-2008 (Revision of IEEE Std 1588-2002)*, pp. c1-269, 2008.
- [18] Cisco, "CIP Sync Sequence of Events," in *Converged Plantwide Ethernet (CPwE) Design and Implementation Guide*, Cisco, Ed., ed, 2011, p. 72.
- [19] "IEEE Standard for Synchrophasors for Power Systems," vol. IEEE Std C37.118-2005 (Revision of IEEE Std 1344-1995), ed, 2005.
- [20] R. A. Inc. (2010). *Integrated Architecture and CIP Sync Configuration*.

- [21] L. M. Surhone, M. T. Tennoe, and S. F. Henssonow, *Common Industrial Protocol*: VDM Verlag Dr. Mueller AG & Co. Kg, 2010.
- [22] ODVA, "The Common Industrial Protocol (CIP) and the Family of CIP Networks," ODVA, Ed., ed, 2010, p. 92.
- [23] ODVA. (2011). *ODVA EtherNet/IP*. Available:
<http://www.odva.org/Home/ODVATECHNOLOGIES/EtherNetIP/tabid/67/Inq/en-US/language/en-US/Default.aspx>
- [24] Cisco, "Managed vs. Unmanaged Switches," ed, 2007.
- [25] J. G. Gilbert and G. R. Diehl, "Application of programmable logic controllers to substation control and protection," *Power Delivery, IEEE Transactions on*, vol. 9, pp. 384-393, 1994.
- [26] N. Instruments. (2010). *A Comparison of PACs to PLCs*. Available:
<http://zone.ni.com/devzone/cda/tut/p/id/2960>
- [27] R. A. Inc., "Control Logix System," R. A. Inc., Ed., ed. Milwaukee, WI: Rockwell Automation Inc., 2011, p. 44.
- [28] R. A. Inc., "Integrated Architecture Technical Resources," R. Automation, Ed., ed. Milwaukee, WI: Rockwell Automation Inc., 2010, p. 60.
- [29] O. 22, "Understanding Programmable Automation Controllers (PACs) in Industrial Automation," ed: Opto 22, 2006, p. 7.
- [30] F. Grandoni, G. Nicosia, G. Oriolo, and L. Sanità, "Stable routing under the Spanning Tree Protocol," *Operations Research Letters*, vol. 38, pp. 399-404, 2010.
- [31] Cisco, "Defining Quality of Service," 2010.
- [32] Sick|Stegmann, "Single Turn Absolute Encoders AFS60 Single Turn Encoders," ed, 2010.
- [33] F. M. Company. (2011). *Fluke 45 Specifications*. Available:
<http://www.fluke.com/fluke/usen/bench-instruments/bench-multimeters/fluke-45.htm?PID=56082>
- [34] S. J. Chapman and S. Chapman, *Electric Machinery Fundamentals*: McGraw-Hill, 2010.
- [35] C. M. De Dominicis, P. Ferrari, A. Flammini, S. Rinaldi, and M. Quarantelli, "On the Use of IEEE 1588 in Existing IEC 61850-Based SASs: Current Behavior and Future Challenges," *Instrumentation and Measurement, IEEE Transactions on*, vol. 60, pp. 3070-3081, 2011.

APPENDIX A THREE PHASE MACHINERY PRIMER

A. Windings

The windings of a synchronous generator can be divided into two categories: field windings and armature windings. In general machinery field windings are associated with producing the main magnetic field while armature windings are where voltages are induced [34]. Using this in principal applied to synchronous machines, the field windings would be located on the rotor while the armature windings would be located on the stator of the machine.

In essence the rotor of a synchronous alternator can be viewed as a large electromagnet. The poles of these electromagnets are what act to created induced voltages on the armature windings. The way the magnetic poles are designed into the rotor construction of a synchronous machine can be divided into two categories: salient or non-salient. Salient pole machines have electromagnetics which protrude radially away from the shaft while non-salient pole machines have their field windings embedded flush with the surface of the rotor [34].

B. Saliency

When designing synchronous machines, a non-salient pole rotor will have the field windings embedded within the notches of the rotor, while field windings are wrapped around the pole itself for salient pole machines [1]. The physical design and construction for these rotors will play a large difference in speed characteristics for each machine. A general rule of thumb is that non-salient pole machines are typically designed with two or four pole rotors, while salient pole machines are typically designed for rotors with four or more poles [34].

C. Electromagnetic Field Interactions

It was previously stated that the rotor of a synchronous machine must either be constructed as a permanent electromagnet or a dc current must be applied to the field

windings in order to produce a permanent electromagnet. A dilemma that arises is how does one physically apply a dc field current to a rotating structure? This problem is resolved commonly in one of two ways in industry: the use of slip rings or brushes, or the uses of brushless exciters.

Slip rings are insulated metal rings which are applied to the shaft of the synchronous machine [34]. Each slip ring is equipped with a stationary carbon fiber brush which conducts electricity and rides on the shaft with very low friction [5]. One wire from the DC source is connected to each slip ring thus allowing for a DC current to be transferred from the source through the brushes and eventually to the field windings. The problem with this solution is the amount of maintenance required in order to constantly check for wear of the brushes. The other downfall to using slip rings is that the voltage drop created across the carbon brushes may be inefficient in terms of power transfer for synchronous machines that require large field currents [34]. In spite of these two downfalls, slip rings are often used on smaller synchronous machines as they are cost-effective [1].

The alternative to using slip rings is the use of brushless exciters. A brushless exciter consists of a small AC exciter generator with its armature windings routed through a DC rectifier and applied to the field windings. The stator windings of the brushless exciter are coupled to the armature windings of the synchronous machine. This setup allows for a small change in the DC current of the exciter generator to be reflected in the field windings of the synchronous machine without the use of slip rings [34].

As there are no brushes or slip rings, the amount of maintenance is greatly reduced by using a brushless exciter, but this setup is less cost-effective for smaller machines. For larger, high-priority machines Krause states that these machines will often support both types of excitation systems; that is the machine will be equipped with a brushless exciter and slip rings to be used in case of a failure [1].

D. Definition of Synchronism

Synchronous alternators are defined as synchronous, meaning the electrical frequency produced is locked or synchronized with the mechanical rate of rotation of the rotor. As previously discussed the rotor of a synchronous machine is like that of a permanent magnet. When the rotor turns within the stator of the machine, so too does the magnetic field. The rate of rotation of the magnetic field can be related to the electrical frequency produced by the following equation. Due to the fact that the rotor rotates at the same rate as the magnetic field, the equation relates the rotor rotation to the electrical frequency [1].

$$f_{se} = \frac{n_m P}{120} [\text{Hertz}] [34]$$

Equation 10.1

f_{se}	Electrical frequency in Hz
n_m	Mechanical speed of magnetic field, in r/min
P	Number of poles

E. Functionality-- Single Phase Machines

It was previously discussed that electrical power is generated by applying a rotating magnetic field to a stationary stator. The figure below depicts a rotating rotor with a sinusoidally distributed magnetic field in the center of a stationary coil. Assume that a magnetic flux density $\vec{B}$ is applied between the rotor and stator and varies sinusoidally with the mechanical angle α [34]. The magnitude of the radial flux density can be given as the following:

$$B = B_M \cos \alpha [\text{Tesla}]$$

Equation 10.2

As some points lie along the air-gap between the rotor and stator, the flux density vector points inward, rather than outward, signifying a negative flux. Since the rotor also rotates

within the stator as angular velocity ω_m , the magnitude of the flux density vector can be re-written as the following:

$$B = B_M \cos(\omega t - \alpha) \text{ [Tesla]}$$

Equation 10.3

After calculating the magnetic flux densities for the various positions of rotor within the stator, the following equation calculates the induced voltage on the stator coils [34]:

$$e = (\vec{v} \times \vec{B}) \cdot l \text{ [Volts]}$$

Equation 10.4

$\vec{v}$ Velocity of the wire relative to the magnetic field

$\vec{B}$ Magnetic flux density vector

l Length of conductor in the magnetic field

Simplifying to a square loop, the total voltage on the coil of wire becomes the following:

$$e_{induced} = -vB_m l \cos(\omega_m t - 180) + vB_m l \cos(\omega_m t)$$

$$e_{induced} = vB_m l \cos(\omega_m t) + vB_m l \cos(\omega_m t)$$

$$e_{induced} = 2vB_m l \cos(\omega_m t)$$

Equation 10.5

Substituting:

$$v = r\omega_m$$

Equation 10.6

Yields:

$$e_{induced} = 2r\omega_m B_m l \cos(\omega_m t)$$

Equation 10.7

Substituting:

$$\phi = 2rlB_m \left[\frac{W}{m^2} \right]$$

Equation 10.8

Yields:

$$e_{induced} = \phi \omega_m \cos(\omega_m t) \text{ [Volts]}$$

Equation 10.9

For multiple turns of wire, the previous equation becomes the following[5]:

$$e_{induced} = N_c \phi \omega_m \cos(\omega_m t) \text{ [Volts]}$$

Equation 10.10

F. Three Phase Machines

If three coils of N_c turns are placed in a stator, geometrically spaced 120 degrees apart, the induced voltages generated will be equal in magnitude, but displaced by 120 from each other.

$$\begin{aligned} E_{as} &= N_c \phi \omega \cos \omega t \\ E_{bs} &= N_c \phi \omega \cos(\omega t - 120) \\ E_{cs} &= N_c \phi \omega \cos(\omega t - 240) \end{aligned}$$

Equation 10.11

As a result, a three phase set of currents can generate a uniform magnetic field in a machine stator, and a uniform rotating magnetic field can generate a three phase set of voltages in a stator. The peak voltage of the induced voltage in the stator can be described by the following equation:

$$E_{max} = N_c \phi \omega \text{ [Volts]}$$

Equation 10.12

Therefore, the RMS voltage value can be described as the following:

$$E_A = \sqrt{2}\pi f \phi N_c [Volts]$$

Equation 10.13

The RMS voltage at the terminals of this machine is determined by the type of connection of the stator windings [5]. If the stator is wye connected, the terminal voltage will be $\sqrt{3}$ time the magnitude of E_A . If the machine is delta connected, the terminal voltage will show as E_A .

G. Torque

An alternative way to view the steady state behavior of a synchronous machine is by observing the torque produced by the machine. Taking the same example of a simplified AC synchronous machine with only one coil, the stator flux can be described as a function of mechanical rotor angle as the following:

$$B_s(\alpha) = B_s \sin \alpha [Tesla]$$

Equation 10.14

The force on the conductor coil is defined by the following equation:

$$\vec{F} = i(\vec{l} \times \vec{B}) [Newtons]$$

Equation 10.15

Which reduces to the following equation:

$$\vec{F} = ilB_s \sin \alpha [Newtons]$$

Equation 10.16

The torque on any conductor is defined by the following equation:

$$\vec{\tau} = (\vec{r} \times \vec{F}) [N - m]$$

Equation 10.17

Where r is the lever-arm of rotation and F is the applied force that was calculated in the previous equation. When substituting the corresponding practical values, the torque equation reduces to the following:

$$\vec{\tau} = 2rilB_s \sin \alpha [N - m]$$

Equation 10.18

The previous equation for torque can be reduced with a few simple assumptions and variable changes. The first assumption that can be made is the current flowing in the rotor produces a magnetic field of its own. The magnitude of the magnetizing intensity $\overline{H_r}$ is directly proportional to the current flowing in the rotor by a constant of proportionality. The second fact that can be stated is the angular difference between the peak stator flux and rotor magnetizing intensity can be defined by γ , where $\gamma = 180 - \alpha$ [34]. Making these changes into the induced torque equation yields the following:

$$\vec{\tau} = KH_r B_s \sin \alpha [N - m]$$

Equation 10.19

This equation in vector form can be written as the following:

$$\overrightarrow{\tau_{induced}} = K\overrightarrow{H_r} \times \overrightarrow{B_s} [N - m]$$

Equation 10.20

The net magnetic flux density for any AC machine is given by the following equation:

$$\overrightarrow{B_{net}} = \overrightarrow{B_R} + \overrightarrow{B_S} [Tesla]$$

Equation 10.21

Substituting this equation in to the induced torque equation yields the following expression:

$$\overrightarrow{\tau_{induced}} = k\overrightarrow{B_r} \times (\overrightarrow{B_{NET}} - \overrightarrow{B_r})[N - m]$$

Equation 10.22

Using vector identities, this expression reduces to the following:

$$\overrightarrow{\tau_{induced}} = k(\overrightarrow{B_r} \times \overrightarrow{B_{NET}}) - k(\overrightarrow{B_r} \times \overrightarrow{B_r})[N - m]$$

Equation 10.23

Which reduces to the following:

$$\overrightarrow{\tau_{induced}} = k\overrightarrow{B_r} \times \overrightarrow{B_{NET}}[N - m]$$

Equation 10.24

In scalar form, this equation becomes the following:

$$\overrightarrow{\tau_{induced}} = kB_r B_{NET} \sin \delta [N - m]$$

Equation 10.25

Where delta is the angle between the rotor and net magnetic flux density fields. The following equation also relates the torque of the machine to various field quantities.

$$T = \frac{\pi}{2} \left(\frac{P}{2} \right)^2 \Phi_R F_f \sin \delta_{Rf} [5]$$

Equation 10.26

Φ_R	Resultant air-gap flux per pole
F_f	Mmf of dc field winding
P	Number of poles
δ_{Rf}	Phase angle between magnetic axes of Φ_R and F_f

APPENDIX B SEVEN STATE MATRIX REDUCTION

1) *Ls Matrix*

The L_s matrix is the stator inductance matrix. This matrix will be a 3x3 matrix containing each of the three windings A, B, and C. The matrix is seen below:

$$L_s = \begin{bmatrix} L_{ls} + L_A - L_B \cos 2\theta_r & -\frac{1}{2}L_A - L_B \cos 2\left(\theta_r - \frac{\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cos 2\left(\theta_r + \frac{\pi}{3}\right) \\ -\frac{1}{2}L_A - L_B \cos 2\left(\theta_r - \frac{\pi}{3}\right) & L_{ls} + L_A - L_B \cos 2\left(\theta_r - \frac{2\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cos 2(\theta_r + \pi) \\ -\frac{1}{2}L_A - L_B \cos 2\left(\theta_r + \frac{\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cos 2(\theta_r + \pi) & L_{ls} + L_A - L_B \cos 2\left(\theta_r + \frac{2\pi}{3}\right) \end{bmatrix}$$

The value of L_{ls} refers to the leakage inductance of the winding and L_A and L_B refer to the average and oscillating inductance of the corresponding winding.

2) *Lsr Matrix*

The L_{sr} matrix is the stator to rotor mutual inductance matrix. This matrix will be a 3x3 matrix containing the quadrature damping winding, field winding, and direct axis damping winding.

$$L_{sr} = \begin{bmatrix} L_{skq} \cos \theta_r & L_{sfd} \sin \theta_r & L_{skd} \sin \theta_r \\ L_{skq} \cos\left(\theta_r - \frac{2\pi}{3}\right) & L_{sfd} \sin\left(\theta_r - \frac{2\pi}{3}\right) & L_{skd} \sin\left(\theta_r - \frac{2\pi}{3}\right) \\ L_{skq} \cos\left(\theta_r + \frac{2\pi}{3}\right) & L_{sfd} \sin\left(\theta_r + \frac{2\pi}{3}\right) & L_{skd} \sin\left(\theta_r + \frac{2\pi}{3}\right) \end{bmatrix}$$

3) *Lr Matrix*

The L_r matrix is the rotor to rotor self-inductance matrix. This matrix will be a 3x3 matrix containing both damping windings as well as the field winding.

$$L_r = \begin{bmatrix} L_{kqkq} & L_{kqfd} & L_{kqkd} \\ L_{fdkq} & L_{fdfd} & L_{fdkd} \\ L_{kd kq} & L_{kd fd} & L_{kd kd} \end{bmatrix}$$

As the field windings and quadrature windings and direct axis damping windings and quadrature axis damping windings are 90 degrees out of phase, there is no inductance between them. Thus this matrix reduces to the following:

$$L_r = \begin{bmatrix} L_{lkq} + L_{mkq} & 0 & 0 \\ 0 & L_{lfd} + L_{mfd} & L_{fdkd} \\ 0 & L_{fdkd} & L_{lkd} + L_{mkd} \end{bmatrix}$$

4) Magnetization Inductances

The following are the definitions of the magnetization inductances of synchronous machines:

$$L_{mq} = \frac{3}{2}(L_A - L_B)$$

$$L_{md} = \frac{3}{2}(L_A + L_B)$$

5) Mutual to Magnetizing Inductance Relationship

The following shows the relationship between the mutual inductances between the rotor and stator and magnetizing inductances previously defined.

$$L_{skq} = \left(\frac{N_{kq}}{N_s}\right) \frac{2}{3} L_{mq}$$

$$L_{sfd} = \left(\frac{N_{fd}}{N_s}\right) \frac{2}{3} L_{md}$$

$$L_{skd} = \left(\frac{N_{kd}}{N_s}\right) \frac{2}{3} L_{md}$$

$$L_{mkq} = \left(\frac{N_{kq}}{N_s}\right)^2 \frac{2}{3} L_{mq}$$

$$L_{mfd} = \left(\frac{N_{fd}}{N_s}\right)^2 \frac{2}{3} L_{md}$$

$$L_{mkd} = \left(\frac{N_{kd}}{N_s}\right)^2 \frac{2}{3} L_{md}$$

$$L_{fdkd} = \left(\frac{N_{kd}}{N_{fd}} \right) L_{mfd} = \left(\frac{N_{fd}}{N_{kd}} \right) L_{mkd}$$

6) Reduction of the Lsr Matrix

Using the previous definitions that relate mutual inductances to magnetizing inductances the Lsr matrix reduces to a new matrix which I will call Lsr'. This matrix can be seen below:

$$L'_{sr} = \begin{bmatrix} L_{mq} \cos \theta_r & L_{md} \sin \theta_r & L_{md} \sin \theta_r \\ L_{mq} \cos(\theta_r - \frac{2\pi}{3}) & L_{md} \sin(\theta_r - \frac{2\pi}{3}) & L_{md} \sin(\theta_r - \frac{2\pi}{3}) \\ L_{mq} \cos(\theta_r + \frac{2\pi}{3}) & L_{md} \sin(\theta_r + \frac{2\pi}{3}) & L_{md} \sin(\theta_r + \frac{2\pi}{3}) \end{bmatrix}$$

7) Reduction of the Lr Matrix

Likewise, the Lr matrix can be reduced by these definitions and will reduce to the following matrix which is called Lr'.

$$L'_r = \begin{bmatrix} L'_{lkq} + L_{mq} & 0 & 0 \\ 0 & L'_{lfd} + L_{md} & L_{md} \\ 0 & L_{md} & L'_{lkd} + L_{md} \end{bmatrix}$$

8) Flux Linkage Matrix Equation

Using the inductance matrices previously developed in this document the flux linkage equations can now be written in matrix form.

$$\begin{bmatrix} \lambda_{abcs} \\ \lambda'_{qdr} \end{bmatrix} = \begin{bmatrix} L_s & L'_{sr} \\ \frac{2}{3}(L'_{sr})^T & L'_r \end{bmatrix} \begin{bmatrix} i_{abcs} \\ i'_{qdr} \end{bmatrix}$$

9) Revised Voltage Matrix Equations

After obtaining the flux linkage equations, the following voltage equations can be developed:

$$\begin{bmatrix} v_{abcs} \\ v'_{qdr} \end{bmatrix} = \begin{bmatrix} r_s + pL_s & pL'_{sr} \\ \frac{2}{3}p(L'_{sr})^T & r'_r + pL'_r \end{bmatrix} \begin{bmatrix} i_{abcs} \\ i'_{qdr} \end{bmatrix}$$

The above equation defines the voltages for the machine in matrix form. The variables used here are in machine variables.

10) Transformation of the L_s Matrix

For a three phase inductive circuit:

$$\begin{aligned} \lambda_{abcs} &= L_s i_{abcs} \\ K_s^{-1} \lambda_{qdo} &= L_s K_s^{-1} i_{qdo} \\ \lambda_{qdo} &= K_s L_s K_s^{-1} i_{qdo} \end{aligned}$$

Applying the Matrix L_s to this equation yields the following:

$$\begin{bmatrix} \lambda_q \\ \lambda_d \\ \lambda_0 \end{bmatrix} = \begin{bmatrix} L_{ls} + L_{mq} & 0 & 0 \\ 0 & L_{ls} + L_{md} & 0 \\ 0 & 0 & L_{ls} \end{bmatrix} \begin{bmatrix} i_q \\ i_d \\ i_0 \end{bmatrix}$$

11) Transformation of the L_r Matrix

As the L_r matrix is already referenced to the rotor, there is no need to transform this matrix and it stays as is:

$$L'_r = \begin{bmatrix} L'_{lkq} + L_{mq} & 0 & 0 \\ 0 & L'_{lfd} + L_{md} & L_{md} \\ 0 & L_{md} & L'_{lkd} + L_{md} \end{bmatrix}$$

12) Transformation of the L_{sr} Matrix

For a three phase inductive circuit with two damping windings and one field winding, the following is true:

$$\begin{aligned} \lambda_{qdr} &= L'_{sr} i_{abcs} \\ \lambda_{qdr} &= L'_{sr} K_s^{-1} i_{qdo} \\ \lambda_{qdr} &= L'_{sr} K_s^{-1} i_{qdo} \end{aligned}$$

This yields the following:

$$\lambda_{sr} = \begin{bmatrix} \lambda_{qs} \\ \lambda_{ds} \\ \lambda_{0s} \end{bmatrix} = \begin{bmatrix} L_{mq} & 0 & 0 \\ 0 & L_{md} & L_{md} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \end{bmatrix}$$

13) (Lsr)T transformed

The transpose of (Lsr) yields the following:

$$(\lambda_{sr})^T = \begin{bmatrix} \lambda_{kq} \\ \lambda_{fd} \\ \lambda_{kd} \end{bmatrix} = \begin{bmatrix} L_{mq} & 0 & 0 \\ 0 & L_{md} & 0 \\ 0 & L_{md} & 0 \end{bmatrix} \begin{bmatrix} i_{kq} \\ i_{fd} \\ i_{kd} \end{bmatrix}$$

14) Ls Transformed

The Ls matrix transformed yields the following:

$$\lambda_s = \begin{bmatrix} \lambda_{qs} \\ \lambda_{ds} \\ \lambda_{0s} \end{bmatrix} = \begin{bmatrix} L_q & 0 & 0 \\ 0 & L_d & 0 \\ 0 & 0 & L_{ts} \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \end{bmatrix}$$

15) Lr Transformed

As the Lr matrix is already in the rotor reference frame, there is no need for a transformation and the matrix yields the following flux linkages:

$$\lambda_r = \begin{bmatrix} \lambda_{kq} \\ \lambda_{fd} \\ \lambda_{kd} \end{bmatrix} = \begin{bmatrix} L_{kq} & 0 & 0 \\ 0 & L_{fd} & L_{md} \\ 0 & L_{md} & L_{kd} \end{bmatrix} \begin{bmatrix} i_{kq} \\ i_{fd} \\ i_{kd} \end{bmatrix}$$

16) Speed Voltages

The speed voltages come into play when the voltage equations are written. These voltages affect the overall terminal voltage of the machine based on the electrical frequency of the system.

$$\begin{aligned} v_{abcs} &= p\lambda_{abcs} + ri \\ K_s^{-1}v_{qd0o} &= p[K_s^{-1}\lambda_{qdo}] \\ v_{qd0} &= K_s p[K_s^{-1}]\lambda_{qdo} + K_s K_s^{-1} p\lambda_{qdo} \end{aligned}$$

$$p[K_s^{-1}] = \omega \begin{bmatrix} -\sin \theta & \cos \theta & 0 \\ -\sin\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta - \frac{2\pi}{3}\right) & 0 \\ -\sin\left(\theta + \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) & 0 \end{bmatrix}$$

$$K_s p K_s^{-1} = \omega \begin{bmatrix} \cos \theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin \theta & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} -\sin \theta & \cos \theta & 0 \\ -\sin\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta - \frac{2\pi}{3}\right) & 0 \\ -\sin\left(\theta + \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) & 0 \end{bmatrix}$$

$$K_s p K_s^{-1} = \omega \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$v_{qd0} = \omega \lambda_{dqs} + p \lambda_{dq0}$$

Where:

$$\lambda_{dqs} = \begin{bmatrix} \lambda_{ds} \\ -\lambda_{qs} \\ 0 \end{bmatrix}$$

This yields as a result:

$$v_{qs} = \omega \lambda_{ds} + p \lambda_{qs}$$

$$v_{ds} = -\omega \lambda_{qs} + p \lambda_{ds}$$

$$v_{0s} = p \lambda_{0s}$$

This yields the following equations:

$$v_{qs} = \omega L_d i_d + \omega L_{md} i_{fd} + \omega L_{md} i_{kd} + p \lambda_{qs} \text{ [Volts]}$$

$$v_{ds} = -\omega L_q i_q - \omega L_{mq} i_{kq} + p \lambda_{ds} \text{ [Volts]}$$

$$v_{0s} = p \lambda_{0s} \text{ [Volts]}$$

APPENDIX C PRECISION TIME PROTOCOL

A. Best Master Clock Algorithm

It was previously stated that PTP was designed to be an administration free protocol in that devices would negotiate amongst themselves to determine a hierarchy of clocks. The algorithm used to determine the hierarchy of clocks is known as the Best Master Clock Algorithm (BMCA). The BMCA as defined by the standard is the strict arbitration process employed to determine the status of each network node, either master or slave [8, 17, 35]. As its name implies, the BMCA will determine the best master clock and name it the grandmaster of the PTP system. All remaining clocks within the system are ultimately synchronized to the grandmaster [8].

Announce messages are sent approximately every two seconds for any PTP device claiming to be a master [8, 17, 35]. These announce messages contain information about how good the clock is compared to a scale. When a node receives an announce message from another device it compares the credentials that it receives to its own. The better of the two clocks will serve as the master, while the lesser, the slave. This process will continue until the status for every clock within the network is determined [17].

The BMCA utilizes four criteria to determine the better of two clocks. These factors include: clock class, accuracy, variance, and priority. Clock class defines the relative measure of clock quality. Accuracy defines how close the clock meters time to an absolute reference. Variance is the measure of the clock's stability. Priority is a manual override that can be established if a network administrator wants one clock to serve as grandmaster over another [8].

One main advantage to implementing PTP in a system is that it will dynamically update to topology changes [17]. For example, if the current grandmaster is removed from the system, the BMCA will attempt to designate a new grandmaster from the remaining clocks. On the other hand, if a clock with better credentials is added to the system, the BMCA will designate this new clock as grandmaster.

Figure 46 Synchronization Messages (latency)

Figure 46 depicts the synchronization process for clocks utilizing the PTP protocol. PTP utilizes four messages in order to synchronize two clocks: sync message, follow up message, delay request message, and delay response message. These messages are transmitted from master to slave and allow the clocks to make frequency adjustments to adjust the rate at which the clocks meter time. Additionally, the messages allow for the clocks to measure the phase delay between master and slave and allow for a value correction [17]. The frequency adjustment is made by utilizing the sync and follow up messages, while the phase adjustment is made by utilizing the delay request and delay response messages [8]. Every time that the synchronization process occurs, timestamps t_1 - t_4 are collected to determine the offset from master and frequency adjustments for the slave clocks. Clocks that are PTP compliant that make both a frequency and value adjustment are known as tunable clocks.

B. CIP Sync Object

It has been previously stated that CIP Sync combines the CIP protocol with PTP for time synchronization. The principle behind how EtherNet/IP synchronizes time across a network of distributed I/O can be attributed to the CIP Sync Object. This object provides an interface to CIP Sync that allows devices such as logic controllers to access the synchronization mechanism.

CIP Sync defines an offset clock model that addresses the requirements for various control applications. This model is necessary as PTP defines a mechanism for distributing and synchronizing time, but fails to define a mechanism for compensating for step changes in time that may occur at the grandmaster source [20]. It should be noted that when a slave synchronizes its clock to that of its master, the frequency of the clock is adjusted, not the absolute time value of the master [8].

Figure 47 CIP Sync Object Model

Figure 47 defines the CIP Sync Object at a high level. This model shows a PTP master represented by the circle and a PTP time slave represented by the top right side rectangle. In this example PTP is used to discipline a local clock such that it ticks and meters time at the same rate of the PTP master. The slave clock also maintains an offset between the local clock time and the PTP system time [18]. Any small delta, or step change in time, will cause the slave device to make a small adjustment to the system to local clock offset value. In addition, the slave device will continue to tune its clock. A large step change will result in the device updating the offset value, but not tune its clock [8, 18, 20]. As a result, cyclic tasks such as SCADA reads, can be scheduled based upon the local clock and will not be affected by large step changes in time at the grandmaster source.

CIP Sync represents time as a 64 bit long-integer (LINT) that can be expressed in either nanoseconds or microseconds. The starting reference point in time for CIP Sync is January 1, 1970 starting at 12:00 AM. This time is represented in Universal Time Coordinated (UTC) and adjusted to include leap seconds [8].

In order to represent this 64 bit LINT in a readable format, an algorithm must be developed to compute the current date and time in a readable, understandable format for analysis. The algorithm consisted of several mathematical operations to convert the LINT into two strings that are in the following format: MM/DD/YYYY and HH:MM:SS:USUSUSUS.

APPENDIX D COMPONENT SELECTION

A. Local Rack Component Descriptions

a) 1756-L73Processor

The main component for any PAC system is the processor. The processor that was chosen for this application was the 1756-L73 processor. The L73 is equipped with 8 MB of onboard program memory as well as 1MB of I/O mapped memory. In addition, programs or start data can be stored on a Secure Digital (SD) card in case of power failure. The L73 processor contains Intel Duo Core RISC chips that allow the controller to operate much like many laptop or desktop computers today. The processor can be programmed over a variety of protocols including, EtherNet/IP, TCP/IP, USB, and Modbus.

For this project, the main processor was placed in slot zero of the PAC rack. Figure 23 above also shows a second processor in slot one. A second processor was placed in the rack for two purposes. The first reason for placing a second processor is simple redundancy. If one processor would fault or fail to operate correctly, the code will react by transferring control to the second processor so that there is no downtime in the data collection process. If the primary processor in slot zero is operating correctly, the second processor will be used as a second CPU to increase the throughput of the system. Since this application requires precise timing and data collection, one processor will be used to handle timing devices, while the other will be used to perform data processing and distribution.

b) 1756-EN2T

In order for any PAC to be able to communicate with the other devices, a form of data network must be utilized. In this case, EtherNet/IP (Ethernet/Industrial Protocol) was chosen as the communication and network protocol for this system. Ethernet/IP was previously defined as a CIP protocol that allows for the communication of industrial control devices. In addition, this protocol also allows for the time synchronization of devices over PTP as well as motion commands to be executed over twisted pair copper media. This module allows for up to 256 simultaneous EtherNet/IP connections as well

as 128 simultaneous TCP/IP (IEEE 802.3) connections. These modules transfer data at a rate of 100 Mbits/second. In addition, the EN2T module acts as an Ethernet bridge segmenting the network into two separate areas. This allows for isolation between the PAC and the remaining portions of the network which is used to data traffic control on the network.

For the implementation of this system, two 1756-EN2T modules will be utilized. These modules will be placed in slots two and seven. There are two EN2T modules in this system for redundancy purposes. The EN2T serves as the only way for the PAC to communicate to the rest of the system that will be discussed later in this document. If there is a system or module fault on one of the modules, the other module will be commanded to start communicating the data back to the network.

The secondary purpose of the communication module is to distribute PTP throughout the network. Since this project requires precise timing, every clock within the system should be synchronized to one master clock, in this case GPS. The EN2T supports CIP Sync, previously discussed in Chapter five, which allows for synchronizing of all IEEE 1588 devices on the network to within +/- 100 ns of a master reference.

c) 1756HP-TIME

In order to leverage the accuracy of GPS clock satellites, the 1756HP-TIME module was used for this project. Since GPS is considered a stable time reference for all precision time applications, it was also selected as a time reference for this project. When dealing with the power system, locations where measurements of interest need to be made are often separated by wide distances. In order for these measurements to be meaningful, they need to be taken at the same time with a high precision of timing. This high precision is offered by GPS time and is why it is chosen for time synchronized distributed measurements today. For the same reasons, this is why GPS was chosen for this application.

The 1756HP-TIME module is a versatile module manufactured by HiProm Technologies Inc. This module functions on four different modes which include: GPS, IRIG-B, PTP, and NTP. The GPS and PTP modes will be used for this application. The GPS mode utilizes a GPS antenna with a TNC to SMA connection and tracks GPS satellites throughout the day. The time received by the module and the PPS will be distributed throughout the system and rack backplane.

Time is distributed throughout the network via a two port Ethernet switch mounted on the front of the time module. The network port is then connected to a managed switch or router, so that PTP can be distributed throughout the network. In the event of a GPS signal loss, the module is equipped with an accurate crystal that will meter time until the GPS signal is regained or a new master clock is chosen by the PTP Best Master Clock Algorithm. This module was ultimately chosen as it supports IEEE-1588, which requires the use of tunable clocks in each eligible PTP device. When time synchronization is required, the processor clock will not only be brought into alignment with the GPS clock, it will be tuned to it once a second in order to meter time at the same rate as the GPS clock.

d) 1756-IB16SOE

As the system requires precise timing across the system, an accurate timestamp must also be generated for the data to become meaningful. This accurate timestamp will be achieved with the 1756-IB16SOE. This module is a sixteen point I/O module with the ability to timestamp when inputs are enabled or disabled. This functional performance is known as a Sequence of Events (SOE) task. This module was placed in slot four of the Control Logix rack. This module can be triggered either internally by the processor or by an external input such as a 24 volt DC signal.

The sequence of events module contains a free-wheeling clock which is synchronized to the GPS clock via the CIP sync protocol. This module was configured for CIP sync per point. This means that whenever an input pin transition, in this case low to high, is recognized, the module applies a UTC timestamp to the data. In this mode, the sequence

of events module is accurate to 100 ns. This module is also able to be triggered via software. In other words, the software program can internally trigger the timestamp by setting bits within the control register for the sequence of events module. Both of these schemes were compared and hardware timestamps were chosen, as it represents a more realistic real world input to the system setup.

e) 1756-HIST2G

Since the L73 processor only contains 8MB of memory, the system needs a way to store large amounts of data from test trials for analysis. The solution to this problem was achieved by using the 1756-HIST2G module. This module is better known as the historian module, and its task as its name implies, acts as a database collection point for user defined data. The historian sits in slot eight of the local Control Logix rack. The historian module is programmed over EtherNet/IP and can be configured easily from any web browser. In all, this module can record up to two gigabytes of data before data either needs to be rewritten or transferred to another device. This value can be changed by substituting a new compact flash (CF) card of a different value into the module. Any standard compact flash card can be used as it will be formatted before data is transferred and stored on the card.

The module is configured by choosing user defined points, or tags from the PAC system to be collected on a user defined time interval. Since the data I was collecting was being sampled every 100 milliseconds, or 0.1 seconds, the historian module was configured to obtain data at 100 millisecond intervals as the historian clock was synchronized to that of the GPS clock via PTP. Three different data points were chosen for collection by the historian for this application. These data points include a UTC time tag, an encoder value between 0 and 4095, and a converted angular value between 0 and 360 degrees.

The historian module also is equipped with a TCP/IP connection on the front of the module. This connection serves as a correlation between the PAC and a server that contains a database to transfer data from the historian module to this server. For this application a server will be setup to store data on the historian module for more post

processing. The option to configure the server correlation option can be set via the webpage connection to the module via an Open Database Connectivity (ODBC) connection.

B. Remote Rack Components

a) 1734-SSI

The module that interfaces with the encoder is the 1734-SSI. SSI or synchronous serial interface and will be discussed later in this chapter with encoder selection and justification. This module is wired to the output of the encoder and takes all seven inputs from the encoder, two data lines, two clock lines, two voltage lines for the SSI sensor, and the shield or ground cable.

In addition, this module had one discrete input which was defined as input one. Input one was used as a latch feature for this module. Whenever input one was enabled the value that the SSI encoder was reading would be latched and stored so that it could be read by the processor. This feature was very important as the encoder was spinning very near 1800 RPM, synchronous speed of a four pole machine, and the values were changing too quickly to be ready by the processor accurately on the fly.

Additionally, this module was configured for the exact string of data sent back by the SSI encoder. For example, every SSI word was 15 bits long from the encoder, but the least three significant bits were cyclic redundancy bits for error checking. The module was configured to read these bits, check for errors. If there were no data errors, these bits were masked and removed them from the word leaving only the twelve bits of true data. Other parameters that are able to be set were the baud rate of the encoder, time delay between successive SSI words, and data format including gray code or normal binary feedback. For this project the SSI module was placed in slot two of the point I/O cluster.

b) 1734-OV4E

The first module placed in the point I/O rack is the 1734-OV4E. This module is an output module that is designed to sink 24 VDC. This module was chosen as the SSI module was implemented with a data latch. The manual of the 1734-SSI module stated that input one was a sourcing input, therefore in order to turn the input on that pin must be grounded. The easiest way to ground a pin that sources 24 VDC is to sink 24 VDC through an output. The key here is to make sure that the common shared by both modules is the same. In essence this module will be used to issue the latch command to the SSI module to freeze the current encoder value.

APPENDIX E INFINITE BUS TEST DATA

A. No Time Synchronization Data

Encoder_Data	<normal>	{...}	Column1
Encoder_Data[0]		231	-28
Encoder_Data[1]		203	-28
Encoder_Data[2]		175	57
Encoder_Data[3]		232	-5
Encoder_Data[4]		227	13
Encoder_Data[5]		240	-39
Encoder_Data[6]		201	-12
Encoder_Data[7]		189	-22
Encoder_Data[8]		167	52
Encoder_Data[9]		219	0
Encoder_Data[10]		219	46
Encoder_Data[11]		265	-28
Encoder_Data[12]		237	18
Encoder_Data[13]		255	-28
Encoder_Data[14]		227	-39
Encoder_Data[15]		188	-11
Encoder_Data[16]		177	75
Encoder_Data[17]		252	-23
Encoder_Data[18]		229	-33
Encoder_Data[19]		196	11
Encoder_Data[20]		207	39
Encoder_Data[21]		246	19
Encoder_Data[22]		265	-16
Encoder_Data[23]		249	-28
Encoder_Data[24]		221	41
Encoder_Data[25]		262	-62
Encoder_Data[26]		200	12
Encoder_Data[27]		212	-80
Encoder_Data[28]		132	114
Encoder_Data[29]		246	-5
Encoder_Data[30]		241	-16
Encoder_Data[31]		225	1
Encoder_Data[32]		226	-64
Encoder_Data[33]		162	98
Encoder_Data[34]		260	-5
Encoder_Data[35]		255	-16
Encoder_Data[36]		239	-56
Encoder_Data[37]		183	46

Encoder_Data[38]	229	33
Encoder_Data[39]	262	-34
Encoder_Data[40]	228	-22
Encoder_Data[41]	206	9
Encoder_Data[42]	215	57
Encoder_Data[43]	272	-24
Encoder_Data[44]	248	-33
Encoder_Data[45]	215	39
Encoder_Data[46]	254	-84
Encoder_Data[47]	170	115
Encoder_Data[48]	285	-23
Encoder_Data[49]	262	18
Encoder_Data[50]	280	-28
Encoder_Data[51]	252	-40
Encoder_Data[52]	212	-17
Encoder_Data[53]	195	86
Encoder_Data[54]	281	-46
Encoder_Data[55]	235	1
Encoder_Data[56]	236	-4
Encoder_Data[57]	232	-34
Encoder_Data[58]	198	73
Encoder_Data[59]	271	-45
Encoder_Data[60]	226	17
Encoder_Data[61]	243	0
Encoder_Data[62]	243	-17
Encoder_Data[63]	226	30
Encoder_Data[64]	256	11
Encoder_Data[65]	267	-11
Encoder_Data[66]	256	-11
Encoder_Data[67]	245	-39
Encoder_Data[68]	206	74
Encoder_Data[69]	280	-12
Encoder_Data[70]	268	6
Encoder_Data[71]	274	-12
Encoder_Data[72]	262	-17
Encoder_Data[73]	245	-29
Encoder_Data[74]	216	34
Encoder_Data[75]	250	12
Encoder_Data[76]	262	23
Encoder_Data[77]	285	-35
Encoder_Data[78]	250	-10

Encoder_Data[79]	240	-46
Encoder_Data[80]	194	62
Encoder_Data[81]	256	29
Encoder_Data[82]	285	-34
Encoder_Data[83]	251	17
Encoder_Data[84]	268	10
Encoder_Data[85]	278	-11
Encoder_Data[86]	267	-34
Encoder_Data[87]	233	-22
Encoder_Data[88]	211	51
Encoder_Data[89]	262	11
Encoder_Data[90]	273	-11
Encoder_Data[91]	262	-86
Encoder_Data[92]	176	96
Encoder_Data[93]	272	-12
Encoder_Data[94]	260	23
Encoder_Data[95]	283	-17
Encoder_Data[96]	266	-40
Encoder_Data[97]	226	-29
Encoder_Data[98]	197	74
Encoder_Data[99]	271	0
Encoder_Data[100]	271	-35
Encoder_Data[101]	236	-6
Encoder_Data[102]	230	-11
Encoder_Data[103]	219	62
Encoder_Data[104]	281	-119
Encoder_Data[105]	162	34
Encoder_Data[106]	196	-11
Encoder_Data[107]	185	39
Encoder_Data[108]	224	0
Encoder_Data[109]	224	-6
Encoder_Data[110]	218	-33
Encoder_Data[111]	185	16
Encoder_Data[112]	201	-57
Encoder_Data[113]	144	91
Encoder_Data[114]	235	-11
Encoder_Data[115]	224	-17
Encoder_Data[116]	207	-18
Encoder_Data[117]	189	-7
Encoder_Data[118]	182	45
Encoder_Data[119]	227	28

Encoder_Data[120]	255	-29
Encoder_Data[121]	226	-57
Encoder_Data[122]	169	-1
Encoder_Data[123]	168	74
Encoder_Data[124]	242	-23
Encoder_Data[125]	219	-52
Encoder_Data[126]	167	73
Encoder_Data[127]	240	34
Encoder_Data[128]	274	-23
Encoder_Data[129]	251	-62
Encoder_Data[130]	189	11
Encoder_Data[131]	200	44
Encoder_Data[132]	244	0
Encoder_Data[133]	244	-16
Encoder_Data[134]	228	-1
Encoder_Data[135]	227	-35
Encoder_Data[136]	192	79
Encoder_Data[137]	271	-18
Encoder_Data[138]	253	-11
Encoder_Data[139]	242	-12
Encoder_Data[140]	230	-40
Encoder_Data[141]	190	34
Encoder_Data[142]	224	27
Encoder_Data[143]	251	-62
Encoder_Data[144]	189	12
Encoder_Data[145]	201	32
Encoder_Data[146]	233	34
Encoder_Data[147]	267	-34
Encoder_Data[148]	233	-12
Encoder_Data[149]	221	-28
Encoder_Data[150]	193	-23
Encoder_Data[151]	170	72
Encoder_Data[152]	242	-52
Encoder_Data[153]	190	18
Encoder_Data[154]	208	44
Encoder_Data[155]	252	-23
Encoder_Data[156]	229	-13
Encoder_Data[157]	216	11
Encoder_Data[158]	227	-41
Encoder_Data[159]	186	-7
Encoder_Data[160]	179	62

Encoder_Data[161]	241	17
Encoder_Data[162]	258	-110
Encoder_Data[163]	148	74
Encoder_Data[164]	222	-126
Encoder_Data[165]	96	159
Encoder_Data[166]	255	-24
Encoder_Data[167]	231	-24
Encoder_Data[168]	207	-23
Encoder_Data[169]	184	67
Encoder_Data[170]	251	-1
Encoder_Data[171]	250	-23
Encoder_Data[172]	227	-7
Encoder_Data[173]	220	-75
Encoder_Data[174]	145	84
Encoder_Data[175]	229	-53
Encoder_Data[176]	176	62
Encoder_Data[177]	238	-81
Encoder_Data[178]	157	22
Encoder_Data[179]	179	-76
Encoder_Data[180]	103	103
Encoder_Data[181]	206	-35
Encoder_Data[182]	171	16
Encoder_Data[183]	187	43
Encoder_Data[184]	230	-1
Encoder_Data[185]	229	-12
Encoder_Data[186]	217	-13
Encoder_Data[187]	204	-24
Encoder_Data[188]	180	-30
Encoder_Data[189]	150	33
Encoder_Data[190]	183	39
Encoder_Data[191]	222	-12
Encoder_Data[192]	210	-59
Encoder_Data[193]	151	-12
Encoder_Data[194]	139	79
Encoder_Data[195]	218	-18
Encoder_Data[196]	200	-42
Encoder_Data[197]	158	-18
Encoder_Data[198]	140	73
Encoder_Data[199]	213	-18
Encoder_Data[200]	195	-47
Encoder_Data[201]	148	-8

Encoder_Data[202]	140	-35
Encoder_Data[203]	105	95
Encoder_Data[204]	200	-36
Encoder_Data[205]	164	-1
Encoder_Data[206]	163	5
Encoder_Data[207]	168	-30
Encoder_Data[208]	138	44
Encoder_Data[209]	182	28
Encoder_Data[210]	210	-25
Encoder_Data[211]	185	-35
Encoder_Data[212]	150	-30
Encoder_Data[213]	120	62
Encoder_Data[214]	182	5
Encoder_Data[215]	187	-2
Encoder_Data[216]	185	-47
Encoder_Data[217]	138	-42
Encoder_Data[218]	96	10
Encoder_Data[219]	106	44
Encoder_Data[220]	150	38
Encoder_Data[221]	188	-64
Encoder_Data[222]	124	-2
Encoder_Data[223]	122	5
Encoder_Data[224]	127	5
Encoder_Data[225]	132	10
Encoder_Data[226]	142	-8
Encoder_Data[227]	134	-24
Encoder_Data[228]	110	-47
Encoder_Data[229]	63	68
Encoder_Data[230]	131	-7
Encoder_Data[231]	124	4
Encoder_Data[232]	128	-58
Encoder_Data[233]	70	15
Encoder_Data[234]	85	61
Encoder_Data[235]	146	-35
Encoder_Data[236]	111	-42
Encoder_Data[237]	69	-1
Encoder_Data[238]	68	-19
Encoder_Data[239]	49	50
Encoder_Data[240]	99	-69
Encoder_Data[241]	30	61
Encoder_Data[242]	91	-64

Encoder_Data[243]	27	90
Encoder_Data[244]	117	16
Encoder_Data[245]	133	-76
Encoder_Data[246]	57	27
Encoder_Data[247]	84	62
Encoder_Data[248]	146	-12
Encoder_Data[249]	134	10
Encoder_Data[250]	144	-58
Encoder_Data[251]	86	22
Encoder_Data[252]	108	21
Encoder_Data[253]	129	-18
Encoder_Data[254]	111	-1
Encoder_Data[255]	110	-18
Encoder_Data[256]	92	51
Encoder_Data[257]	143	-36
Encoder_Data[258]	107	33
Encoder_Data[259]	140	-109
Encoder_Data[260]	31	44
Encoder_Data[261]	75	-46
Encoder_Data[262]	29	107
Encoder_Data[263]	136	-29
Encoder_Data[264]	107	-36
Encoder_Data[265]	71	0
Encoder_Data[266]	71	51
Encoder_Data[267]	122	10
Encoder_Data[268]	132	-58
Encoder_Data[269]	74	11
Encoder_Data[270]	85	-46
Encoder_Data[271]	39	68
Encoder_Data[272]	107	-24
Encoder_Data[273]	83	-1
Encoder_Data[274]	82	-36
Encoder_Data[275]	46	33
Encoder_Data[276]	79	-63
Encoder_Data[277]	16	108
Encoder_Data[278]	124	-76
Encoder_Data[279]	48	16
Encoder_Data[280]	64	-40
Encoder_Data[281]	24	61
Encoder_Data[282]	85	22
Encoder_Data[283]	107	-12

Encoder_Data[284]	95	-35
Encoder_Data[285]	60	-30
Encoder_Data[286]	30	90
Encoder_Data[287]	120	-35
Encoder_Data[288]	85	-17
Encoder_Data[289]	68	-58
Encoder_Data[290]	10	50
Encoder_Data[291]	60	-18
Encoder_Data[292]	42	67
Encoder_Data[293]	109	-29
Encoder_Data[294]	80	-41
Encoder_Data[295]	39	0
Encoder_Data[296]	39	-23
Encoder_Data[297]	16	33
Encoder_Data[298]	49	10
Encoder_Data[299]	59	-24
Encoder_Data[300]	35	-35

B. Varying Frequency with Time Synchronization Data

Encoder_Data	<normal>	{...}
Encoder_Data[0]		1610
Encoder_Data[1]		1604
Encoder_Data[2]		1579
Encoder_Data[3]		1648
Encoder_Data[4]		1643
Encoder_Data[5]		1581
Encoder_Data[6]		1576
Encoder_Data[7]		1644
Encoder_Data[8]		1602
Encoder_Data[9]		1634
Encoder_Data[10]		1535
Encoder_Data[11]		1586
Encoder_Data[12]		1524
Encoder_Data[13]		1612
Encoder_Data[14]		1643
Encoder_Data[15]		1638
Encoder_Data[16]		1557
Encoder_Data[17]		1533
Encoder_Data[18]		1639
Encoder_Data[19]		1615

Encoder_Data[20]	1591
Encoder_Data[21]	1529
Encoder_Data[22]	1636
Encoder_Data[23]	1611
Encoder_Data[24]	1606
Encoder_Data[25]	1563
Encoder_Data[26]	1539
Encoder_Data[27]	1608
Encoder_Data[28]	1603
Encoder_Data[29]	1560
Encoder_Data[30]	1555
Encoder_Data[31]	1531
Encoder_Data[32]	1581
Encoder_Data[33]	1576
Encoder_Data[34]	1571
Encoder_Data[35]	1528
Encoder_Data[36]	1615
Encoder_Data[37]	1573
Encoder_Data[38]	1548
Encoder_Data[39]	1506
Encoder_Data[40]	1593
Encoder_Data[41]	1569
Encoder_Data[42]	1546
Encoder_Data[43]	1503
Encoder_Data[44]	1591
Encoder_Data[45]	1548
Encoder_Data[46]	1562
Encoder_Data[47]	1556
Encoder_Data[48]	1514
Encoder_Data[49]	1490
Encoder_Data[50]	1596
Encoder_Data[51]	1535
Encoder_Data[52]	1511
Encoder_Data[53]	1543
Encoder_Data[54]	1556
Encoder_Data[55]	1589
Encoder_Data[56]	1546
Encoder_Data[57]	1540
Encoder_Data[58]	1516
Encoder_Data[59]	1567
Encoder_Data[60]	1562

Encoder_Data[61]	1519
Encoder_Data[62]	1496
Encoder_Data[63]	1583
Encoder_Data[64]	1522
Encoder_Data[65]	1479
Encoder_Data[66]	1493
Encoder_Data[67]	1543
Encoder_Data[68]	1538
Encoder_Data[69]	1533
Encoder_Data[70]	1490
Encoder_Data[71]	1485
Encoder_Data[72]	1461
Encoder_Data[73]	1549
Encoder_Data[74]	1543
Encoder_Data[75]	1575
Encoder_Data[76]	1514
Encoder_Data[77]	1546
Encoder_Data[78]	1521
Encoder_Data[79]	1515
Encoder_Data[80]	1454
Encoder_Data[81]	1542
Encoder_Data[82]	1480
Encoder_Data[83]	1512
Encoder_Data[84]	1488
Encoder_Data[85]	1464
Encoder_Data[86]	1440
Encoder_Data[87]	1472
Encoder_Data[88]	1485
Encoder_Data[89]	1480
Encoder_Data[90]	1474
Encoder_Data[91]	1524
Encoder_Data[92]	1537
Encoder_Data[93]	1513
Encoder_Data[94]	1470
Encoder_Data[95]	1445
Encoder_Data[96]	1533
Encoder_Data[97]	1509
Encoder_Data[98]	1484
Encoder_Data[99]	1478
Encoder_Data[100]	1529
Encoder_Data[101]	1504

Encoder_Data[102]	1517
Encoder_Data[103]	1455
Encoder_Data[104]	1468
Encoder_Data[105]	1443
Encoder_Data[106]	1512
Encoder_Data[107]	1505
Encoder_Data[108]	1500
Encoder_Data[109]	1438
Encoder_Data[110]	1451
Encoder_Data[111]	1388
Encoder_Data[112]	1458
Encoder_Data[113]	1470
Encoder_Data[114]	1464
Encoder_Data[115]	1421
Encoder_Data[116]	1508
Encoder_Data[117]	1502
Encoder_Data[118]	1459
Encoder_Data[119]	1415
Encoder_Data[120]	1390
Encoder_Data[121]	1459
Encoder_Data[122]	1434
Encoder_Data[123]	1465
Encoder_Data[124]	1402
Encoder_Data[125]	1452
Encoder_Data[126]	1464
Encoder_Data[127]	1477
Encoder_Data[128]	1395
Encoder_Data[129]	1464
Encoder_Data[130]	1401
Encoder_Data[131]	1413
Encoder_Data[132]	1388
Encoder_Data[133]	1381
Encoder_Data[134]	1431
Encoder_Data[135]	1444
Encoder_Data[136]	1456
Encoder_Data[137]	1430
Encoder_Data[138]	1424
Encoder_Data[139]	1379
Encoder_Data[140]	1410
Encoder_Data[141]	1328
Encoder_Data[142]	1341

Encoder_Data[143]	1429
Encoder_Data[144]	1403
Encoder_Data[145]	1378
Encoder_Data[146]	1409
Encoder_Data[147]	1403
Encoder_Data[148]	1339
Encoder_Data[149]	1371
Encoder_Data[150]	1345
Encoder_Data[151]	1358
Encoder_Data[152]	1332
Encoder_Data[153]	1307
Encoder_Data[154]	1395
Encoder_Data[155]	1389
Encoder_Data[156]	1344
Encoder_Data[157]	1318
Encoder_Data[158]	1368
Encoder_Data[159]	1380
Encoder_Data[160]	1317
Encoder_Data[161]	1348
Encoder_Data[162]	1246
Encoder_Data[163]	1391
Encoder_Data[164]	1346
Encoder_Data[165]	1321
Encoder_Data[166]	1314
Encoder_Data[167]	1288
Encoder_Data[168]	1263
Encoder_Data[169]	1370
Encoder_Data[170]	1306
Encoder_Data[171]	1280
Encoder_Data[172]	1293
Encoder_Data[173]	1343
Encoder_Data[174]	1298
Encoder_Data[175]	1330
Encoder_Data[176]	1266
Encoder_Data[177]	1278
Encoder_Data[178]	1329
Encoder_Data[179]	1265
Encoder_Data[180]	1258
Encoder_Data[181]	1233
Encoder_Data[182]	1321
Encoder_Data[183]	1295

Encoder_Data[184]	1308
Encoder_Data[185]	1244
Encoder_Data[186]	1218
Encoder_Data[187]	1306
Encoder_Data[188]	1299
Encoder_Data[189]	1254
Encoder_Data[190]	1248
Encoder_Data[191]	1317
Encoder_Data[192]	1234
Encoder_Data[193]	1246
Encoder_Data[194]	1258
Encoder_Data[195]	1213
Encoder_Data[196]	1263
Encoder_Data[197]	1294
Encoder_Data[198]	1193
Encoder_Data[199]	1243
Encoder_Data[200]	1180
Encoder_Data[201]	1268
Encoder_Data[202]	1223
Encoder_Data[203]	1178
Encoder_Data[204]	1191
Encoder_Data[205]	1241
Encoder_Data[206]	1272
Encoder_Data[207]	1209
Encoder_Data[208]	1221
Encoder_Data[209]	1158
Encoder_Data[210]	1209
Encoder_Data[211]	1221
Encoder_Data[212]	1214
Encoder_Data[213]	1133
Encoder_Data[214]	1164
Encoder_Data[215]	1120
Encoder_Data[216]	1245
Encoder_Data[217]	1238
Encoder_Data[218]	1232
Encoder_Data[219]	1149
Encoder_Data[220]	1124
Encoder_Data[221]	1099
Encoder_Data[222]	1204
Encoder_Data[223]	1216
Encoder_Data[224]	1153

Encoder_Data[225]	1146
Encoder_Data[226]	1177
Encoder_Data[227]	1189
Encoder_Data[228]	1201
Encoder_Data[229]	1176
Encoder_Data[230]	1150
Encoder_Data[231]	1125
Encoder_Data[232]	1213
Encoder_Data[233]	1150
Encoder_Data[234]	1162
Encoder_Data[235]	1100
Encoder_Data[236]	1130
Encoder_Data[237]	1180
Encoder_Data[238]	1100
Encoder_Data[239]	1093
Encoder_Data[240]	1180
Encoder_Data[241]	1173
Encoder_Data[242]	1166
Encoder_Data[243]	1122
Encoder_Data[244]	1097
Encoder_Data[245]	1073
Encoder_Data[246]	1139
Encoder_Data[247]	1132
Encoder_Data[248]	1144
Encoder_Data[249]	1082
Encoder_Data[250]	1041
Encoder_Data[251]	1124
Encoder_Data[252]	1154
Encoder_Data[253]	1111
Encoder_Data[254]	1123
Encoder_Data[255]	1062
Encoder_Data[256]	1074
Encoder_Data[257]	1122
Encoder_Data[258]	1079
Encoder_Data[259]	1072
Encoder_Data[260]	1085
Encoder_Data[261]	1007
Encoder_Data[262]	1053
Encoder_Data[263]	1101
Encoder_Data[264]	1077
Encoder_Data[265]	1070

Encoder_Data[266]	1029
Encoder_Data[267]	1022
Encoder_Data[268]	998
Encoder_Data[269]	1063
Encoder_Data[270]	1057
Encoder_Data[271]	1033
Encoder_Data[272]	1026
Encoder_Data[273]	1074
Encoder_Data[274]	1067
Encoder_Data[275]	1043
Encoder_Data[276]	1002
Encoder_Data[277]	977
Encoder_Data[278]	1042
Encoder_Data[279]	1018
Encoder_Data[280]	994
Encoder_Data[281]	1023
Encoder_Data[282]	944
Encoder_Data[283]	956
Encoder_Data[284]	1022
Encoder_Data[285]	1016
Encoder_Data[286]	973
Encoder_Data[287]	948
Encoder_Data[288]	960
Encoder_Data[289]	1026
Encoder_Data[290]	1002
Encoder_Data[291]	978
Encoder_Data[292]	1025
Encoder_Data[293]	1019
Encoder_Data[294]	994
Encoder_Data[295]	1006
Encoder_Data[296]	964
Encoder_Data[297]	1012
Encoder_Data[298]	1023
Encoder_Data[299]	981
Encoder_Data[300]	993
Encoder_Data[301]	912
Encoder_Data[302]	962
Encoder_Data[303]	1010
Encoder_Data[304]	930
Encoder_Data[305]	960
Encoder_Data[306]	972

Encoder_Data[307]	984
Encoder_Data[308]	941
Encoder_Data[309]	971
Encoder_Data[310]	947
Encoder_Data[311]	995
Encoder_Data[312]	970
Encoder_Data[313]	964
Encoder_Data[314]	939
Encoder_Data[315]	969
Encoder_Data[316]	869
Encoder_Data[317]	975
Encoder_Data[318]	894
Encoder_Data[319]	944
Encoder_Data[320]	975
Encoder_Data[321]	950
Encoder_Data[322]	980
Encoder_Data[323]	956
Encoder_Data[324]	931
Encoder_Data[325]	886
Encoder_Data[326]	879
Encoder_Data[327]	967
Encoder_Data[328]	961
Encoder_Data[329]	955
Encoder_Data[330]	892
Encoder_Data[331]	905
Encoder_Data[332]	898
Encoder_Data[333]	930
Encoder_Data[334]	942
Encoder_Data[335]	859
Encoder_Data[336]	795
Encoder_Data[337]	942
Encoder_Data[338]	917
Encoder_Data[339]	892
Encoder_Data[340]	886
Encoder_Data[341]	918
Encoder_Data[342]	911
Encoder_Data[343]	924
Encoder_Data[344]	899
Encoder_Data[345]	892
Encoder_Data[346]	789
Encoder_Data[347]	899

Encoder_Data[348]	931
Encoder_Data[349]	924
Encoder_Data[350]	860
Encoder_Data[351]	873
Encoder_Data[352]	828
Encoder_Data[353]	918
Encoder_Data[354]	815
Encoder_Data[355]	906
Encoder_Data[356]	861
Encoder_Data[357]	855
Encoder_Data[358]	809
Encoder_Data[359]	881
Encoder_Data[360]	778
Encoder_Data[361]	888
Encoder_Data[362]	823
Encoder_Data[363]	798
Encoder_Data[364]	869
Encoder_Data[365]	862
Encoder_Data[366]	817
Encoder_Data[367]	849
Encoder_Data[368]	823
Encoder_Data[369]	817
Encoder_Data[370]	811
Encoder_Data[371]	843
Encoder_Data[372]	817
Encoder_Data[373]	831
Encoder_Data[374]	825
Encoder_Data[375]	760
Encoder_Data[376]	812
Encoder_Data[377]	864
Encoder_Data[378]	839
Encoder_Data[379]	794
Encoder_Data[380]	730
Encoder_Data[381]	802
Encoder_Data[382]	835
Encoder_Data[383]	771
Encoder_Data[384]	785
Encoder_Data[385]	721
Encoder_Data[386]	753
Encoder_Data[387]	825
Encoder_Data[388]	800

Encoder_Data[389]	775
Encoder_Data[390]	769
Encoder_Data[391]	821
Encoder_Data[392]	816
Encoder_Data[393]	791
Encoder_Data[394]	747
Encoder_Data[395]	741
Encoder_Data[396]	794
Encoder_Data[397]	788
Encoder_Data[398]	783
Encoder_Data[399]	797
Encoder_Data[400]	734
Encoder_Data[401]	806
Encoder_Data[402]	762
Encoder_Data[403]	738
Encoder_Data[404]	790
Encoder_Data[405]	804
Encoder_Data[406]	800
Encoder_Data[407]	775
Encoder_Data[408]	750
Encoder_Data[409]	707
Encoder_Data[410]	703
Encoder_Data[411]	793
Encoder_Data[412]	788
Encoder_Data[413]	764
Encoder_Data[414]	759
Encoder_Data[415]	716
Encoder_Data[416]	787
Encoder_Data[417]	687
Encoder_Data[418]	797
Encoder_Data[419]	753
Encoder_Data[420]	673
Encoder_Data[421]	802
Encoder_Data[422]	816
Encoder_Data[423]	754
Encoder_Data[424]	749
Encoder_Data[425]	688
Encoder_Data[426]	798
Encoder_Data[427]	773
Encoder_Data[428]	712
Encoder_Data[429]	784

Encoder_Data[430]	760
Encoder_Data[431]	774
Encoder_Data[432]	750
Encoder_Data[433]	727
Encoder_Data[434]	760
Encoder_Data[435]	794
Encoder_Data[436]	770
Encoder_Data[437]	746
Encoder_Data[438]	704
Encoder_Data[439]	795
Encoder_Data[440]	733
Encoder_Data[441]	767
Encoder_Data[442]	743
Encoder_Data[443]	739
Encoder_Data[444]	660
Encoder_Data[445]	788
Encoder_Data[446]	745
Encoder_Data[447]	760
Encoder_Data[448]	737
Encoder_Data[449]	751
Encoder_Data[450]	766
Encoder_Data[451]	724
Encoder_Data[452]	739
Encoder_Data[453]	734
Encoder_Data[454]	807
Encoder_Data[455]	784
Encoder_Data[456]	741
Encoder_Data[457]	737
Encoder_Data[458]	752
Encoder_Data[459]	748
Encoder_Data[460]	744
Encoder_Data[461]	702
Encoder_Data[462]	716
Encoder_Data[463]	808
Encoder_Data[464]	765
Encoder_Data[465]	741
Encoder_Data[466]	795
Encoder_Data[467]	790
Encoder_Data[468]	767
Encoder_Data[469]	724
Encoder_Data[470]	701

Encoder_Data[471]	773
Encoder_Data[472]	730
Encoder_Data[473]	670
Encoder_Data[474]	779
Encoder_Data[475]	755
Encoder_Data[476]	751
Encoder_Data[477]	708
Encoder_Data[478]	686
Encoder_Data[479]	700
Encoder_Data[480]	771
Encoder_Data[481]	767
Encoder_Data[482]	762
Encoder_Data[483]	701
Encoder_Data[484]	697
Encoder_Data[485]	788
Encoder_Data[486]	764
Encoder_Data[487]	721
Encoder_Data[488]	697
Encoder_Data[489]	693
Encoder_Data[490]	764
Encoder_Data[491]	759
Encoder_Data[492]	755
Encoder_Data[493]	712
Encoder_Data[494]	688
Encoder_Data[495]	665
Encoder_Data[496]	754
Encoder_Data[497]	731
Encoder_Data[498]	726
Encoder_Data[499]	665
Encoder_Data[500]	661
Encoder_Data[501]	750
Encoder_Data[502]	727
Encoder_Data[503]	703
Encoder_Data[504]	699
Encoder_Data[505]	694
Encoder_Data[506]	746
Encoder_Data[507]	741
Encoder_Data[508]	699
Encoder_Data[509]	675
Encoder_Data[510]	765
Encoder_Data[511]	741

Encoder_Data[512]	755
Encoder_Data[513]	712
Encoder_Data[514]	670
Encoder_Data[515]	684
Encoder_Data[516]	699
Encoder_Data[517]	713
Encoder_Data[518]	671
Encoder_Data[519]	685
Encoder_Data[520]	718
Encoder_Data[521]	751
Encoder_Data[522]	671
Encoder_Data[523]	685
Encoder_Data[524]	756
Encoder_Data[525]	732
Encoder_Data[526]	765
Encoder_Data[527]	703
Encoder_Data[528]	755
Encoder_Data[529]	712
Encoder_Data[530]	745
Encoder_Data[531]	703
Encoder_Data[532]	698
Encoder_Data[533]	693
Encoder_Data[534]	726
Encoder_Data[535]	721
Encoder_Data[536]	736
Encoder_Data[537]	675
Encoder_Data[538]	689
Encoder_Data[539]	722
Encoder_Data[540]	737
Encoder_Data[541]	695
Encoder_Data[542]	691
Encoder_Data[543]	686
Encoder_Data[544]	683
Encoder_Data[545]	734
Encoder_Data[546]	730
Encoder_Data[547]	707
Encoder_Data[548]	684
Encoder_Data[549]	717
Encoder_Data[550]	750
Encoder_Data[551]	746
Encoder_Data[552]	722

Encoder_Data[553]	680
Encoder_Data[554]	658
Encoder_Data[555]	654
Encoder_Data[556]	705
Encoder_Data[557]	739
Encoder_Data[558]	678
Encoder_Data[559]	637
Encoder_Data[560]	764
Encoder_Data[561]	741
Encoder_Data[562]	756
Encoder_Data[563]	714
Encoder_Data[564]	692
Encoder_Data[565]	669
Encoder_Data[566]	647
Encoder_Data[567]	717
Encoder_Data[568]	732
Encoder_Data[569]	691
Encoder_Data[570]	725
Encoder_Data[571]	721
Encoder_Data[572]	755
Encoder_Data[573]	713
Encoder_Data[574]	673
Encoder_Data[575]	669
Encoder_Data[576]	740
Encoder_Data[577]	717
Encoder_Data[578]	714
Encoder_Data[579]	691
Encoder_Data[580]	744
Encoder_Data[581]	722
Encoder_Data[582]	718
Encoder_Data[583]	696
Encoder_Data[584]	693
Encoder_Data[585]	727
Encoder_Data[586]	743
Encoder_Data[587]	759
Encoder_Data[588]	662
Encoder_Data[589]	696
Encoder_Data[590]	769
Encoder_Data[591]	746
Encoder_Data[592]	763
Encoder_Data[593]	721

Encoder_Data[594]	718
Encoder_Data[595]	678
Encoder_Data[596]	675
Encoder_Data[597]	747
Encoder_Data[598]	725
Encoder_Data[599]	741
Encoder_Data[600]	663
Encoder_Data[601]	698
Encoder_Data[602]	732
Encoder_Data[603]	730
Encoder_Data[604]	689
Encoder_Data[605]	705
Encoder_Data[606]	684
Encoder_Data[607]	776
Encoder_Data[608]	716
Encoder_Data[609]	695
Encoder_Data[610]	692
Encoder_Data[611]	671
Encoder_Data[612]	762
Encoder_Data[613]	722
Encoder_Data[614]	738
Encoder_Data[615]	698
Encoder_Data[616]	714
Encoder_Data[617]	768
Encoder_Data[618]	747
Encoder_Data[619]	669
Encoder_Data[620]	704
Encoder_Data[621]	702
Encoder_Data[622]	756
Encoder_Data[623]	734
Encoder_Data[624]	751
Encoder_Data[625]	749
Encoder_Data[626]	728
Encoder_Data[627]	764
Encoder_Data[628]	762
Encoder_Data[629]	759
Encoder_Data[630]	738
Encoder_Data[631]	698
Encoder_Data[632]	733
Encoder_Data[633]	788
Encoder_Data[634]	805

Encoder_Data[635]	764
Encoder_Data[636]	781
Encoder_Data[637]	779
Encoder_Data[638]	795
Encoder_Data[639]	774
Encoder_Data[640]	752
Encoder_Data[641]	788
Encoder_Data[642]	729
Encoder_Data[643]	823
Encoder_Data[644]	820
Encoder_Data[645]	799
Encoder_Data[646]	797
Encoder_Data[647]	853
Encoder_Data[648]	812
Encoder_Data[649]	830
Encoder_Data[650]	847
Encoder_Data[651]	826
Encoder_Data[652]	785
Encoder_Data[653]	841
Encoder_Data[654]	878
Encoder_Data[655]	837
Encoder_Data[656]	835
Encoder_Data[657]	814
Encoder_Data[658]	811
Encoder_Data[659]	810
Encoder_Data[660]	866
Encoder_Data[661]	806
Encoder_Data[662]	842
Encoder_Data[663]	763
Encoder_Data[664]	838
Encoder_Data[665]	914
Encoder_Data[666]	874
Encoder_Data[667]	891
Encoder_Data[668]	870
Encoder_Data[669]	829
Encoder_Data[670]	867
Encoder_Data[671]	884
Encoder_Data[672]	863
Encoder_Data[673]	842
Encoder_Data[674]	841
Encoder_Data[675]	801

Encoder_Data[676]	915
Encoder_Data[677]	875
Encoder_Data[678]	874
Encoder_Data[679]	853
Encoder_Data[680]	851
Encoder_Data[681]	888
Encoder_Data[682]	906
Encoder_Data[683]	905
Encoder_Data[684]	845
Encoder_Data[685]	863
Encoder_Data[686]	938
Encoder_Data[687]	937
Encoder_Data[688]	916
Encoder_Data[689]	915
Encoder_Data[690]	932
Encoder_Data[691]	949
Encoder_Data[692]	929
Encoder_Data[693]	965
Encoder_Data[694]	907
Encoder_Data[695]	906
Encoder_Data[696]	923
Encoder_Data[697]	941
Encoder_Data[698]	939
Encoder_Data[699]	974
Encoder_Data[700]	898
Encoder_Data[701]	990
Encoder_Data[702]	988
Encoder_Data[703]	931
Encoder_Data[704]	929
Encoder_Data[705]	983
Encoder_Data[706]	945
Encoder_Data[707]	943
Encoder_Data[708]	978
Encoder_Data[709]	940
Encoder_Data[710]	976
Encoder_Data[711]	974
Encoder_Data[712]	972
Encoder_Data[713]	952
Encoder_Data[714]	931
Encoder_Data[715]	1003
Encoder_Data[716]	965

Encoder_Data[717]	945
Encoder_Data[718]	943
Encoder_Data[719]	923
Encoder_Data[720]	902
Encoder_Data[721]	900
Encoder_Data[722]	917
Encoder_Data[723]	934
Encoder_Data[724]	875
Encoder_Data[725]	893
Encoder_Data[726]	985
Encoder_Data[727]	965
Encoder_Data[728]	964
Encoder_Data[729]	944
Encoder_Data[730]	980
Encoder_Data[731]	1014
Encoder_Data[732]	977
Encoder_Data[733]	939
Encoder_Data[734]	975
Encoder_Data[735]	991
Encoder_Data[736]	990
Encoder_Data[737]	915
Encoder_Data[738]	1041
Encoder_Data[739]	1021
Encoder_Data[740]	1037
Encoder_Data[741]	964
Encoder_Data[742]	943
Encoder_Data[743]	960
Encoder_Data[744]	1030
Encoder_Data[745]	1029
Encoder_Data[746]	1027
Encoder_Data[747]	1007
Encoder_Data[748]	1094
Encoder_Data[749]	1074
Encoder_Data[750]	1055
Encoder_Data[751]	1072
Encoder_Data[752]	1016
Encoder_Data[753]	1032
Encoder_Data[754]	1084
Encoder_Data[755]	1082
Encoder_Data[756]	1081
Encoder_Data[757]	1043

Encoder_Data[758]	1006
Encoder_Data[759]	1021
Encoder_Data[760]	1072
Encoder_Data[761]	1070
Encoder_Data[762]	979
Encoder_Data[763]	1030
Encoder_Data[764]	1099
Encoder_Data[765]	1079
Encoder_Data[766]	1095
Encoder_Data[767]	1057
Encoder_Data[768]	1108
Encoder_Data[769]	1071
Encoder_Data[770]	1104
Encoder_Data[771]	1102
Encoder_Data[772]	1082
Encoder_Data[773]	1045
Encoder_Data[774]	1115
Encoder_Data[775]	1113
Encoder_Data[776]	1074
Encoder_Data[777]	1072
Encoder_Data[778]	1124
Encoder_Data[779]	1122
Encoder_Data[780]	1120
Encoder_Data[781]	1081
Encoder_Data[782]	1079
Encoder_Data[783]	1042
Encoder_Data[784]	1148
Encoder_Data[785]	1164
Encoder_Data[786]	1143
Encoder_Data[787]	1123
Encoder_Data[788]	1084
Encoder_Data[789]	1136
Encoder_Data[790]	1152
Encoder_Data[791]	1169
Encoder_Data[792]	1129
Encoder_Data[793]	1073
Encoder_Data[794]	1125
Encoder_Data[795]	1160
Encoder_Data[796]	1157
Encoder_Data[797]	1155
Encoder_Data[798]	1134

Encoder_Data[799]	1168
Encoder_Data[800]	1148
Encoder_Data[801]	1164
Encoder_Data[802]	1106
Encoder_Data[803]	1122
Encoder_Data[804]	1156
Encoder_Data[805]	1191
Encoder_Data[806]	1170
Encoder_Data[807]	1167
Encoder_Data[808]	1091
Encoder_Data[809]	1162
Encoder_Data[810]	1141
Encoder_Data[811]	1139
Encoder_Data[812]	1137
Encoder_Data[813]	1116
Encoder_Data[814]	1169
Encoder_Data[815]	1093
Encoder_Data[816]	1164
Encoder_Data[817]	1162
Encoder_Data[818]	1141
Encoder_Data[819]	1102
Encoder_Data[820]	1136
Encoder_Data[821]	1171
Encoder_Data[822]	1187
Encoder_Data[823]	1128
Encoder_Data[824]	1144
Encoder_Data[825]	1216
Encoder_Data[826]	1214
Encoder_Data[827]	1192
Encoder_Data[828]	1190
Encoder_Data[829]	1150
Encoder_Data[830]	1203
Encoder_Data[831]	1220
Encoder_Data[832]	1236
Encoder_Data[833]	1158
Encoder_Data[834]	1155
Encoder_Data[835]	1097
Encoder_Data[836]	1225
Encoder_Data[837]	1222
Encoder_Data[838]	1162
Encoder_Data[839]	1160

Encoder_Data[840]	1232
Encoder_Data[841]	1249
Encoder_Data[842]	1170
Encoder_Data[843]	1167
Encoder_Data[844]	1203
Encoder_Data[845]	1238
Encoder_Data[846]	1216
Encoder_Data[847]	1233
Encoder_Data[848]	1212
Encoder_Data[849]	1209
Encoder_Data[850]	1264
Encoder_Data[851]	1242
Encoder_Data[852]	1220
Encoder_Data[853]	1199
Encoder_Data[854]	1273
Encoder_Data[855]	1233
Encoder_Data[856]	1230
Encoder_Data[857]	1189
Encoder_Data[858]	1263
Encoder_Data[859]	1241
Encoder_Data[860]	1239
Encoder_Data[861]	1217
Encoder_Data[862]	1234
Encoder_Data[863]	1175
Encoder_Data[864]	1210
Encoder_Data[865]	1264
Encoder_Data[866]	1262
Encoder_Data[867]	1221
Encoder_Data[868]	1219
Encoder_Data[869]	1273
Encoder_Data[870]	1252
Encoder_Data[871]	1249
Encoder_Data[872]	1228
Encoder_Data[873]	1282
Encoder_Data[874]	1260
Encoder_Data[875]	1277
Encoder_Data[876]	1255
Encoder_Data[877]	1253
Encoder_Data[878]	1231
Encoder_Data[879]	1304
Encoder_Data[880]	1244

Encoder_Data[881]	1261
Encoder_Data[882]	1202
Encoder_Data[883]	1237
Encoder_Data[884]	1253
Encoder_Data[885]	1288
Encoder_Data[886]	1305
Encoder_Data[887]	1283
Encoder_Data[888]	1242
Encoder_Data[889]	1296
Encoder_Data[890]	1256
Encoder_Data[891]	1291
Encoder_Data[892]	1288
Encoder_Data[893]	1247
Encoder_Data[894]	1301
Encoder_Data[895]	1317
Encoder_Data[896]	1315
Encoder_Data[897]	1255
Encoder_Data[898]	1271
Encoder_Data[899]	1307
Encoder_Data[900]	1342
Encoder_Data[901]	1264
Encoder_Data[902]	1299
Encoder_Data[903]	1277
Encoder_Data[904]	1255
Encoder_Data[905]	1214
Encoder_Data[906]	1325
Encoder_Data[907]	1341
Encoder_Data[908]	1263
Encoder_Data[909]	1279
Encoder_Data[910]	1276
Encoder_Data[911]	1254
Encoder_Data[912]	1309
Encoder_Data[913]	1325
Encoder_Data[914]	1304
Encoder_Data[915]	1282
Encoder_Data[916]	1299
Encoder_Data[917]	1315
Encoder_Data[918]	1351
Encoder_Data[919]	1311
Encoder_Data[920]	1308
Encoder_Data[921]	1343

Encoder_Data[922]	1321
Encoder_Data[923]	1280
Encoder_Data[924]	1259
Encoder_Data[925]	1294
Encoder_Data[926]	1272
Encoder_Data[927]	1327
Encoder_Data[928]	1324
Encoder_Data[929]	1303
Encoder_Data[930]	1300
Encoder_Data[931]	1336
Encoder_Data[932]	1371
Encoder_Data[933]	1293
Encoder_Data[934]	1347
Encoder_Data[935]	1325
Encoder_Data[936]	1247
Encoder_Data[937]	1339
Encoder_Data[938]	1356
Encoder_Data[939]	1334
Encoder_Data[940]	1294
Encoder_Data[941]	1310
Encoder_Data[942]	1345
Encoder_Data[943]	1324
Encoder_Data[944]	1321
Encoder_Data[945]	1357
Encoder_Data[946]	1335
Encoder_Data[947]	1352
Encoder_Data[948]	1330
Encoder_Data[949]	1328
Encoder_Data[950]	1344
Encoder_Data[951]	1323
Encoder_Data[952]	1282
Encoder_Data[953]	1298
Encoder_Data[954]	1371
Encoder_Data[955]	1350
Encoder_Data[956]	1309
Encoder_Data[957]	1306
Encoder_Data[958]	1322
Encoder_Data[959]	1395
Encoder_Data[960]	1336
Encoder_Data[961]	1352
Encoder_Data[962]	1292

Encoder_Data[963]	1404
Encoder_Data[964]	1401
Encoder_Data[965]	1360
Encoder_Data[966]	1320
Encoder_Data[967]	1336
Encoder_Data[968]	1390
Encoder_Data[969]	1388
Encoder_Data[970]	1404
Encoder_Data[971]	1344
Encoder_Data[972]	1360
Encoder_Data[973]	1396
Encoder_Data[974]	1392
Encoder_Data[975]	1351
Encoder_Data[976]	1349
Encoder_Data[977]	1327
Encoder_Data[978]	1437
Encoder_Data[979]	1416
Encoder_Data[980]	1450
Encoder_Data[981]	1353
Encoder_Data[982]	1388
Encoder_Data[983]	1309
Encoder_Data[984]	1438
Encoder_Data[985]	1397
Encoder_Data[986]	1394
Encoder_Data[987]	1410
Encoder_Data[988]	1445
Encoder_Data[989]	1403
Encoder_Data[990]	1419
Encoder_Data[991]	1434
Encoder_Data[992]	1356
Encoder_Data[993]	1390
Encoder_Data[994]	1388
Encoder_Data[995]	1460
Encoder_Data[996]	1419
Encoder_Data[997]	1416
Encoder_Data[998]	1451
Encoder_Data[999]	1448

C. Fluke Metering Test Data

Encoder_Data	<normal>	{...}	Column1
Encoder_Data[0]		2876	99
Encoder_Data[1]		2975	-42
Encoder_Data[2]		2933	-7
Encoder_Data[3]		2926	-6
Encoder_Data[4]		2920	-7
Encoder_Data[5]		2913	-41
Encoder_Data[6]		2872	46
Encoder_Data[7]		2918	-42
Encoder_Data[8]		2876	29
Encoder_Data[9]		2905	29
Encoder_Data[10]		2934	-7
Encoder_Data[11]		2927	-76
Encoder_Data[12]		2851	11
Encoder_Data[13]		2862	81
Encoder_Data[14]		2943	-6
Encoder_Data[15]		2937	-24
Encoder_Data[16]		2913	-24
Encoder_Data[17]		2889	-24
Encoder_Data[18]		2865	82
Encoder_Data[19]		2947	-60
Encoder_Data[20]		2887	-24
Encoder_Data[21]		2863	-41
Encoder_Data[22]		2822	98
Encoder_Data[23]		2920	-59
Encoder_Data[24]		2861	11
Encoder_Data[25]		2872	-24
Encoder_Data[26]		2848	64
Encoder_Data[27]		2912	-6
Encoder_Data[28]		2906	-76
Encoder_Data[29]		2830	-7
Encoder_Data[30]		2823	-24
Encoder_Data[31]		2799	116
Encoder_Data[32]		2915	-58
Encoder_Data[33]		2857	11
Encoder_Data[34]		2868	-24
Encoder_Data[35]		2844	-23
Encoder_Data[36]		2821	45
Encoder_Data[37]		2866	-6

Encoder_Data[38]	2860	-59
Encoder_Data[39]	2801	29
Encoder_Data[40]	2830	45
Encoder_Data[41]	2875	-23
Encoder_Data[42]	2852	10
Encoder_Data[43]	2862	-23
Encoder_Data[44]	2839	-24
Encoder_Data[45]	2815	-24
Encoder_Data[46]	2791	46
Encoder_Data[47]	2837	-24
Encoder_Data[48]	2813	46
Encoder_Data[49]	2859	-94
Encoder_Data[50]	2765	29
Encoder_Data[51]	2794	-24
Encoder_Data[52]	2770	63
Encoder_Data[53]	2833	12
Encoder_Data[54]	2845	-24
Encoder_Data[55]	2821	-7
Encoder_Data[56]	2814	-42
Encoder_Data[57]	2772	-24
Encoder_Data[58]	2748	64
Encoder_Data[59]	2812	-6
Encoder_Data[60]	2806	11
Encoder_Data[61]	2817	-42
Encoder_Data[62]	2775	29
Encoder_Data[63]	2804	11
Encoder_Data[64]	2815	11
Encoder_Data[65]	2826	-24
Encoder_Data[66]	2802	-42
Encoder_Data[67]	2760	-25
Encoder_Data[68]	2735	-43
Encoder_Data[69]	2692	47
Encoder_Data[70]	2739	-25
Encoder_Data[71]	2714	47
Encoder_Data[72]	2761	29
Encoder_Data[73]	2790	-61
Encoder_Data[74]	2729	-7
Encoder_Data[75]	2722	30
Encoder_Data[76]	2752	11
Encoder_Data[77]	2763	-25
Encoder_Data[78]	2738	29

Encoder_Data[79]	2767	-61
Encoder_Data[80]	2706	11
Encoder_Data[81]	2717	47
Encoder_Data[82]	2764	-25
Encoder_Data[83]	2739	-25
Encoder_Data[84]	2714	-43
Encoder_Data[85]	2671	11
Encoder_Data[86]	2682	64
Encoder_Data[87]	2746	-61
Encoder_Data[88]	2685	-7
Encoder_Data[89]	2678	-43
Encoder_Data[90]	2635	47
Encoder_Data[91]	2682	65
Encoder_Data[92]	2747	-43
Encoder_Data[93]	2704	-26
Encoder_Data[94]	2678	-25
Encoder_Data[95]	2653	-25
Encoder_Data[96]	2628	101
Encoder_Data[97]	2729	-7
Encoder_Data[98]	2722	-25
Encoder_Data[99]	2697	-43
Encoder_Data[100]	2654	-26
Encoder_Data[101]	2628	47
Encoder_Data[102]	2675	-7
Encoder_Data[103]	2668	-26
Encoder_Data[104]	2642	-26
Encoder_Data[105]	2616	-7
Encoder_Data[106]	2609	83
Encoder_Data[107]	2692	-43
Encoder_Data[108]	2649	-8
Encoder_Data[109]	2641	-7
Encoder_Data[110]	2634	-82
Encoder_Data[111]	2552	48
Encoder_Data[112]	2600	48
Encoder_Data[113]	2648	11
Encoder_Data[114]	2659	-63
Encoder_Data[115]	2596	-7
Encoder_Data[116]	2589	48
Encoder_Data[117]	2637	11
Encoder_Data[118]	2648	-44
Encoder_Data[119]	2604	-26

Encoder_Data[120]	2578	-45
Encoder_Data[121]	2533	67
Encoder_Data[122]	2600	10
Encoder_Data[123]	2610	-7
Encoder_Data[124]	2603	-64
Encoder_Data[125]	2539	-8
Encoder_Data[126]	2531	11
Encoder_Data[127]	2542	30
Encoder_Data[128]	2572	11
Encoder_Data[129]	2583	-45
Encoder_Data[130]	2538	-9
Encoder_Data[131]	2529	49
Encoder_Data[132]	2578	-8
Encoder_Data[133]	2570	11
Encoder_Data[134]	2581	-45
Encoder_Data[135]	2536	-27
Encoder_Data[136]	2509	11
Encoder_Data[137]	2520	30
Encoder_Data[138]	2550	-27
Encoder_Data[139]	2523	-7
Encoder_Data[140]	2516	-66
Encoder_Data[141]	2450	-7
Encoder_Data[142]	2443	105
Encoder_Data[143]	2548	-27
Encoder_Data[144]	2521	-26
Encoder_Data[145]	2495	-27
Encoder_Data[146]	2468	29
Encoder_Data[147]	2497	11
Encoder_Data[148]	2508	-46
Encoder_Data[149]	2462	49
Encoder_Data[150]	2511	-84
Encoder_Data[151]	2427	30
Encoder_Data[152]	2457	29
Encoder_Data[153]	2486	11
Encoder_Data[154]	2497	10
Encoder_Data[155]	2507	-8
Encoder_Data[156]	2499	-46
Encoder_Data[157]	2453	-27
Encoder_Data[158]	2426	29
Encoder_Data[159]	2455	30
Encoder_Data[160]	2485	-27

Encoder_Data[161]	2458	-84
Encoder_Data[162]	2374	11
Encoder_Data[163]	2385	-47
Encoder_Data[164]	2338	106
Encoder_Data[165]	2444	-27
Encoder_Data[166]	2417	-27
Encoder_Data[167]	2390	29
Encoder_Data[168]	2419	11
Encoder_Data[169]	2430	-9
Encoder_Data[170]	2421	11
Encoder_Data[171]	2432	-104
Encoder_Data[172]	2328	30
Encoder_Data[173]	2358	29
Encoder_Data[174]	2387	-27
Encoder_Data[175]	2360	29
Encoder_Data[176]	2389	10
Encoder_Data[177]	2399	-65
Encoder_Data[178]	2334	-27
Encoder_Data[179]	2307	67
Encoder_Data[180]	2374	-27
Encoder_Data[181]	2347	-9
Encoder_Data[182]	2338	10
Encoder_Data[183]	2348	-9
Encoder_Data[184]	2339	-46
Encoder_Data[185]	2293	30
Encoder_Data[186]	2323	29
Encoder_Data[187]	2352	-84
Encoder_Data[188]	2268	29
Encoder_Data[189]	2297	-27
Encoder_Data[190]	2270	-27
Encoder_Data[191]	2243	67
Encoder_Data[192]	2310	-65
Encoder_Data[193]	2245	29
Encoder_Data[194]	2274	-64
Encoder_Data[195]	2210	10
Encoder_Data[196]	2220	86
Encoder_Data[197]	2306	-46
Encoder_Data[198]	2260	-27
Encoder_Data[199]	2233	-8
Encoder_Data[200]	2225	-45
Encoder_Data[201]	2180	84

Encoder_Data[202]	2264	-100
Encoder_Data[203]	2164	84
Encoder_Data[204]	2248	-27
Encoder_Data[205]	2221	10
Encoder_Data[206]	2231	-45
Encoder_Data[207]	2186	85
Encoder_Data[208]	2271	-64
Encoder_Data[209]	2207	-27
Encoder_Data[210]	2180	10
Encoder_Data[211]	2190	-26
Encoder_Data[212]	2164	10
Encoder_Data[213]	2174	10
Encoder_Data[214]	2184	29
Encoder_Data[215]	2213	-45
Encoder_Data[216]	2168	-62
Encoder_Data[217]	2106	102
Encoder_Data[218]	2208	-8
Encoder_Data[219]	2200	-8
Encoder_Data[220]	2192	-63
Encoder_Data[221]	2129	-8
Encoder_Data[222]	2121	-8
Encoder_Data[223]	2113	102
Encoder_Data[224]	2215	-27
Encoder_Data[225]	2188	-8
Encoder_Data[226]	2180	-26
Encoder_Data[227]	2154	-63
Encoder_Data[228]	2091	102
Encoder_Data[229]	2193	-45
Encoder_Data[230]	2148	10
Encoder_Data[231]	2158	-26
Encoder_Data[232]	2132	-44
Encoder_Data[233]	2088	-9
Encoder_Data[234]	2079	-8
Encoder_Data[235]	2071	83
Encoder_Data[236]	2154	-44
Encoder_Data[237]	2110	-8
Encoder_Data[238]	2102	-44
Encoder_Data[239]	2058	46
Encoder_Data[240]	2104	29
Encoder_Data[241]	2133	-8
Encoder_Data[242]	2125	-44

Encoder_Data[243]	2081	-26
Encoder_Data[244]	2055	29
Encoder_Data[245]	2084	10
Encoder_Data[246]	2094	10
Encoder_Data[247]	2104	-62
Encoder_Data[248]	2042	65
Encoder_Data[249]	2107	-7
Encoder_Data[250]	2100	-8
Encoder_Data[251]	2092	-26
Encoder_Data[252]	2066	-26
Encoder_Data[253]	2040	47
Encoder_Data[254]	2087	-26
Encoder_Data[255]	2061	-26
Encoder_Data[256]	2035	-8
Encoder_Data[257]	2027	29
Encoder_Data[258]	2056	-45
Encoder_Data[259]	2011	47
Encoder_Data[260]	2058	-26
Encoder_Data[261]	2032	-25
Encoder_Data[262]	2007	-26
Encoder_Data[263]	1981	29
Encoder_Data[264]	2010	11
Encoder_Data[265]	2021	-45
Encoder_Data[266]	1976	-26
Encoder_Data[267]	1950	48
Encoder_Data[268]	1998	29
Encoder_Data[269]	2027	-8
Encoder_Data[270]	2019	-44
Encoder_Data[271]	1975	-62
Encoder_Data[272]	1913	84
Encoder_Data[273]	1997	-8
Encoder_Data[274]	1989	-7
Encoder_Data[275]	1982	-8
Encoder_Data[276]	1974	-63
Encoder_Data[277]	1911	66
Encoder_Data[278]	1977	11
Encoder_Data[279]	1988	-26
Encoder_Data[280]	1962	-44
Encoder_Data[281]	1918	66
Encoder_Data[282]	1984	-136
Encoder_Data[283]	1848	85

Encoder_Data[284]	1933	11
Encoder_Data[285]	1944	-8
Encoder_Data[286]	1936	-43
Encoder_Data[287]	1893	29
Encoder_Data[288]	1922	29
Encoder_Data[289]	1951	-25
Encoder_Data[290]	1926	-44
Encoder_Data[291]	1882	-26
Encoder_Data[292]	1856	-44
Encoder_Data[293]	1812	103
Encoder_Data[294]	1915	-26
Encoder_Data[295]	1889	-63
Encoder_Data[296]	1826	67
Encoder_Data[297]	1893	10
Encoder_Data[298]	1903	30
Encoder_Data[299]	1933	-26
Encoder_Data[300]	1907	-81
Encoder_Data[301]	1826	12
Encoder_Data[302]	1838	-26
Encoder_Data[303]	1812	103
Encoder_Data[304]	1915	-62
Encoder_Data[305]	1853	-26
Encoder_Data[306]	1827	11
Encoder_Data[307]	1838	-25
Encoder_Data[308]	1813	-26
Encoder_Data[309]	1787	67
Encoder_Data[310]	1854	-26
Encoder_Data[311]	1828	-8
Encoder_Data[312]	1820	-25
Encoder_Data[313]	1795	30
Encoder_Data[314]	1825	-7
Encoder_Data[315]	1818	11
Encoder_Data[316]	1829	-81
Encoder_Data[317]	1748	11
Encoder_Data[318]	1759	-26
Encoder_Data[319]	1733	86
Encoder_Data[320]	1819	-25
Encoder_Data[321]	1794	30
Encoder_Data[322]	1824	-63
Encoder_Data[323]	1761	-7
Encoder_Data[324]	1754	30

Encoder_Data[325]	1784	11
Encoder_Data[326]	1795	-6
Encoder_Data[327]	1789	-26
Encoder_Data[328]	1763	-26
Encoder_Data[329]	1737	49
Encoder_Data[330]	1786	12
Encoder_Data[331]	1798	-44
Encoder_Data[332]	1754	67
Encoder_Data[333]	1821	-137
Encoder_Data[334]	1684	86
Encoder_Data[335]	1770	-63
Encoder_Data[336]	1707	49
Encoder_Data[337]	1756	-63
Encoder_Data[338]	1693	124
Encoder_Data[339]	1817	-26
Encoder_Data[340]	1791	-63
Encoder_Data[341]	1728	-7
Encoder_Data[342]	1721	49
Encoder_Data[343]	1770	-7
Encoder_Data[344]	1763	12
Encoder_Data[345]	1775	-63
Encoder_Data[346]	1712	-44
Encoder_Data[347]	1668	68
Encoder_Data[348]	1736	11
Encoder_Data[349]	1747	12
Encoder_Data[350]	1759	-44
Encoder_Data[351]	1715	-26
Encoder_Data[352]	1689	-7
Encoder_Data[353]	1682	87
Encoder_Data[354]	1769	-81
Encoder_Data[355]	1688	48
Encoder_Data[356]	1736	-63
Encoder_Data[357]	1673	86
Encoder_Data[358]	1759	-25
Encoder_Data[359]	1734	-7
Encoder_Data[360]	1727	-25
Encoder_Data[361]	1702	-26
Encoder_Data[362]	1676	-7
Encoder_Data[363]	1669	67
Encoder_Data[364]	1736	-7
Encoder_Data[365]	1729	-44

Encoder_Data[366]	1685	-26
Encoder_Data[367]	1659	49
Encoder_Data[368]	1708	-26
Encoder_Data[369]	1682	12
Encoder_Data[370]	1694	-7
Encoder_Data[371]	1687	-7
Encoder_Data[372]	1680	-45
Encoder_Data[373]	1635	30
Encoder_Data[374]	1665	31
Encoder_Data[375]	1696	-82
Encoder_Data[376]	1614	12
Encoder_Data[377]	1626	48
Encoder_Data[378]	1674	12
Encoder_Data[379]	1686	-63
Encoder_Data[380]	1623	11
Encoder_Data[381]	1634	12
Encoder_Data[382]	1646	-7
Encoder_Data[383]	1639	12
Encoder_Data[384]	1651	-26
Encoder_Data[385]	1625	-8
Encoder_Data[386]	1617	-7
Encoder_Data[387]	1610	-25
Encoder_Data[388]	1585	68
Encoder_Data[389]	1653	-26
Encoder_Data[390]	1627	12
Encoder_Data[391]	1639	-82
Encoder_Data[392]	1557	86
Encoder_Data[393]	1643	-45
Encoder_Data[394]	1598	-26
Encoder_Data[395]	1572	-26
Encoder_Data[396]	1546	68
Encoder_Data[397]	1614	-8
Encoder_Data[398]	1606	-26
Encoder_Data[399]	1580	-7
Encoder_Data[400]	1573	-44
Encoder_Data[401]	1529	67
Encoder_Data[402]	1596	-26
Encoder_Data[403]	1570	12
Encoder_Data[404]	1582	-8
Encoder_Data[405]	1574	-7
Encoder_Data[406]	1567	50

Encoder_Data[407]	1617	-82
Encoder_Data[408]	1535	-45
Encoder_Data[409]	1490	31
Encoder_Data[410]	1521	48
Encoder_Data[411]	1569	-7
Encoder_Data[412]	1562	-7
Encoder_Data[413]	1555	-63
Encoder_Data[414]	1492	11
Encoder_Data[415]	1503	68
Encoder_Data[416]	1571	-8
Encoder_Data[417]	1563	-26
Encoder_Data[418]	1537	-26
Encoder_Data[419]	1511	-45
Encoder_Data[420]	1466	87
Encoder_Data[421]	1553	11
Encoder_Data[422]	1564	-63
Encoder_Data[423]	1501	-27
Encoder_Data[424]	1474	-26
Encoder_Data[425]	1448	68
Encoder_Data[426]	1516	-63
Encoder_Data[427]	1453	-8
Encoder_Data[428]	1445	68
Encoder_Data[429]	1513	-8
Encoder_Data[430]	1505	-64
Encoder_Data[431]	1441	-26
Encoder_Data[432]	1415	-7
Encoder_Data[433]	1408	87
Encoder_Data[434]	1495	-7
Encoder_Data[435]	1488	-45
Encoder_Data[436]	1443	-8
Encoder_Data[437]	1435	-26
Encoder_Data[438]	1409	68
Encoder_Data[439]	1477	-45
Encoder_Data[440]	1432	-8
Encoder_Data[441]	1424	12
Encoder_Data[442]	1436	-46
Encoder_Data[443]	1390	68
Encoder_Data[444]	1458	12
Encoder_Data[445]	1470	-64
Encoder_Data[446]	1406	-8
Encoder_Data[447]	1398	11

Encoder_Data[448]	1409	31
Encoder_Data[449]	1440	-46
Encoder_Data[450]	1394	-7
Encoder_Data[451]	1387	49
Encoder_Data[452]	1436	-46
Encoder_Data[453]	1390	50
Encoder_Data[454]	1440	-46
Encoder_Data[455]	1394	-27
Encoder_Data[456]	1367	-26
Encoder_Data[457]	1341	49
Encoder_Data[458]	1390	11
Encoder_Data[459]	1401	-45
Encoder_Data[460]	1356	-46
Encoder_Data[461]	1310	12
Encoder_Data[462]	1322	49
Encoder_Data[463]	1371	31
Encoder_Data[464]	1402	-46
Encoder_Data[465]	1356	-45
Encoder_Data[466]	1311	11
Encoder_Data[467]	1322	49
Encoder_Data[468]	1371	-26
Encoder_Data[469]	1345	-8
Encoder_Data[470]	1337	-64
Encoder_Data[471]	1273	11
Encoder_Data[472]	1284	68
Encoder_Data[473]	1352	-27
Encoder_Data[474]	1325	31
Encoder_Data[475]	1356	-46
Encoder_Data[476]	1310	-8
Encoder_Data[477]	1302	11
Encoder_Data[478]	1313	11
Encoder_Data[479]	1324	-26
Encoder_Data[480]	1298	11
Encoder_Data[481]	1309	49
Encoder_Data[482]	1358	-27
Encoder_Data[483]	1331	-64
Encoder_Data[484]	1267	49
Encoder_Data[485]	1316	-65
Encoder_Data[486]	1251	69
Encoder_Data[487]	1320	-8
Encoder_Data[488]	1312	-8

Encoder_Data[489]	1304	-84
Encoder_Data[490]	1220	50
Encoder_Data[491]	1270	-102
Encoder_Data[492]	1168	30
Encoder_Data[493]	1198	87
Encoder_Data[494]	1285	11
Encoder_Data[495]	1296	-46
Encoder_Data[496]	1250	50
Encoder_Data[497]	1300	-7
Encoder_Data[498]	1293	-27
Encoder_Data[499]	1266	-27
Encoder_Data[500]	1239	11
Encoder_Data[501]	1250	-83
Encoder_Data[502]	1167	106
Encoder_Data[503]	1273	-27
Encoder_Data[504]	1246	12
Encoder_Data[505]	1258	-65
Encoder_Data[506]	1193	-7
Encoder_Data[507]	1186	49
Encoder_Data[508]	1235	-45
Encoder_Data[509]	1190	30
Encoder_Data[510]	1220	-63
Encoder_Data[511]	1157	67
Encoder_Data[512]	1224	12
Encoder_Data[513]	1236	-27
Encoder_Data[514]	1209	-7
Encoder_Data[515]	1202	-45
Encoder_Data[516]	1157	11
Encoder_Data[517]	1168	49
Encoder_Data[518]	1217	-8
Encoder_Data[519]	1209	-45
Encoder_Data[520]	1164	11
Encoder_Data[521]	1175	-62
Encoder_Data[522]	1113	-7
Encoder_Data[523]	1106	85
Encoder_Data[524]	1191	11
Encoder_Data[525]	1202	-63
Encoder_Data[526]	1139	-25
Encoder_Data[527]	1114	-25
Encoder_Data[528]	1089	84
Encoder_Data[529]	1173	-26

Encoder_Data[530]	1147	11
Encoder_Data[531]	1158	-97
Encoder_Data[532]	1061	66
Encoder_Data[533]	1127	-96
Encoder_Data[534]	1031	82
Encoder_Data[535]	1113	11
Encoder_Data[536]	1124	-43
Encoder_Data[537]	1081	-7
Encoder_Data[538]	1074	83
Encoder_Data[539]	1157	-61
Encoder_Data[540]	1096	10
Encoder_Data[541]	1106	-42
Encoder_Data[542]	1064	46
Encoder_Data[543]	1110	11
Encoder_Data[544]	1121	-25
Encoder_Data[545]	1096	-43
Encoder_Data[546]	1053	-24
Encoder_Data[547]	1029	64
Encoder_Data[548]	1093	-24
Encoder_Data[549]	1069	-7
Encoder_Data[550]	1062	28
Encoder_Data[551]	1090	-6
Encoder_Data[552]	1084	-25
Encoder_Data[553]	1059	-6
Encoder_Data[554]	1053	-24
Encoder_Data[555]	1029	10
Encoder_Data[556]	1039	47
Encoder_Data[557]	1086	-42
Encoder_Data[558]	1044	-7
Encoder_Data[559]	1037	-43
Encoder_Data[560]	994	65
Encoder_Data[561]	1059	-25
Encoder_Data[562]	1034	29
Encoder_Data[563]	1063	-60
Encoder_Data[564]	1003	11
Encoder_Data[565]	1014	65
Encoder_Data[566]	1079	-43
Encoder_Data[567]	1036	-25
Encoder_Data[568]	1011	-7
Encoder_Data[569]	1004	-25
Encoder_Data[570]	979	30

Encoder_Data[571]	1009	-25
Encoder_Data[572]	984	47
Encoder_Data[573]	1031	-24
Encoder_Data[574]	1007	-25
Encoder_Data[575]	982	30
Encoder_Data[576]	1012	11
Encoder_Data[577]	1023	-25
Encoder_Data[578]	998	-26
Encoder_Data[579]	972	49
Encoder_Data[580]	1021	11
Encoder_Data[581]	1032	-25
Encoder_Data[582]	1007	-63
Encoder_Data[583]	944	-25
Encoder_Data[584]	919	87
Encoder_Data[585]	1006	-26
Encoder_Data[586]	980	-6
Encoder_Data[587]	974	-7
Encoder_Data[588]	967	-104
Encoder_Data[589]	863	128
Encoder_Data[590]	991	-45
Encoder_Data[591]	946	49
Encoder_Data[592]	995	-25
Encoder_Data[593]	970	-65
Encoder_Data[594]	905	13
Encoder_Data[595]	918	50
Encoder_Data[596]	968	12
Encoder_Data[597]	980	-64
Encoder_Data[598]	916	12
Encoder_Data[599]	928	-65
Encoder_Data[600]	863	108
Encoder_Data[601]	971	-45
Encoder_Data[602]	926	13
Encoder_Data[603]	939	-26
Encoder_Data[604]	913	-27
Encoder_Data[605]	886	51
Encoder_Data[606]	937	13
Encoder_Data[607]	950	-46
Encoder_Data[608]	904	-7
Encoder_Data[609]	897	-65
Encoder_Data[610]	832	71
Encoder_Data[611]	903	12

Encoder_Data[612]	915	-46
Encoder_Data[613]	869	-7
Encoder_Data[614]	862	52
Encoder_Data[615]	914	-7
Encoder_Data[616]	907	-7
Encoder_Data[617]	900	-7
Encoder_Data[618]	893	-46
Encoder_Data[619]	847	13
Encoder_Data[620]	860	12
Encoder_Data[621]	872	32
Encoder_Data[622]	904	-85
Encoder_Data[623]	819	71
Encoder_Data[624]	890	-7
Encoder_Data[625]	883	-7
Encoder_Data[626]	876	-67
Encoder_Data[627]	809	-7
Encoder_Data[628]	802	91
Encoder_Data[629]	893	-27
Encoder_Data[630]	866	-8
Encoder_Data[631]	858	-26
Encoder_Data[632]	832	-46
Encoder_Data[633]	786	71
Encoder_Data[634]	857	-7
Encoder_Data[635]	850	13
Encoder_Data[636]	863	-66
Encoder_Data[637]	797	-26
Encoder_Data[638]	771	71
Encoder_Data[639]	842	13
Encoder_Data[640]	855	-27
Encoder_Data[641]	828	-65
Encoder_Data[642]	763	-26
Encoder_Data[643]	737	108
Encoder_Data[644]	845	-27
Encoder_Data[645]	818	-26
Encoder_Data[646]	792	12
Encoder_Data[647]	804	-83
Encoder_Data[648]	721	12
Encoder_Data[649]	733	50
Encoder_Data[650]	783	12
Encoder_Data[651]	795	12
Encoder_Data[652]	807	-45

Encoder_Data[653]	762	-26
Encoder_Data[654]	736	-25
Encoder_Data[655]	711	-7
Encoder_Data[656]	704	68
Encoder_Data[657]	772	-26
Encoder_Data[658]	746	-7
Encoder_Data[659]	739	-81
Encoder_Data[660]	658	124
Encoder_Data[661]	782	-26
Encoder_Data[662]	756	12
Encoder_Data[663]	768	-26
Encoder_Data[664]	742	-44
Encoder_Data[665]	698	-7
Encoder_Data[666]	691	-44
Encoder_Data[667]	647	124
Encoder_Data[668]	771	-100
Encoder_Data[669]	671	29
Encoder_Data[670]	700	-43
Encoder_Data[671]	657	85
Encoder_Data[672]	742	12
Encoder_Data[673]	754	-44
Encoder_Data[674]	710	-62
Encoder_Data[675]	648	30
Encoder_Data[676]	678	48
Encoder_Data[677]	726	-7
Encoder_Data[678]	719	-6
Encoder_Data[679]	713	-25
Encoder_Data[680]	688	-26
Encoder_Data[681]	662	49
Encoder_Data[682]	711	12
Encoder_Data[683]	723	-26
Encoder_Data[684]	697	-25
Encoder_Data[685]	672	30
Encoder_Data[686]	702	-7
Encoder_Data[687]	695	-81
Encoder_Data[688]	614	12
Encoder_Data[689]	626	48
Encoder_Data[690]	674	30
Encoder_Data[691]	704	-44
Encoder_Data[692]	660	-7
Encoder_Data[693]	653	-43

Encoder_Data[694]	610	85
Encoder_Data[695]	695	-26
Encoder_Data[696]	669	-62
Encoder_Data[697]	607	-6
Encoder_Data[698]	601	84
Encoder_Data[699]	685	-43
Encoder_Data[700]	642	11
Encoder_Data[701]	653	-7
Encoder_Data[702]	646	-44
Encoder_Data[703]	602	11
Encoder_Data[704]	613	30
Encoder_Data[705]	643	-7
Encoder_Data[706]	636	-44
Encoder_Data[707]	592	12
Encoder_Data[708]	604	-62
Encoder_Data[709]	542	103
Encoder_Data[710]	645	-7
Encoder_Data[711]	638	-7
Encoder_Data[712]	631	-25
Encoder_Data[713]	606	-25
Encoder_Data[714]	581	-26
Encoder_Data[715]	555	49
Encoder_Data[716]	604	-7
Encoder_Data[717]	597	-7
Encoder_Data[718]	590	-7
Encoder_Data[719]	583	-7
Encoder_Data[720]	576	31
Encoder_Data[721]	607	-7
Encoder_Data[722]	600	-7
Encoder_Data[723]	593	-7
Encoder_Data[724]	586	-26
Encoder_Data[725]	560	-62
Encoder_Data[726]	498	104
Encoder_Data[727]	602	-62
Encoder_Data[728]	540	30
Encoder_Data[729]	570	-44
Encoder_Data[730]	526	11
Encoder_Data[731]	537	12
Encoder_Data[732]	549	11
Encoder_Data[733]	560	-7
Encoder_Data[734]	553	-44

Encoder_Data[735]	509	67
Encoder_Data[736]	576	-7
Encoder_Data[737]	569	-26
Encoder_Data[738]	543	-7
Encoder_Data[739]	536	-44
Encoder_Data[740]	492	12
Encoder_Data[741]	504	30
Encoder_Data[742]	534	12
Encoder_Data[743]	546	-26
Encoder_Data[744]	520	-44
Encoder_Data[745]	476	67
Encoder_Data[746]	543	-63
Encoder_Data[747]	480	11
Encoder_Data[748]	491	-25
Encoder_Data[749]	466	-45
Encoder_Data[750]	421	-26
Encoder_Data[751]	395	123
Encoder_Data[752]	518	-7
Encoder_Data[753]	511	-45
Encoder_Data[754]	466	-6
Encoder_Data[755]	460	67
Encoder_Data[756]	527	-63
Encoder_Data[757]	464	-26
Encoder_Data[758]	438	49
Encoder_Data[759]	487	-82
Encoder_Data[760]	405	-8
Encoder_Data[761]	397	49
Encoder_Data[762]	446	-26
Encoder_Data[763]	420	11
Encoder_Data[764]	431	-26
Encoder_Data[765]	405	-45
Encoder_Data[766]	360	12
Encoder_Data[767]	372	30
Encoder_Data[768]	402	11
Encoder_Data[769]	413	-8
Encoder_Data[770]	405	31
Encoder_Data[771]	436	-27
Encoder_Data[772]	409	-44
Encoder_Data[773]	365	11
Encoder_Data[774]	376	-26
Encoder_Data[775]	350	49

Encoder_Data[776]	399	11
Encoder_Data[777]	410	-63
Encoder_Data[778]	347	66
Encoder_Data[779]	413	-26
Encoder_Data[780]	387	-26
Encoder_Data[781]	361	30
Encoder_Data[782]	391	-26
Encoder_Data[783]	365	11
Encoder_Data[784]	376	-63
Encoder_Data[785]	313	30
Encoder_Data[786]	343	11
Encoder_Data[787]	354	48
Encoder_Data[788]	402	-7
Encoder_Data[789]	395	-45
Encoder_Data[790]	350	-25
Encoder_Data[791]	325	-8
Encoder_Data[792]	317	48
Encoder_Data[793]	365	-7
Encoder_Data[794]	358	-8
Encoder_Data[795]	350	-44
Encoder_Data[796]	306	-45
Encoder_Data[797]	261	67
Encoder_Data[798]	328	29
Encoder_Data[799]	357	-63
Encoder_Data[800]	294	-26
Encoder_Data[801]	268	48
Encoder_Data[802]	316	48
Encoder_Data[803]	364	-45
Encoder_Data[804]	319	-7
Encoder_Data[805]	312	-45
Encoder_Data[806]	267	-26
Encoder_Data[807]	241	66
Encoder_Data[808]	307	-26
Encoder_Data[809]	281	-26
Encoder_Data[810]	255	-27
Encoder_Data[811]	228	86
Encoder_Data[812]	314	-26
Encoder_Data[813]	288	29
Encoder_Data[814]	317	-26
Encoder_Data[815]	291	-26
Encoder_Data[816]	265	-8

Encoder_Data[817]	257	-64
Encoder_Data[818]	193	86
Encoder_Data[819]	279	-26
Encoder_Data[820]	253	-7
Encoder_Data[821]	246	-26
Encoder_Data[822]	220	29
Encoder_Data[823]	249	30
Encoder_Data[824]	279	-82
Encoder_Data[825]	197	48
Encoder_Data[826]	245	-65
Encoder_Data[827]	180	68
Encoder_Data[828]	248	11
Encoder_Data[829]	259	-46
Encoder_Data[830]	213	-7
Encoder_Data[831]	206	-46
Encoder_Data[832]	160	49
Encoder_Data[833]	209	48
Encoder_Data[834]	257	-45
Encoder_Data[835]	212	-27
Encoder_Data[836]	185	49
Encoder_Data[837]	234	-64
Encoder_Data[838]	170	30
Encoder_Data[839]	200	-8
Encoder_Data[840]	192	-7
Encoder_Data[841]	185	-46
Encoder_Data[842]	139	68
Encoder_Data[843]	207	-8
Encoder_Data[844]	199	-8
Encoder_Data[845]	191	-64
Encoder_Data[846]	127	-27
Encoder_Data[847]	100	68
Encoder_Data[848]	168	11
Encoder_Data[849]	179	-27
Encoder_Data[850]	152	-27
Encoder_Data[851]	125	-45
Encoder_Data[852]	80	49
Encoder_Data[853]	129	-8
Encoder_Data[854]	121	-9
Encoder_Data[855]	112	31
Encoder_Data[856]	143	11
Encoder_Data[857]	154	-9

Encoder_Data[858]	145	-7
Encoder_Data[859]	138	-65
Encoder_Data[860]	73	11
Encoder_Data[861]	84	30
Encoder_Data[862]	114	-8
Encoder_Data[863]	106	11
Encoder_Data[864]	117	-83
Encoder_Data[865]	34	87
Encoder_Data[866]	121	-46
Encoder_Data[867]	75	-7
Encoder_Data[868]	68	11
Encoder_Data[869]	79	-65
Encoder_Data[870]	14	68
Encoder_Data[871]	82	11
Encoder_Data[872]	93	-27
Encoder_Data[873]	66	-26
Encoder_Data[874]	40	-8
Encoder_Data[875]	32	11
Encoder_Data[876]	43	11
Encoder_Data[877]	54	-8
Encoder_Data[878]	46	-46
Encoder_Data[879]	0	68
Encoder_Data[880]	68	3994
Encoder_Data[881]	4062	-4048
Encoder_Data[882]	14	-7
Encoder_Data[883]	7	4032
Encoder_Data[884]	4039	-3991
Encoder_Data[885]	48	-27
Encoder_Data[886]	21	-8
Encoder_Data[887]	13	4051
Encoder_Data[888]	4064	-4029
Encoder_Data[889]	35	-26
Encoder_Data[890]	9	11
Encoder_Data[891]	20	4013
Encoder_Data[892]	4033	48
Encoder_Data[893]	4081	-4066
Encoder_Data[894]	15	-8
Encoder_Data[895]	7	4088
Encoder_Data[896]	4095	-26
Encoder_Data[897]	4069	-8
Encoder_Data[898]	4061	-45

Encoder_Data[899]	4016	-7
Encoder_Data[900]	4009	11
Encoder_Data[901]	4020	49
Encoder_Data[902]	4069	-101
Encoder_Data[903]	3968	67
Encoder_Data[904]	4035	30
Encoder_Data[905]	4065	-26
Encoder_Data[906]	4039	10
Encoder_Data[907]	4049	-63
Encoder_Data[908]	3986	-26
Encoder_Data[909]	3960	85
Encoder_Data[910]	4045	-45
Encoder_Data[911]	4000	30
Encoder_Data[912]	4030	-27
Encoder_Data[913]	4003	-26
Encoder_Data[914]	3977	11
Encoder_Data[915]	3988	11
Encoder_Data[916]	3999	29
Encoder_Data[917]	4028	-44
Encoder_Data[918]	3984	29
Encoder_Data[919]	4013	11
Encoder_Data[920]	4024	-26
Encoder_Data[921]	3998	-64
Encoder_Data[922]	3934	30
Encoder_Data[923]	3964	-64
Encoder_Data[924]	3900	86
Encoder_Data[925]	3986	-8
Encoder_Data[926]	3978	-7
Encoder_Data[927]	3971	-8
Encoder_Data[928]	3963	-44
Encoder_Data[929]	3919	48
Encoder_Data[930]	3967	-7
Encoder_Data[931]	3960	-8
Encoder_Data[932]	3952	-45
Encoder_Data[933]	3907	-7
Encoder_Data[934]	3900	67
Encoder_Data[935]	3967	-8
Encoder_Data[936]	3959	-8
Encoder_Data[937]	3951	-26
Encoder_Data[938]	3925	-44
Encoder_Data[939]	3881	66

Encoder_Data[940]	3947	-7
Encoder_Data[941]	3940	-26
Encoder_Data[942]	3914	-26
Encoder_Data[943]	3888	11
Encoder_Data[944]	3899	-100
Encoder_Data[945]	3799	122
Encoder_Data[946]	3921	-8
Encoder_Data[947]	3913	-26
Encoder_Data[948]	3887	-26
Encoder_Data[949]	3861	-45
Encoder_Data[950]	3816	104
Encoder_Data[951]	3920	-45
Encoder_Data[952]	3875	30
Encoder_Data[953]	3905	-26
Encoder_Data[954]	3879	-8
Encoder_Data[955]	3871	-62
Encoder_Data[956]	3809	85
Encoder_Data[957]	3894	-8
Encoder_Data[958]	3886	-26
Encoder_Data[959]	3860	-25
Encoder_Data[960]	3835	11
Encoder_Data[961]	3846	11
Encoder_Data[962]	3857	11
Encoder_Data[963]	3868	-44
Encoder_Data[964]	3824	-62
Encoder_Data[965]	3762	11
Encoder_Data[966]	3773	66
Encoder_Data[967]	3839	-7
Encoder_Data[968]	3832	11
Encoder_Data[969]	3843	-44
Encoder_Data[970]	3799	29
Encoder_Data[971]	3828	11
Encoder_Data[972]	3839	-80
Encoder_Data[973]	3759	29
Encoder_Data[974]	3788	29
Encoder_Data[975]	3817	-98
Encoder_Data[976]	3719	66
Encoder_Data[977]	3785	-7
Encoder_Data[978]	3778	11
Encoder_Data[979]	3789	-62
Encoder_Data[980]	3727	11

Encoder_Data[981]	3738	30
Encoder_Data[982]	3768	-7
Encoder_Data[983]	3761	-7
Encoder_Data[984]	3754	-61
Encoder_Data[985]	3693	47
Encoder_Data[986]	3740	11
Encoder_Data[987]	3751	-116
Encoder_Data[988]	3635	66
Encoder_Data[989]	3701	11
Encoder_Data[990]	3712	11
Encoder_Data[991]	3723	11
Encoder_Data[992]	3734	11
Encoder_Data[993]	3745	-61
Encoder_Data[994]	3684	11
Encoder_Data[995]	3695	-26
Encoder_Data[996]	3669	48
Encoder_Data[997]	3717	11
Encoder_Data[998]	3728	-26
Encoder_Data[999]	3702	-3702

APPENDIX F DYNAMOMETER TESTING RESULTS

A. 7.5 HP Machine

1) Raw Data

20 N- m	30 N- m	40 N- m	50 N- m	60 N- m	70 N- m	80 N- m	90 N- m	100 N-m	110 N-m	120 N-m	130 N-m
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-
2.1E- 08	3.1E- 08	4.1E- 08	5.2E- 08	6.2E- 08	7.2E- 08	8.3E- 08	9.3E- 08	-1E- 07	1.1E- 07	1.2E- 07	1.3E- 07
-	-	-	-	-	-	-	-	-	-	-	-
0.336 52	0.504 78	0.673 05	0.841 32	1.009 59	1.177 87	1.346 15	1.514 43	1.682 71	- 1.851	2.019 29	2.187 58
-	-	-	-	-	-	-	-	-	-	-	-
1.248 71	1.873 25	2.497 91	3.122 7	3.747 62	4.372 66	4.997 83	5.623 13	6.248 55	6.874 11	7.499 79	8.125 61
-	-	-	-	-	-	-	-	-	-	-	-
2.511 21	3.768 32	5.026 45	6.285 62	7.545 84	8.807 13	10.06 95	11.33 3	12.59 75	13.86 32	- 15.13	16.39 8
-	-	-	-	-	-	-	-	-	-	-	-
3.876 15	5.820 17	7.768 31	9.720 66	11.67 73	13.63 84	15.60 41	17.57 43	19.54 93	21.52 92	23.51 4	25.50 39
-	-	-	-	-	-	-	-	-	-	-	-
5.142 04	7.728 7	10.32 64	12.93 56	15.55 67	18.19 02	20.83 66	23.49 62	26.16 97	28.85 73	31.55 97	34.27 73
-	-	-	-	-	-	-	-	-	-	-	-
6.182 82	9.305 72	12.45 14	15.62 13	18.81 67	22.03 9	25.28 96	28.56 99	31.88 13	35.22 53	38.60 33	42.01 67
-	-	-	-	-	-	-	-	-	-	-	-
6.947 95	10.47 44	- 14.04	17.64 8	21.30 16	25.00 42	28.75 91	32.56 96	36.43 92	40.37 14	44.36 96	48.43 74
-	-	-	-	-	-	-	-	-	-	-	-
7.446 15	11.24 5	15.10 26	19.02 53	23.01 94	27.09 15	31.24 85	35.49 71	39.84 45	44.29 76	48.86 39	53.55 06
-	-	-	-	-	-	-	-	-	-	-	-
7.723 21	11.68 3	15.72 2	19.85 08	24.08 07	28.42 31	- 32.89	37.49 36	42.24 68	47.16 25	52.25 43	57.53 6
-	-	-	-	-	-	-	-	-	-	-	-
7.840 87	11.87 78	16.01 29	20.26 25	24.64 37	29.17 47	33.87 45	38.76 31	43.86 15	49.19 17	54.77 65	60.63 99
-	-	-	-	-	-	-	-	-	-	-	-
7.860	11.91	16.09	20.40	24.87	29.52	-34.4	39.51	44.91	50.61	56.66	63.10

45	96	25	17	16	84		65		51	8	76
-	-	-	-	-	-	-	-	-	-	-	-
7.832	11.88	16.06	20.39	24.90	29.64	34.63	39.93	45.58	51.63	58.13	65.14
69	4	01	01	59	28	92	77	46	08	16	65
-	-	-	-	-	-	-	-	-	-	-	-
7.793	11.82	15.98	20.31	24.85	29.63	34.72	40.16	46.03	52.39	59.32	66.91
32	53	84	79	31	86	48	82	3	07	15	39
-	-	-	-	-	-	-	-	-	-	-	-
7.763	11.77	15.92	20.24	24.78	29.59	34.74	40.30	46.35	52.99	60.34	68.51
04	61	26	26	27	66	66	54	72	95	41	83
-	-	-	-	-	-	-	-	-	-	-	-
7.750	11.75	15.88	20.19	24.73	29.56	34.75	40.40	46.62	53.52	61.26	70.02
23	09	4	38	24	22	75	77	01	23	6	87
-	-	-	-	-	-	-	-	-	-	-	-
7.754	11.75	15.87	20.17	24.71	29.55	34.78	40.50	46.85	53.99	62.12	71.48
85	11	71	98	57	46	23	52	53	54	51	6
-	-	-	-	-	-	-	-	-	-	-	-
7.772	11.77	15.89	20.19	24.73	29.57	34.82	40.60	47.07	54.43	62.94	72.91
25	12	65	65	11	6	74	84	71	56	05	32
-	-	-	-	-	-	-	-	-	-	-	-
7.796	11.80	15.93	20.23	24.76	29.61	34.88	40.71	47.28	54.84	63.72	74.32
3	28	22	37	93	93	89	73	9	91	06	29
-	-	-	-	-	-	-	-	-	-	-	-
7.821	11.83	15.97	20.28	24.81	29.67	34.95	40.82	47.48	55.23	64.46	75.72
38	75	42	03	92	45	91	68	9	68	87	3
-	-	-	-	-	-	-	-	-	-	-	-
7.843	11.86	16.01	20.32	24.87	29.73	35.02	40.93	47.67	55.59	65.18	77.11
52	95	44	69	08	21	99	14	42	78	6	92
-	-	-	-	-	-	-	-	-	-	-	-
7.860	11.89	16.04	20.36	24.91	29.78	35.09	41.02	47.84	55.93	65.87	78.51
59	51	79	73	71	52	53	65	2	15	37	73
-	-	-	-	-	-	-	-	-	-	-	-
7.872	11.91	16.07	20.39	24.95	29.82	35.15	41.10	47.99	56.23	66.53	79.92
06	31	26	85	46	96	14	95	11	8	31	34
-	-	-	-	-	-	-	-	-	-	-	-
7.878	11.92	16.08	-	24.98	29.86	35.19	41.17	48.12	56.51	67.16	81.34
54	39	85	20.42	2	41	7	95	18	84	62	39
-	-	-	-	-	-	-	-	-	-	-	-
7.881	11.92	16.09	20.43	25.00	29.88	35.23	41.23	48.23	56.77	67.77	82.78
24	91	72	3	03	89	24	72	53	44	54	6
-	-	-	-	-	-	-	-	-	-	-	-
7.881	11.93	16.10	20.43	25.01	29.90	35.25	41.28	48.33	57.00	68.36	84.25
52	05	06	95	11	57	88	39	31	81	3	73
-	-	-	-	-	-	-	-	-	-	-	-
7.880	11.92	16.10	20.44	25.01	29.91	35.27	41.32	48.41	57.22	68.93	85.76
55	97	08	18	68	64	8	14	75	16	16	59
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.01	29.92	35.29	41.35	48.49	57.41	69.48	87.32

25	81	95	17	91	29	19	14	02	72	33	03
-	-	-	-	-	-	-	-	-	-	-	-
7.878	11.92	16.09	20.44	25.01	29.92	35.30	41.37	48.55	57.59	70.02	88.92
14	65	78	07	97	68	21	57	32	66	02	98
-	-	-	-	-	-	-	-	-	-	-	-
7.877	11.92	16.09	20.43	25.01	29.92	35.30	41.39	48.60	57.76	70.54	90.60
49	54	65	96	97	92	96	55	78	15	41	42
-	-	-	-	-	-	-	-	-	-	-	-
7.877	11.92	16.09	20.43	25.01	29.93	35.31	41.41	48.65	57.91	71.05	92.35
31	5	58	89	97	09	55	19	55	34	67	47
-	-	-	-	-	-	-	-	-	-	-	-
7.877	11.92	16.09	20.43	25.01	29.93	35.32	41.42	48.69	58.05	71.55	94.19
51	5	57	88	99	24	02	55	72	36	95	31
-	-	-	-	-	-	-	-	-	-	-	-
7.877	11.92	16.09	20.43	25.02	29.93	35.32	41.43	48.73	58.18	72.05	96.13
94	55	61	91	04	38	42	7	37	3	37	33
-	-	-	-	-	-	-	-	-	-	-	-
7.878	11.92	16.09	20.43	25.02	29.93	35.32	41.44	48.76	58.30	72.54	98.19
45	61	68	98	12	52	76	67	57	27	06	04
-	-	-	-	-	-	-	-	-	-	-	-
7.878	11.92	16.09	20.44	25.02	29.93	35.33	41.45	48.79	58.41	73.02	100.3
94	68	75	06	21	65	05	5	38	34	13	82
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.93	35.33	41.46	48.81	58.51	73.49	102.7
34	74	82	14	3	78	3	2	85	59	68	28
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.93	35.33	41.46	48.84	58.61	73.96	105.2
61	78	88	2	38	88	51	79	02	09	82	52
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.93	35.33	41.47	48.85	58.69	74.43	107.9
77	81	92	26	44	97	69	3	93	89	64	81
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.33	41.47	48.87	58.78	74.90	110.9
84	82	94	29	49	05	83	72	6	06	23	46
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.33	41.48	48.89	58.85	75.36	114.1
85	83	95	31	52	1	95	07	06	63	68	83
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.48	48.90	58.92	75.83	117.7
82	83	96	32	54	14	04	37	34	66	07	35
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.48	48.91	58.99	76.29	121.6
79	82	96	33	56	17	11	62	47	19	49	53
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.48	48.92	59.05	76.76	125.9
75	82	95	33	56	19	17	82	46	26	03	97
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.93	59.10	77.22	130.8

73	82	95	33	57	21	22		32	9	76	37
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.94	59.16	77.69	136.2
72	81	95	32	57	22	26	14	08	14	76	54
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.94	59.21	78.17	142.3
72	81	95	32	57	23	28	26	75	01	14	47
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.95	59.25	78.64	149.2
73	81	95	32	57	23	31	36	34	55	96	27
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.95	59.29	79.13	157.0
74	81	95	32	57	24	33	45	85	77	31	25
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.96	59.33	79.62	165.8
75	82	95	33	58	24	34	52	3	7	3	87
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.96	59.37	-	175.9
76	82	95	33	58	25	36	58	69	35	80.12	77
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.97	59.40	80.62	187.4
76	82	95	33	58	25	37	63	04	76	52	66
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.97	59.43	81.13	200.5
77	82	95	33	58	25	37	67	34	93	96	29
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.97	59.46	81.66	215.3
77	82	95	33	58	25	38	71	61	89	42	29
-	-	-	-	-	-	-	-	-	-	-	-
7.879	11.92	16.09	20.44	25.02	29.94	35.34	41.49	48.97	59.49	82.20	232.0
77	82	95	33	58	26	39	74	85	64	02	01

2) Plots

B. 10 HP Machine

1) Raw Data

20 N- m	30 N-m	40 N- m	50 N- m	60 N- m	70 N- m	80 N- m	90 N- m	100 N-m	110 N-m	120 N-m
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
-1.1E-08	-1.7E-08	-2.2E-08	-2.8E-08	-3.4E-08	-3.9E-08	-4.5E-08	-5.1E-08	-5.6E-08	-6.2E-08	-6.7E-08
-	-	-	-	-	-	-	-	-	-	-
0.18473	-0.2771	0.36946	0.46183	0.5542	0.64657	0.73893	0.8313	0.92367	1.01604	1.10841
-	-	-	-	-	-	-	-	-	-	-
0.70945	-1.0642	1.41898	1.77378	2.1286	2.48344	2.8383	3.19318	3.54809	3.90301	4.25796
-	-	-	-	-	-	-	-	-	-	-
1.50228	2.25369	3.00529	3.75707	4.50904	5.26121	6.01356	6.7661	7.51884	8.27178	9.02491
-	-	-	-	-	-	-	-	-	-	-
2.47388	3.71202	4.95097	6.19074	7.43135	8.67281	9.91513	11.1583	12.4024	13.6474	14.8932
-	-	-	-	-	-	-	-	-	-	-
3.53475	5.30565	7.07897	8.85478	10.6331	12.4141	14.1977	15.984	17.7731	19.565	21.3598

-	-	-	-	-	-	-	-	-	-	-
7.947	11.979	16.05	20.18	24.36	28.60	32.90	37.28	41.72	46.25	-50.86
32	1	59	23	33	34	75	05	7	2	-
-	-	-	-	-	-	-	-	-	-	-
8.426	12.717	17.07	21.49	25.99	30.57		40.04	44.93	49.94	55.08
47	3	02	3	35	98	-35.26	23	48	56	29
-	-	-		-	-	-	-	-	-	-
8.759	13.237	17.79		27.20	32.08	37.09	42.24	47.55	53.03	58.69
73	9	73	-22.45	83	48	22	36	2	06	24
-	-	-	-	-	-	-	-	-	-	-
8.960	13.560	18.26	23.08	28.03	33.15	38.44	43.92	49.62	55.55	61.74
96	5	18	22	99	34	2	51	26	5	24
-	-	-	-	-	-	-	-	-	-	-
9.048	13.711	18.49	23.42	28.53	33.83	39.36	45.14	51.20	57.58	64.29
83	8	74	95	33	49	17	2	5	07	96
-	-	-	-	-	-	-	-	-	-	-
9.044	13.722	18.54	23.53	28.74	34.18	39.91	45.95	52.36	59.17	66.43
54	7	35	85	08	61	16	71	39	51	53
-	-	-	-	-	-	-	-	-	-	-
8.969		18.44	23.45	28.71	34.26	40.15	46.43	53.16	60.40	68.21
99	-13.625	11	77	74	64	43	47	48	5	84
-	-	-	-	-	-	-	-	-	-	-
8.846	13.449	18.23	23.23	28.51	34.13	40.15	46.63		61.33	69.71
24	8	01	5	71	43	08	71	-53.67	24	25
-	-	-	-	-	-	-	-	-	-	-
8.692	13.225	17.94	22.91	28.19	33.84	39.95	46.62	53.93	62.01	70.97
47	4	77	5	03	46	83	19	64	32	33
-	-	-	-	-	-	-	-	-	-	-
8.525	12.976	17.62	22.53	27.78	33.44	39.62	46.44	54.01	62.49	72.04
19	8	62	7	23	66	82	07	41	57	85
-	-	-	-	-	-	-	-	-	-	-
8.357	12.724	17.29	22.13	27.33	32.98	39.20	46.13	53.94	62.82	72.97
86	2	32	49	21	32	55	85	65	06	77
-	-	-	-	-	-	-	-	-	-	-
8.200	12.483	16.97	21.73	26.87	32.49	38.72	45.75	53.77	63.02	73.79
8	8	02	57	18	03	84	38	04	21	34
-	-	-	-	-	-	-	-	-	-	-
8.061	12.266	16.67	21.36	26.42	31.99	38.22	45.31	53.51	63.12	74.52
24	9	33	02	67	71	83	86	63	8	16
-	-	-	-	-	-	-	-	-	-	-
7.943		16.41	21.02	26.01	31.52	37.73	44.85	53.20	63.16	75.18
64	-12.081	34	33	56	59	05	91	93	06	32
-	-	-	-	-	-	-	-	-	-	-
7.850	11.929	16.19	20.73	25.65	31.09	37.25	44.39	52.86	63.13	75.79
04	9	67	41	13	31	42	62	98	81	45
-	-	-	-	-	-	-	-	-	-	-
7.780	11.814	16.02	20.49	25.34	30.70	36.81	43.94	52.51	63.07	76.36
52	4	55	72	11	96	34	58	38	46	83

-	-	-	-	-	-	-	-	-	-	-
7.733	11.733	15.89	20.31	25.08	30.38	36.41		52.15	62.98	76.91
65	1	9	33	83	14	75	-43.52	41	14	47
-	-	-	-	-	-	-	-	-	-	-
7.706	11.682	15.81		24.89	30.11	36.07	43.12	51.80	62.86	77.44
96	7	4	-20.18	21	07	2	72	05	73	14
-	-	-	-	-	-	-	-	-	-	-
7.697		15.76	20.09	24.74	29.89	35.77	42.77	51.46	62.73	77.95
29	-11.659	55	27	92	65	9	27	02	9	45
-	-	-	-	-	-	-	-	-	-	-
7.701	11.657	15.74	20.04	24.65	29.73	35.53	42.45	51.13	62.60	78.45
19	1	79	53	45	56	81	95	86	2	88
-	-	-	-	-	-	-	-	-	-	-
7.715	11.671	15.75	20.03	24.60	29.62	35.34	42.18	50.83	62.46	78.95
2	9	47	1	15	29	7	84	91	03	81
-	-	-	-	-	-	-	-	-	-	-
7.736	11.698	15.77	20.04	24.58	29.55	35.20	41.95	50.56	62.31	79.45
05	6	99	29	3	22	18	86	41	69	55
-	-	-	-	-	-	-	-	-	-	-
7.760	11.732	15.81	20.07	24.59	29.51	35.09	41.76	50.31	62.17	79.95
84	8	77	41	18	7	75	83	45	42	33
-	-	-	-	-	-	-	-	-	-	-
7.787	11.770	15.86	20.11	24.62	29.51	35.02	41.61	50.09	62.03	80.45
09	7	29	83	11	05	88	47	08	4	38
-	-	-	-	-	-	-	-	-	-	-
7.812	11.809	15.91		24.66	29.52		41.49	49.89	61.89	80.95
83	1	12	-20.17	44	62	-34.99	43	25	75	87
-	-	-	-	-	-	-	-	-	-	-
7.836	11.845	15.95	20.22	24.71	29.55	34.97	41.40	49.71	61.76	81.46
6	6	9	47	62	81	57	33	85	57	96
-	-	-	-	-	-	-	-	-	-	-
7.857	11.878	16.00	20.27	24.77	29.60	34.98	41.33	49.56	61.63	81.98
4	4	37	85	19	07	06	8	76	93	78
-	-	-	-	-	-	-	-	-	-	-
7.874	11.906	16.04	20.32	24.82	29.64	34.99	41.29	49.43	61.51	82.51
65	5	34	88	76	96	99	45	82	85	49
-	-	-	-	-	-	-	-	-	-	-
7.888	11.929	16.07	20.37	24.88	29.70	35.02	41.26	49.32	61.40	83.05
12	3	71	36	05	09	93	9	85	37	2
-	-	-	-	-	-	-	-	-	-	-
7.897	11.946	16.10	20.41	24.92	29.75	35.06	41.25	49.23	61.29	83.60
88	6	41	17	84	16	51	81	67	5	05
-	-	-	-	-	-	-	-	-	-	-
7.904	11.958	16.12	20.44	24.97	29.79	35.10	41.25	49.16	61.19	84.16
21	8	45	26	01	94	43	86	08	24	17
-	-	-	-	-	-	-	-	-	-	-
7.907	11.966	16.13	20.46	25.00	29.84	35.14	41.26	49.09	61.09	84.73
53	2	87	63	49	27	43	78	91	58	68

-	-	-	-	-	-	-	-	-	-	-
7.908	11.969	16.14	20.48	25.03	29.88	35.18	41.28	49.04	61.00	85.32
35	6	73	32	25	05	34	32	97	51	74
-	-	-	-	-	-	-	-	-	-	-
7.907	11.969	16.15	20.49	25.05	29.91	35.21	41.30	49.01	60.92	85.93
22	8	11	39	33	23	99	28	1	02	48
-	-	-	-	-	-	-	-	-	-	-
7.904	11.967	16.15	20.49	25.06	29.93	35.25	41.32	48.98	60.84	86.56
65	4	11	93	79	81	31	48	15	08	07
-	-	-	-	-	-	-	-	-	-	-
7.901	11.963	16.14	20.50	25.07	29.95	35.28	41.34	48.95	60.76	87.20
16	2	81	04	68	79	22	78	97	66	65
-	-	-	-	-	-	-	-	-	-	-
7.897	11.957	16.14	20.49	25.08	29.97	35.30	41.37	48.94	60.69	87.87
18	9	3	81	1	24	7	07	43	76	42
-	-	-	-	-	-	-	-	-	-	-
7.893	11.952	16.13	20.49	25.08	29.98	35.32	41.39	48.93	60.63	88.56
07	1	65	33	13	2	74	28	41	33	56
-	-	-	-	-	-	-	-	-	-	-
7.889	11.946	16.12	20.48	25.07	29.98	35.34	41.41	48.92	60.57	89.28
11	2	95	69	88	75	37	34	82	37	29
-	-	-	-	-	-	-	-	-	-	-
7.885	11.940	16.12	20.47	25.07	29.98	35.35	41.43	48.92	60.51	90.02
51	6	24	96	41	95	61	21	57	83	81
-	-	-	-	-	-	-	-	-	-	-
7.882	11.935	16.11	20.47	25.06	29.98	35.36	41.44	48.92	60.46	90.80
42	6	56	2	8	89	51	88	58	7	39
-	-	-	-	-	-	-	-	-	-	-
7.879	11.931	16.10	20.46	25.06	29.98	35.37	41.46	48.92	60.41	91.61
89	3	95	47	12	62	1	32	78	95	28
-	-	-	-	-	-	-	-	-	-	-
7.877	11.927	16.10	20.45	25.05	29.98	35.37	41.47	48.93	60.37	92.45
96	9	43	78	41	2	43	54	13	55	78
-	-	-	-	-	-	-	-	-	-	-
7.876	11.925		20.45	25.04	29.97	35.37	41.48	48.93	60.33	93.34
59	3	-16.1	18	73	7	56	56	57	49	21
-	-	-	-	-	-	-	-	-	-	-
7.875	11.923	16.09	20.44	25.04	29.97	35.37	41.49	48.94	60.29	94.26
75	4	67	68	09	15	52	37	07	75	92
-	-	-	-	-	-	-	-	-	-	-
7.875	11.922	16.09	20.44	25.03	29.96	35.37	41.50	48.94	60.26	95.24
35	3	43	27	53	6	36	01	6	29	31
-	-	-	-	-	-	-	-	-	-	-
7.875	11.921	16.09	20.43	25.03	29.96	35.37	41.50	48.95	60.23	96.26
32	8	28	96	05	06	11	49	14	11	8
-	-	-	-	-	-	-	-	-	-	-
7.875	11.921	16.09	20.43	25.02	29.95	35.36	41.50	48.95	60.20	97.34
57	8	2	75	66	57	81	83	67	17	87

-	-	-	-	-	-	-	-	-	-	-
7.876	11.922	16.09	20.43	25.02	29.95	35.36	41.51	48.96	60.17	98.49
02	2	19	61	36	13	48	05	18	48	06
-	-	-	-	-	-	-	-	-	-	-
7.876	11.922	16.09	20.43	25.02	29.94	35.36	41.51	48.96	60.14	99.69
6	9	22	55	14	76	14	18	66	99	96
-	-	-	-	-	-	-	-	-	-	-
7.877	11.923	16.09	20.43	25.01	29.94	35.35	41.51	48.97	60.12	100.9
23	7	29	55	99	44	8	23	1	71	82
-	-	-	-	-	-	-	-	-	-	-
7.877	11.924	16.09	20.43	25.01	29.94	35.35	41.51	48.97	60.10	102.3
87	6	38	59	91	2	49	22	5	62	46
-	-	-	-	-	-	-	-	-	-	-
7.878	11.925	16.09	20.43	25.01	29.94	35.35	41.51	48.97	60.08	103.7
48	5	48	66	89	01	21	16	86	69	99
-	-	-	-	-	-	-	-	-	-	-
7.879	11.926	16.09	20.43	25.01	29.93	35.34	41.51	48.98	60.06	105.3
02	3	59	75	9	88	96	07	18	93	51
-	-	-	-	-	-	-	-	-	-	-
7.879	11.927	16.09	20.43	25.01	29.93	35.34	41.50	48.98	60.05	107.0
48	1	69	85	95	8	74	96	46	31	12
-	-	-	-	-	-	-	-	-	-	-
7.879	11.927	16.09	20.43	25.02	29.93	35.34	41.50	48.98	60.03	108.7
85	7	78	96	02	76	57	84	7	82	94
-	-	-	-	-	-	-	-	-	-	-
7.880	11.928	16.09	20.44	25.02	29.93	35.34	41.50	48.98	60.02	110.7
12	3	86	06	1	76	42	71	9	46	1
-	-	-	-	-	-	-	-	-	-	-
7.880	11.928	16.09	20.44	25.02	29.93	35.34	41.50	48.99	60.01	112.7
31	7	93	15	19	78	32	59	07	22	76
-	-	-	-	-	-	-	-	-	-	-
7.880	11.928	16.09	20.44	25.02	29.93	35.34	41.50	48.99	60.00	115.0
42	9	98	23	27	81	24	46	21	07	09
-	-	-	-	-	-	-	-	-	-	-
7.880	11.929	16.10	20.44	25.02	29.93	35.34	41.50	48.99	59.99	117.4
46	1	02	3	36	86	19	35	33	03	28
-	-	-	-	-	-	-	-	-	-	-
7.880	11.929	16.10	20.44	25.02	29.93	35.34	41.50	48.99	59.98	120.0
46	2	05	35	43	92	16	25	42	07	55
-	-	-	-	-	-	-	-	-	-	-
7.880	11.929	16.10	20.44	25.02	29.93	35.34	41.50	48.99	59.97	122.9
41	2	06	39	5	98	14	16	49	2	17
-	-	-	-	-	-	-	-	-	-	-
7.880	11.929	16.10	20.44	25.02	29.94	35.34	41.50	48.99	59.96	126.0
33	1	07	42	55	04	15	08	55	4	41
-	-	-	-	-	-	-	-	-	-	-
7.880		16.10	20.44	25.02	29.94	35.34	41.50	48.99	59.95	129.4
23	-11.929	06	43	59	09	16	01	59	67	6

-	-	-	-	-	-	-	-	-	-	-
7.880	11.928	16.10	20.44	25.02	29.94	35.34	41.49	48.99		133.2
13	9	05	44	63	14	18	96	62	-59.95	12
-	-	-	-	-	-	-	-	-	-	-
7.880	11.928	16.10	20.44	25.02	29.94	35.34	41.49	48.99	59.94	137.3
04	7	04	44	65	19	21	92	64	38	4
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.10	20.44	25.02	29.94	35.34	41.49	48.99	59.93	141.8
94	6	03	43	66	23	23	88	65	83	91
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.10	20.44	25.02	29.94	35.34	41.49	48.99	59.93	146.9
86	5	01	42	67	26	26	86	66	32	21
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928		20.44	25.02	29.94	35.34	41.49	48.99	59.92	152.4
79	4	-16.1	41	67	28	29	84	66	85	91
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.92	158.6
74	3	98	39	67	3	32	83	65	42	69
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.92	165.5
7	2	97	38	66	31	34	82	65	03	31
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.91	173.1
68	1	96	37	65	31	36	82	64	68	59
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.91	181.6
66	1	95	35	64	32	38	82	63	35	43
-	-	-	-	-	-	-	-	-	-	-
7.879		16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.91	191.0
66	-11.928	94	34	63	32	4	82	62	06	76
-	-	-	-	-	-	-	-	-	-	-
7.879		16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.90	201.5
66	-11.928	94	33	62	31	41	83	61	79	53
-	-	-	-	-	-	-	-	-	-	-
7.879		16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.90	213.1
67	-11.928	94	33	61	31	42	83	6	54	68
-	-	-	-	-	-	-	-	-	-	-
7.879		16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.90	226.0
68	-11.928	94	32	6	3	43	84	59	32	09
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.90	240.1
7	1	94	32	59	3	43	85	58	11	5
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.89	255.6
71	1	94	32	58	29	43	85	57	92	41
-	-	-	-	-	-	-	-	-	-	-
7.879	11.928	16.09	20.44	25.02	29.94	35.34	41.49	48.99	59.89	272.4
73	1	94	32	58	28	43	86	56	75	9

2) Plots

APPENDIX G MATLAB CODE

```
%%%Synchronous Machines Master's Thesis
%%% David C. Mazur P.Eng.
```

```
%%% define constants
```

```
ssfreq=120*pi;%steady state frequency
rstator=.26;%stator resistance
Xleakstator=.4297;
Xq=4.1469;
rprimekq=0.02;
Xprimeleakkq=.52778;
Xd=5.16477;
rfd=.13;
Xprimeleakfd=.79168;
rprimekd=.0224;
Xprimeleakkd=.5277;
```

```
Xmd=Xd-Xleakstator;
Xmq=Xq-Xleakstator;
Xprimekq=Xprimeleakkq+Xmq;
Xprimekd=Xprimeleakkd+Xmd;
Xprimefd=Xprimeleakfd+Xmd;
```

```
Lq=Xq/ssfreq; %Q axis inductance
Ld=Xd/ssfreq; %D axis inductance
Lmq=Xmq/ssfreq; %mutual Q axis inductance
Lmd=Xmd/ssfreq; %mutual D axis inductance
Lprimekq=Xprimekq/ssfreq; %Q Axis Damper Winding Inductance
Lprimefd=Xprimefd/ssfreq; %D Axis Damper Winding Inductance
Lprimekd=Xprimekd/ssfreq; %Field winding Inductance
Lleakstator=Xleakstator/ssfreq;
```

```
Vasn=480/sqrt(3);
```

```
%%%Calculate Initial conditions:
```

```
steps=1000000;
timestep=.00001;
timevector=0:(timestep):((timestep)*(steps-1));
```

```
vqs=zeros(steps,1);
vds=zeros(steps,1);
v0s=zeros(steps,1);
vkq=zeros(steps,1);
vfd=zeros(steps,1);
vkd=zeros(steps,1);
iqs=zeros(steps,1);
ids=zeros(steps,1);
i0s=zeros(steps,1);
ikq=zeros(steps,1);
ifd=zeros(steps,1);
ikd=zeros(steps,1);
ias=zeros(steps,1);
P=zeros(steps,1);
Q=zeros(steps,1);
```

```

S=zeros(steps,1);
pf=zeros(steps,1);
torque=zeros(steps,1);
omega=zeros(steps,1);
lambdads=zeros(steps,1);
lambdaq=zeros(steps,1);
domega=zeros(steps,1);
delta=zeros(steps,1);
ddelta=zeros(steps,1);
Tmech=zeros(steps,1);

iqs(1)=0;
ids(1)=0;
i0s(1)=0;
ikq(1)=0;
ikd(1)=0;
vqs(1)=Vasn;
vds(1)=0;
v0s(1)=0;
vkq(1)=0;
vkd(1)=0;
ifd(1)=vqs(1)/(ssfreq*Lmd);
vfd(1)=rfd*ifd(1);

for k=2:1:steps
vfd(k)=vfd(1);
end

omega(1)=ssfreq;
%J=35.1e6;
J=30.6;

for j =steps/10:steps
Tmech(j)= 130;
end

delta(1)=0;

C=zeros(6,6);
D=zeros(6,6);
Vmatrix=zeros(1,6);
dImatrix=zeros(1,6);
Imatrix=zeros(1,6);
deltaI=zeros(1,6);

%%Matrix Setup
for i=2:1:steps

```

```

lambdads(i)= (Ld*ids(i-1))+ (Lmd* ifd(i-1)) + (Lmd* ikd(i-1));
lambdaqs(i)= (Lq*iqs(i-1))+ (Lmq*(ikq(i-1)));

torque(i)=(1.5*2*((lambdads(i)*iqs(i-1))-lambdaqs(i)*ids(i-1)));

domega(i)=(Tmech(i)-torque(i))/J)*timestep;
omega(i)=domega(i)+(omega(i-1));

ddelta(i)=(omega(i)-ssfreq)*timestep;
delta(i)=delta(i-1)+ddelta(i);

if delta(i)>=(2*pi)
    delta(i)=delta(i)-(2*pi);

end

vqs(i)=Vasn*cos(delta(i));
vds(i)=Vasn*-1*sin(delta(i));
v0s(i)=0;

P(i)=torque(i)*omega(i);

C(1,1)=rstator;
C(1,2)=omega(i)*Ld;
C(1,5)=omega(i)*Lmd;
C(1,6)=omega(i)*Lmd;
C(2,1)=-1*omega(i)*Lq;
C(2,2)=rstator;
C(2,4)=-1*omega(i)*Lmq;
C(3,3)=rstator;
C(4,4)=rprimekq;
C(5,5)=rfd;
C(6,6)=rprimekd;

D(1,1)=Lq;
D(1,4)=Lmq;
D(2,2)=Ld;
D(2,5)=Lmd;
D(2,6)=Lmd;
D(3,3)=Lleakstator;
D(4,1)=Lmq;
D(4,4)=Lprimekq;
D(5,2)=Lmd;
D(5,5)=Lprimefd;
D(5,6)=Lmd;
D(6,2)=Lmd;
D(6,5)=Lmd;
D(6,6)=Lprimekd;

Vmatrix(1,1)=vqs(i);
Vmatrix(1,2)=vds(i);
Vmatrix(1,3)=v0s(i);

```

```

Vmatrix(1,4)=vkq(i);
Vmatrix(1,5)=vfd(i);
Vmatrix(1,6)=vkd(i);

Imatrix(1,1)=iqs(i-1);
Imatrix(1,2)=ids(i-1);
Imatrix(1,3)=i0s(i-1);
Imatrix(1,4)=ikq(i-1);
Imatrix(1,5)=ifd(i-1);
Imatrix(1,6)=ikd(i-1);

%dImatrix=(-1*inv(D)*C*(Imatrix)')+(inv(D)*(Vmatrix)')
G=(-1*inv(D)*C*(Imatrix)');
H=(inv(D)*(Vmatrix)');
dImatrix=(G+H);
deltaI=(dImatrix*timestep)';

iqs(i)=iqs(i-1)+(deltaI(1,1));
ids(i)=ids(i-1)+(deltaI(1,2));
i0s(i)=i0s(i-1)+(deltaI(1,3));
ikq(i)=ikq(i-1)+(deltaI(1,4));
ifd(i)=ifd(i-1)+(deltaI(1,5));
ikd(i)=ikd(i-1)+(deltaI(1,6));
ias(i)=(iqs(i)+(1i*ids(i)));
pf(i)=cos(angle(ias(i)));
end

figure(1)
subplot(3,1,1);
plot(timevector,iqs);
title('iqs vs time');
ylabel('iqs [A]');
xlabel('time [s]');

subplot(3,1,2);
plot(timevector,ids);
title('ids vs time');
ylabel('ids [A]');
xlabel('time [s]');

subplot(3,1,3);
plot(timevector,i0s);
title('i0s vs. time');
ylabel('i0s [A]');
xlabel('time [s]');

figure(2)
subplot(2,1,1)
plot(timevector,omega);
title('Omega vs Time');
ylabel('Omega [rad/s]');
xlabel('time [s]');

```

```

subplot(2,1,2)
plot(timevector, (delta*(180/pi)));
title('Delta vs Time');
ylabel('Delta [degrees]');
xlabel('time [s]');

figure(3)
subplot(2,1,1)
plot(timevector, Tmech);
title('Mechanical Torque vs. Time');
ylabel('Torque [N-m]');
xlabel('time [s]');

subplot(2,1,2)
plot(timevector, torque);
title('Electrical Torque vs. Time');
ylabel('Torque [N-m]');
xlabel('time [s]');

figure(4)
plot(timevector, P);
title('Power vs Time');
ylabel('Power [W]');
xlabel('time [s]');

figure(5)
plot(delta, torque);
title('Electrical Torque vs. Delta');
ylabel('Torque [N-m]');
xlabel('Delta [rad]');

figure(6)
plot(timevector, (Tmech-torque));
title('Difference in Torque');
ylabel('Torque [N-m]');
xlabel('time [s]');

figure(7)
subplot(2,1,1);
plot(timevector, vqs);
title('vqs vs. time');
ylabel('vqs [Volts]');
xlabel('time [s]');
subplot(2,1,2);
plot(timevector, vds);
title('vds vs. time');
ylabel('vds [Volts]');
xlabel('time [s]');

figure(8)
plot(timevector, ias);
title('Ias vs. time');
ylabel('Ias [A]');
xlabel('time [s]');

```

```

figure(9)
plot(timevector, pf);
title('power factor vs. time');
ylabel('power factor []');
xlabel('time [s]');
counter=1;

for p=1:10000:(length(delta))
    newdelta(counter,1)=delta(p);
    counter=counter+1;
end
save my_data.out newdelta -ASCII

```