

Old Dominion University

ODU Digital Commons

Mechanical & Aerospace Engineering Theses & Dissertations

Mechanical & Aerospace Engineering

Summer 2021

A Digital One Degree of Freedom Model of an Electromagnetic Position Sensor

Michelle Elizabeth Weinmann

Old Dominion University, michelle.e.weinmann@gmail.com

Follow this and additional works at: https://digitalcommons.odu.edu/mae_etds



Part of the [Aerospace Engineering Commons](#), [Computer Sciences Commons](#), and the [Electrical and Computer Engineering Commons](#)

Recommended Citation

Weinmann, Michelle E.. "A Digital One Degree of Freedom Model of an Electromagnetic Position Sensor" (2021). Master of Science (MS), Thesis, Mechanical & Aerospace Engineering, Old Dominion University, DOI: 10.25777/zg4c-ye02

https://digitalcommons.odu.edu/mae_etds/337

This Thesis is brought to you for free and open access by the Mechanical & Aerospace Engineering at ODU Digital Commons. It has been accepted for inclusion in Mechanical & Aerospace Engineering Theses & Dissertations by an authorized administrator of ODU Digital Commons. For more information, please contact digitalcommons@odu.edu.

A DIGITAL ONE DEGREE OF FREEDOM MODEL
OF AN ELECTROMAGNETIC POSITION SENSOR

by

Michelle Elizabeth Weinmann
B.A. May 2012, St. John's College of Annapolis

A Thesis Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

AEROSPACE ENGINEERING

OLD DOMINION UNIVERSITY

August 2021

Approved by:

Colin P. Britcher (Director)

Robert Ash (Member)

Tian-Bing Xu (Member)

Mark Schoenenberger (Member)

ABSTRACT

A DIGITAL ONE DEGREE OF FREEDOM MODEL OF AN ELECTROMAGNETIC POSITION SENSOR

Michelle Elizabeth Weinmann
Old Dominion University, 2021
Director: Dr. Colin P. Britcher

The purpose of this project was to improve an existing system currently in use by NASA Langley Research Center (LaRC). The 6-inch Magnetic Suspension and Balance System (MSBS) built at MIT is operational with control in three degrees of freedom, with two additional degrees of freedom exhibiting passive stability. The means for measuring model displacement within the magnetic environment is an Electromagnetic Position Sensor (EPS), consisting of excitation coils at 20 kHz and multiple sets of pickup coils. The pickup coil voltages are proportional to model displacement in each degree of freedom. However, the EPS electronic signal processing system is analog and outdated; setup and adjustment are time consuming. The task was to construct a one degree of freedom model of the EPS including its electronics system in order to explore digital signal processing. The model core is allowed only axial displacement for simplicity; this model coil set is essentially a Linear Variable Differential Transformer (LVDT). The source signal was chosen to be 2.36 kHz for convenience, with scaling up to the full size system at 20 kHz possible. An amplification and filtering circuit board was constructed to modify the signal for the proper functionality of the model EPS. By comparing the reference or excitation coil signal and the measured or pickup coil signal by means of a digital phase measurement method using cross-correlation analysis, the digital algorithm resolves the phase shift between the two signals and their amplitude ratio. The key proof of concept is the digital signal processing algorithm; since the defining characteristics of any signal are the amplitude and phase this algorithm can be adapted to suit various control and setup needs of the MSBS and EPS. By proving that a digital interface with the EPS is possible, the analog interface can be replaced with a digital system.

Copyright, 2021, by Michelle Elizabeth Weinmann, All Rights Reserved

ACKNOWLEDGEMENTS

This work has been completed thanks to the contributions of many others. To my advisor Dr. Britcher I extend my deepest gratitude for his patience, guidance, and mentorship over the years I have worked on this project. I would also like to thank Timothy Schott, the electronics expert in our research team at NASA Langley, for his continuous support in the lab. My appreciation is extended to Mark Schoenenberger, the lead of our research team, and to Dr. Robert Ash and Dr. Tian-Bing Xu for serving on my thesis committee and for their support during my graduate studies.

TABLE OF CONTENTS

	Page
LIST OF FIGURES	vii
LIST OF TABLES	viii
Chapter	
1. INTRODUCTION	1
1.1 THEORETICAL FORMULATIONS	4
1.2 PURPOSE	13
1.3 PROBLEM	13
2. SOFTWARE EMPLOYED IN RESEARCH	15
2.1 COMSOL	15
2.2 MATLAB	17
2.3 LABVIEW	18
3. METHODOLOGY	19
3.1 MODEL DESIGN	19
3.1.1 CONDITIONING CIRCUIT	21
3.1.2 COMSOL MODEL	29
3.2 DATA COLLECTION PROCEDURE	31
3.3 ALGORITHM DESIGN	32
3.3.1 SIMULATION ALGORITHM	32
3.3.1.1 NOISE ANALYSIS	35

3.3.1.2 SAMPLE RATE ANALYSIS	43
3.3.1.3 RECORD LENGTH ANALYSIS	46
3.3.1.4 QUANTIZATION ANALYSIS	46
3.3.1.5 MIXED ERROR SOURCE ANALYSIS	49
4. RESULTS	52
4.1 COMSOL ANALYSIS	52
4.2 DIGITAL PROCESSING ALGORITHM	57
4.3 DATA ANALYSIS	58
4.4 ALGORITHM RESULTS	63
4.5 DESCRIPTIVE ANALYSIS	70
4.5.1 QUANTIZATION ANALYSIS	70
4.5.2 SAMPLE RATE ANALYSIS	72
4.5.3 RECORD LENGTH ANALYSIS	76
4.5.4 NOISE ANALYSIS	77
4.5.5 MIXED ERROR SOURCE ANALYSIS	79
4.6 REAL TIME CODE	83
5. CONCLUSIONS AND FUTURE WORK	86
5.1 CONCLUSION	86
5.2 FUTURE WORK	86
REFERENCES	87
APPENDICES	88

A. CORE DISPLACEMENT SYSTEM	88
B. MEASUREMENT DETAILS	94
C. LABVIEW CODES	95
D. ALGORITHM CODE	99
D.1 MATLAB SIMULATION SCRIPTS	99
D.2 SIMULATION SIGNAL GRAPHS	106
D.3 MATLAB DATA PROCESSING SCRIPTS	114
E. GRAPHS OF PICKUP SIGNALS, RAW DATA	127
F. DISPLACEMENT DIRECTION DIFFERENCE	130
VITA	135

LIST OF FIGURES

Figure	Page
1 MIT EPS Construction	4
2 MIT EPS Coil Winding Schematic	5
3 EPS Coupling Variation with Axial Position of Core	7
4 Photo of the 1 Degree of Freedom EPS	8
5 Schematic of Entire Signal Conditioning System	8
6 Amplitude ratio of source vs. amplitude ratio of measured signal	11
7 Demodulation Procedure Schematic	12
8 Magnetic Flux of EPS Windings	15
9 Schematic of 1 Degree of Freedom EPS Coil Form (in)	20
10 Schematic of Entire Signal Conditioning System	22
11 Schematic of EVAL-CN0301 LVDT Conditioning Circuit Board	22
12 Frequency Response of LTC1060	27
12 Schematic of Conditioning Circuit	28
13 EPS Conditioning Circuit PCB Design	29
14 Surface Magnetic Flux Density Norm (T) of Cross-section of 1D EPS in COMSOL	30
15 Graphs of Simulated Amplitude Ratio and Phase Difference	34
16 Comparison of Random Noise: 10% & 20% of Maximum Pickup Amplitude	36
17 Random Noise Level Standard Deviation (V)	37
18 Random Noise Level Deviation	38
19 Comparison of Ripple Noise: 0.05 V peak and 0.1 V Peak Amplitude	39

20	Current Ripple Noise Level Standard Deviation (V)	40
21	Current Ripple Noise Level Deviation	41
22	Noise Comparison: Random & Current Ripple Mixed: 0.05 V Peak & 0.1 V Peak	42
23	Mixed Noise Level Standard Deviation (V)	42
24	Mixed Noise Level Deviation	43
25	Graphs of Signals: Downsampling at Half Sample Rate Comparison	44
26	Graphs of Signals: Downsampling at One Fourth Sample Rate Comparison	45
27	Comparison of Different Sample Rates	45
28	Graphs of Amplitude Ratio and Phase Difference: Record Length One Cycle	46
29	3-bit Quantized Signal with 8 Levels	47
30	Comparison of Bit Levels	48
31	Deviation of Bit Values	49
32	Comparison of Quantization and Downsampling Errors: 0.1 V Noise	50
33	Deviation of Quantization and Downsampling Errors at 0.1 V Noise	51
34	Comparison of Core Materials	53
35	Comparison of Steel Voltage Magnitude: COMSOL vs. Measured	55
36	Deviation of Steel Voltage Magnitude: COMSOL vs. Measured	56
37	Data graphs as model core displaces 11 mm, Pickup Amplitude Detail	60
38	Data graphs as model core displaces 11 mm, Phase Difference	62
39	Raw Measured Relative Amplitude	64
40	Nulled Relative Amplitude	65
41	Phasor Representation of Background Signal Addition	65
42	Corrected Relative Amplitude with Linear Fit	67

43	Phase Difference and Relative Amplitude Graphs	69
44	Quantization Error of Relative Amplitude Graphs	71
45	Deviation from 16 Bit	72
46	50 kHz vs. 25 kHz	73
47	50 kHz vs. 5 kHz Sample Rate	74
48	Downsampling Comparison of Relative Amplitude	75
49	Downsampling Comparison Deviation	75
50	Window Length Comparison	76
51	Window Length Comparison Deviation	77
52	Set Relative Amplitude Comparison	78
53	Set Comparison Standard Deviation (V)	78
54	Mixed Error Sources Relative Amplitude Comparison: 12 bits	80
55	Mixed Error Sources Deviation: 12 bits	81
56	Mixed Error Sources Relative Amplitude Comparison: 10 bits	82
57	Mixed Error Sources Deviation: 10 bits	82
58	Time Analysis of Real Time Code	84
59	Real Time Code	85

LIST OF TABLES

Table		Page
1	Table of Resistor Values	26
2	Table of Relative Amplitude and Phase Difference Values	69

CHAPTER 1

INTRODUCTION

While the technology has existed for many years, Magnetic Suspension and Balance Systems (MSBS) have yet to reach a significant level of popularity among wind tunnel experimentalists. Most prefer a traditional wind tunnel configuration with a mechanically supported model using struts or stings. However, to produce interference-free data in a wind tunnel, it is necessary to levitate the model. There are two sources of error caused by mechanical supports which the MSBS seeks to rectify: the flow distortions over the model and the distortion of the flow pattern throughout the test section.[1] This interference increases with increasing pressure in the tunnel and with aircraft innovation; the range of movement of the model is also limited.[2] The MSBS was theorized as a system which would employ magnetic suspension to allow for uninhibited aerodynamic analysis. The first MSBS was built in 1957 but feedback controllers were inadequate until around 1980 when high speed computing became a reality. The 6-inch five degree of freedom MSBS was built from 1966 to 1969 by MIT for NASA Langley Research Center (LaRC) to operate in conjunction with the Electromagnetic Position Sensor (EPS) for a low-speed wind tunnel operating at a maximum Mach number of around 0.5.[3][4] The EPS was needed to act as a sensor for feedback control of the model within the MSBS. The levitated model is inherently unstable, therefore the EPS was constructed as a sensor providing the model's position and attitude for a closed-loop stabilization system.[5] The EPS is unique to the 6 inch MSBS, all other suspension wind tunnel systems currently use optical methods.[4] It acts as a multicoil linear variable differential transformer which provides position and attitude information for the feedback control system.[2][6] The EPS also provides a greater range of measurement of model displacement.[4] While the control system of the 6-inch MSBS has been fully digitized, the conditioning of the EPS is analog.

The existing EPS consists of a transducer coil assembly including Helmholtz excitation coils and seven pairs of oppositely wound pickup coils which sense one axial, three vertical

plane, and three horizontal plane position components due to the coil geometry.[4][7] The geometry of the coils also results in the voltages induced in each pickup coil to mostly cancel when no model is within the EPS or when the model core is centered.[4][8] The excitation coils are driven by 20 Volt power amplifiers which results in terminal voltage of 200 Volts peak to peak at 20 kHz. Multiple channels correspond to each degree of freedom, each of which requires its own amplifiers and demodulators. This forms a complex system of amplifiers, demodulators, and filters which ensure each signal can be compared with minimal noise interference.[7] In order to begin to operate the MSBS the EPS must first be properly aligned. Each pickup signal must be nulled, or made to have zero amplitude. When a model is present in the EPS the magnetic coupling between the excitation and pickup coils is distorted, resulting in a voltage difference on the pickup coils. The amplitudes of these differential voltages on the pickup coil pairs depend on the model's position in each direction.[4]

In order to glean motion data from the EPS, the amplitude gain and phase shift must be obtained from the differential voltages on the pickup coils. Gain signifies the signal's amplitude and corresponds to displacement of the model in the EPS. Phase signifies the delay in time (or angle) at which the peaks of the signal occur and corresponds to the direction of motion of the model in the EPS. The adjustment of the gain and phase shift is required to equalize each input from pairs of directional pickup coils to the EPS conditioning circuitry. There are then routine adjustments that must be made once the levitation magnets are turned on since the magnetic environment interferes with the EPS. Further nulling and interference cancellation is required to ensure accuracy of the gain and phase shift which correspond to core displacement.[7] This process is time consuming and is difficult to execute unless one has done it before. Since the technological trend has been towards digitization of the MSBS, corresponding digitization of the functionality of the EPS is logical. One person can then control the composite systems with software on one computer, streamlining the conditioning and processing of the EPS.

There are many applications for using an EPS/MSBS system for wind tunnel testing. The geometry of the model will not require modification in order to be supported, instead newer and more innovative aircraft, entry vehicles, and ballistics with more intricate or unusual geometries will be able to be experimentally tested. Dynamic stability testing for re-entry vehicles is one such application that is receiving attention at NASA LaRC, especially since the levitated model can freely oscillate.[9]

1.1 THEORETICAL FORMULATIONS

The theory of operation of the MIT EPS is based upon the geometry of the Helmholtz excitation coils and pickup coils. A schematic of the coil construction is included in Figure 1 [7].

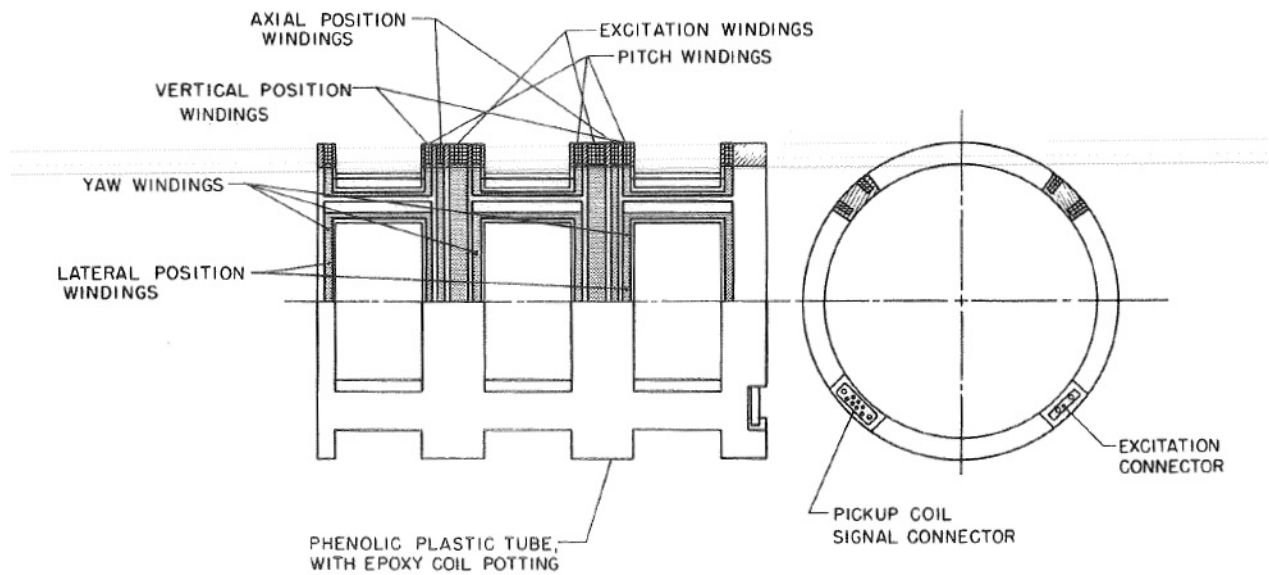


Figure 1: MIT EPS Construction

The excitation coil windings carry a 20 kHz current producing an oscillating axial magnetic field which remains uniform throughout the region surrounding the geometric center. The core, consisting of a ferromagnetic and/or electrically conducting material, becomes periodically magnetized at the excitation frequency when placed into the magnetic field; therefore the far field of the core fluctuates as well. The fluctuating far field component can be detected with electrical coils, previously referenced as the pickup coils.

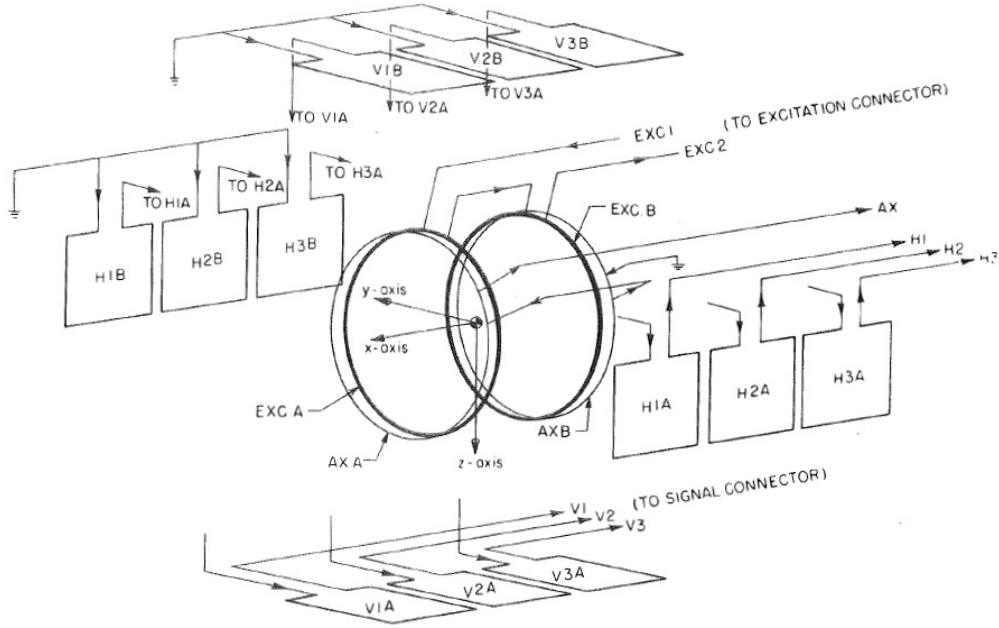


Figure 2: MIT EPS Coil Winding Schematic

Figure 2 shows a depiction of the transducer coil windings of the original MIT EPS [7]; the geometry of the pickup coils was designed to enable a null received signal when no model is present or when the model is at the geometric centroid of the EPS windings.

The coupling of the magnetic field of the model core with the pickup coils is due to two transmission mechanisms depending on the material of the model core. If the core is soft iron then fluctuating magnetization of the model is responsible for the coupling; if the core is conductive then induced eddy currents within the surface are responsible for the coupling. The two phenomena are detailed in the equations below. The definitions of each term are as follows: B is the magnetic flux density which is measured in Teslas (T), M is the magnetization field measured in amperes per meter ($\frac{A}{m}$), I is electric current measured in amperes (A), and V is voltage measured in volts (V). The subscripts are as follows: PA for power amplifier, E for excitation, M for the magnetized iron core transmission mechanism, and C for the conductive core transmission mechanism. For example, I_E signifies the current in the excitation coils and B_M signifies the magnetic flux density produced by the magnetized soft iron core. Constants of proportionality are omitted. The first equation describes the

magnetization coupling case; it indicates that the magnetic flux density produced by the magnetized iron core, magnetized core transmission mechanism, magnetic flux density produced by the excitation coils, current in the excitation coils, and voltage supplied by the power amplifier are all proportional. The second describes the eddy current coupling case; the magnetic flux density produced by the conductive core transmission mechanism, the current in the excitation coils, the rate of change of the magnetic flux density of the excitation coils, the rate of change of the current in the excitation coils, and the rate of change of the voltage supplied by the power amplifier are all proportional. The third relates the rates of change of the magnetic flux density in either transmission case to the output voltage.

$$B_M \propto M_E \propto B_E \propto I_E \propto V_{PA} = V_0 \sin(\omega t) \quad (1)$$

$$B_C \propto I_E \propto \frac{dB_E}{dt} \propto \frac{dI_E}{dt} \propto \frac{dV_{PA}}{dt} = V_0 \cos(\omega t) \quad (2)$$

$$V_{out} \propto \frac{dB_M}{dt} \text{ or } \frac{dB_C}{dt} \quad (3)$$

In general, both mechanisms occur simultaneously in some combination. Yet the main difference between the physics of transmission for soft iron versus a conductive material is that the conductive material is proportional to the rates of change of the excitation magnetic flux density, the excitation current, and the voltage on the power amplifier; similarly the signal of the conductive core is a cosine wave, the derivative of a sine wave. The two transmission mechanisms are also 90 degrees apart from each other. However, in both cases the transmitted signal to the pickup coils is caused by the rate of change of radiated fields whether they be magnetized or conducted. [10] A visual representation of the changing magnetic field coupling is included in Figure 3.

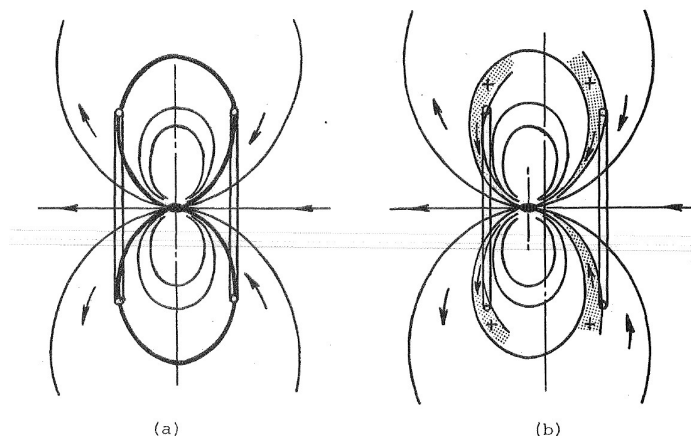


Figure 3: EPS Coupling Variation with Axial Position of Core

For the purposes of this thesis, examining transmission in all six degrees of freedom is too complex. Therefore, since the axial coupling has the simplest transmission mechanism, the axial direction was chosen to be the only degree of freedom for the model EPS operation in this study. Restraining the model to one direction of movement allows the EPS to be treated as a Linear Variable Differential Transformer (LVDT) with a null center. When the model displaces from the center more flux couples with the nearer pickup coil and less flux couples with the farther pickup coil resulting in a net output voltage which varies proportionally to the axial displacement. The phase of the received signal indicates the displacement direction with zero degrees or positive 90 degrees corresponding to forward motion and 180 degrees or negative 90 degrees corresponding to backwards motion.[7]

The one degree of freedom test apparatus of the EPS using only axial displacement was 3D printed previously. Coil construction was necessary, therefore winding two sets of copper coils on the existing form was carried out. The excitation coils consist of 30 turns of 18 gage copper magnet wire wound in the same direction. The pickup coils consist of 30 gage enameled copper wire wound in opposition; there are two layers of windings with 30 turns each in order to have access to both the 30 turn and 60 turn signal, the latter being nominally twice as large. A photo of the coil form with windings is included in Figure 4.

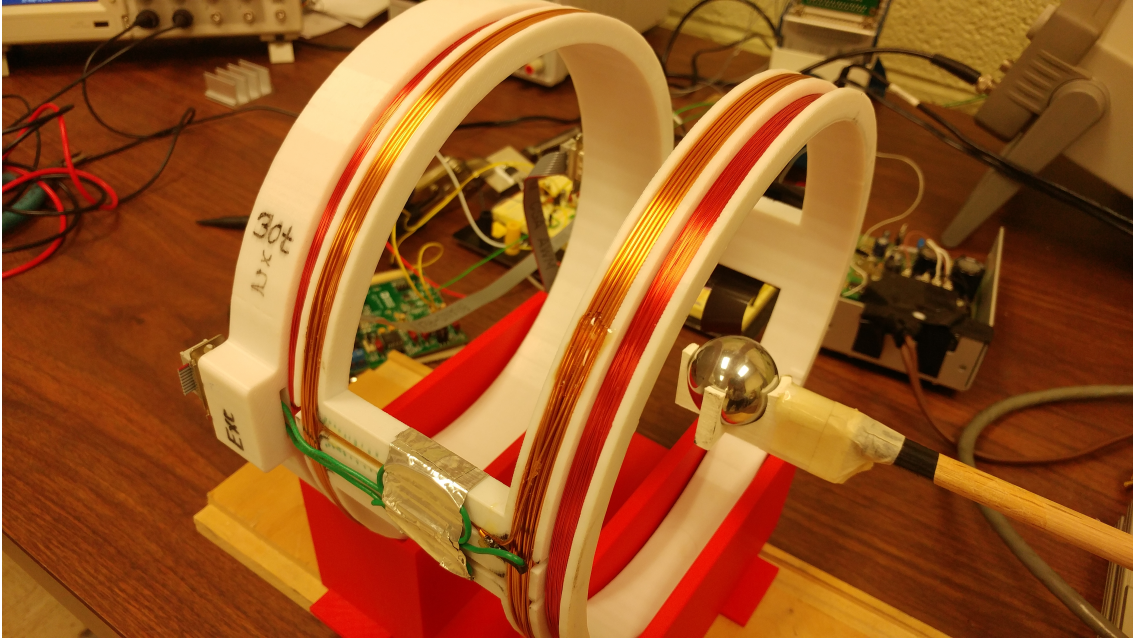


Figure 4: Photo of the 1 Degree of Freedom EPS

The system built for this thesis includes the test apparatus, an EPS conditioning circuit, and a stepper motor for precise model motion in millimeters through the axis of the EPS. Since the one degree of freedom EPS essentially operates as a transformer, the conditioning circuit is necessary to supply enough voltage and current to the excitation coils to transmit a strong enough signal to the pickup coils. Below is a high level diagram including all system components.

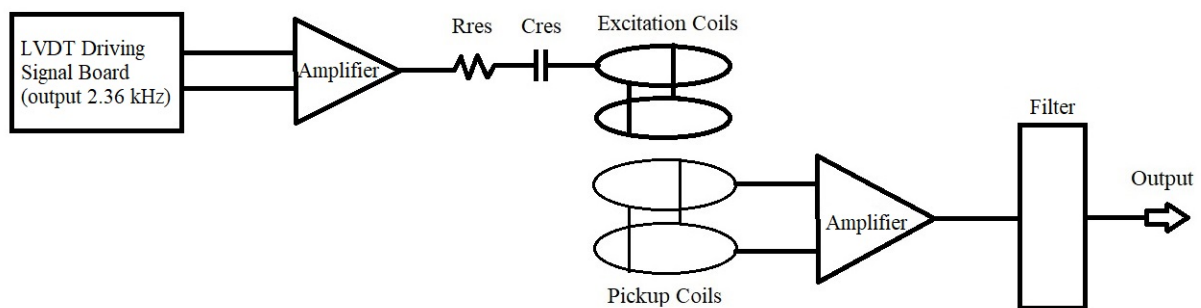


Figure 5: Schematic of Entire Signal Conditioning System

However, the signal output of this system is analog and is required to be digitized in order to perform real-time digital analysis. In order to digitize the calculations of the gain

and the phase of the signal an algorithm using cross-correlation analysis is necessary. Cross-correlation compares the measured signal (the pickup signal) to the reference signal (the excitation signal) at a specified time lag which in this case is zero since the signals are produced simultaneously.[11] As the model core moves through the coils, the measured signal will change in phase and amplitude. This changing measured signal is compared to the stationary reference signal at multiple points along the core's displacement. The maximum correlation will be when the signals are in phase and match amplitudes; the more the signals resemble each other the higher the correlation will be. This method is useful due to the noise rejection inherent in correlation; since noise typically does not resemble the reference signal there would be no correlation. [11]

The mathematical definition of cross-correlation at zero time lag is

$$R_{xy}(0) = \frac{1}{T} \int_0^T x(t)y(t)dt \quad (4)$$

where $x(t)$ is the reference signal, $y(t)$ is the measured signal, and T is the length of signal record.

The auto-correlation is the same operation but instead of comparing two different signals the autocorrelation compares a signal to itself. Hence, the mathematical definition of auto-correlation at zero time lag is

$$R_x(0) = \frac{1}{T} \int_0^T x(t)^2 dt \quad (5)$$

$$R_y(0) = \frac{1}{T} \int_0^T y(t)^2 dt \quad (6)$$

The auto-correlation is useful because once the autocorrelation is determined, the peak amplitude is related to it as

$$R_x(0) = \frac{A_x^2}{2} \quad (7)$$

$$R_y(0) = \frac{A_y^2}{2} \quad (8)$$

Using the above equations to extract the amplitude ratio between the two signals allows for the displacement of the model to be calculated.

The phase difference between the measured and reference signals is expressed by

$$\phi_y = \arccos \frac{R_{xy}(0)}{\sqrt{R_x(0)R_y(0)}} \quad (9)$$

The phase information describes the direction of motion of the model. While the phase is roughly constant for each model core material, there is a 180 degree phase shift, or sign reversal, when the model crosses through the null point. For the analog system a phase value between 0 and 90 degrees corresponded to the core moving towards the center of the EPS; a phase value between 90 and 180 degrees corresponded to the core moving away from the center of the EPS. These equations are the basis of the digital algorithm serving as the phase and amplitude measuring system.[11][12]

Theoretically, the analysis of the amplitude ratio as the model displaces should show a linear relationship. The figure below is the representation of simulated signal data using a Fast Fourier Transform (FFT) algorithm for processing.[9] The FFT is a popular digital signal processing scheme utilizing the discrete Fourier transform to convert a signal from the time domain into a frequency domain representation. The signal is decomposed into component frequencies; analysis consists of the amplitudes of each component frequency. These amplitudes are extracted from the FFT analysis; the simulated signals for each data point begin at a negative maximum and decrease in amplitude until the signal has zero amplitude, then switch sign to increase in amplitude until the positive maximum is reached. The amplitude of the signal at each data point is proportional to the displacement of the model core. The expected linear relationship is shown below in Figure 6.

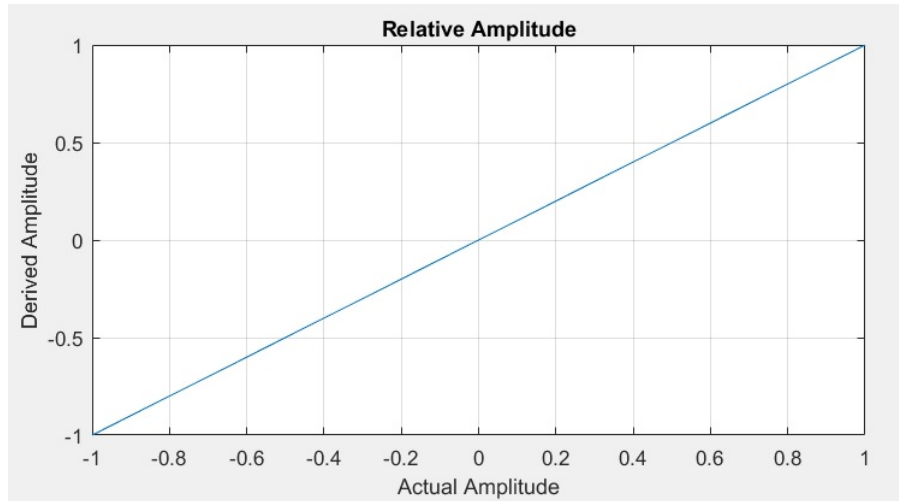


Figure 6: Amplitude ratio of source vs. amplitude ratio of measured signal

The performance of the algorithm can be measured by the degree of linearity of the relationship: the straighter the line, the better the performance. A goal of this research is to have as linear a representation as possible. However, a slight non-linearity in the center of the line around the origin is likely to occur due to the signal approaching zero amplitude; this is the expected behavior for the amplitude ratio around the origin. This non-linearity can be exacerbated by error sources in the demodulation including quantization, discretization, and noise sensitivity. A schematic of the demodulation procedure is included in Figure 7. First, the continuous signals are discretized with a data acquisition unit. Then the digital signals are imported into a Matlab script for processing. The cross-correlation and autocorrelation techniques described above extract the amplitude and phase information for both the reference and pickup signals. The ratio of the amplitude of the pickup signal to the amplitude of the reference signal is then computed for each displacement position of the model core within the EPS. This results in a graph as in Figure 6. The relative phase is computed between the two signals for each displacement position as well. This information is graphed as well to demonstrate the phase shift at the null center.

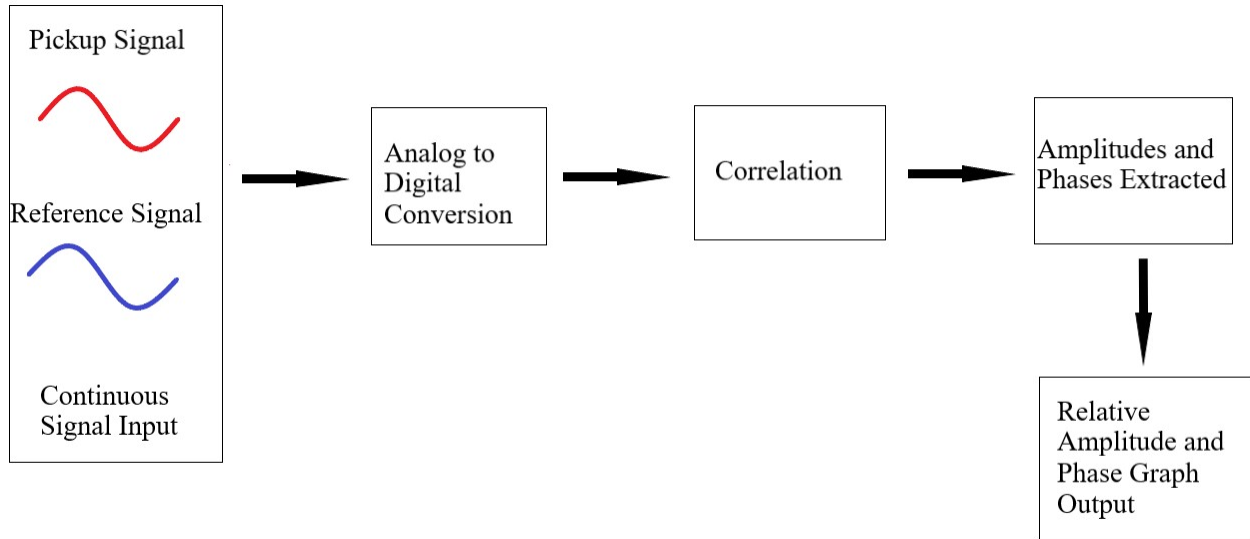


Figure 7: Demodulation Procedure Schematic

1.2 PURPOSE

The purpose of this study is to prove possible the concept of a digital EPS conditioning system. Since the EPS is a unique device, the conditioning system that provides the displacement measurements must allow for the electromagnetic environment within which the EPS exists. A characteristic problem with the MSBS and EPS system is an abundance of electrical noise, therefore it is necessary that the digital algorithm employing cross-correlation reduces noise contamination. The other benefits of digitization of the EPS conditioning system include: decreased set up time, simpler displacement measuring, and access to the digital data output from the EPS. Ultimately, it is the hope of the team at NASA Langley to adapt this model's conditioning algorithm to the five degree of freedom MSBS and its EPS. The processing algorithm will be carried out in real time in order to provide position feedback and enhance the controls of the MSBS. There are four performance metrics of the processing algorithm that must be evaluated in order for digitization to occur; this thesis will test these qualities to ascertain the requirements of the digitization algorithm. First, the minimum number of bits necessary to represent the waveform output of the pickup and excitation signals must be found. Second, the minimum sample rate required; third, the minimum record length required; and fourth, the effect of noise.

1.3 PROBLEM

To construct a model to illustrate the digitization of the EPS some simplifications were made: model motion was restricted to axial in millimeters, frequency was scaled down to 2.36 kHz from 20 kHz, and a steel 1 inch diameter ball was used as the model core. The frequency was scaled down to 2.36 kHz for convenience to match a commercial LVDT driving signal circuit. Since the model EPS allows only axial motion it operates as an LVDT, for which there are existing signal conditioning systems available commercially off the shelf.

The breakdown of the EPS is as follows: excitation, preamplification, signal combination, phase-sensitive demodulation, final amplification, position indication, and reference amplification.[7] Due to the simplifying assumptions made, only excitation, preamplification, and

phase-sensitive demodulation are necessary for the model EPS. The excitation and preamplification of the model EPS are analog; the driving signal must be amplified in order to provide enough voltage to the excitation coils. The induced signal in the pickup coils is very small and must also be amplified and filtered in order to extract the signal with accuracy. It is then necessary to record the processed output as digital data to be post-processed. The phase-sensitive demodulation was developed by means of a MATLAB algorithm employing cross-correlation analysis. Since the digital phase measurement algorithm must be used in real time in the future, this algorithm must be robust enough to handle different sample rates, noise levels, and record lengths to be efficient enough for high speed and real time execution.

CHAPTER 2

SOFTWARE EMPLOYED IN RESEARCH

In order to formulate accurate simulations, three different software were used each with a unique purpose. COMSOL, a multiphysics finite-element analysis software, was used to compare voltage data acquired with theoretical computed voltages. MATLAB was used as the environment for the amplitude and phase measurement algorithm. LabView was used as the driving software behind the stepper motor which allowed for precise core displacement increments. Each package will now be briefly described.

2.1 COMSOL

COMSOL Multiphysics is an all-in-one modeling and simulation platform with additive physics capabilities. It allows the user to define geometries, material properties, and the physics equations associated with the model. This platform was used to predict and verify the voltage levels obtained from the pickup coils as the model core is displaced one millimeter at a time. Once the geometry of the one degree of freedom model EPS was input into the system, material properties and other parameters are specified according to the physical characteristics of the model. A cross-sectional view of the COMSOL representation of magnetic flux on each coil and on the spherical core in the center position is shown in Figure 8.

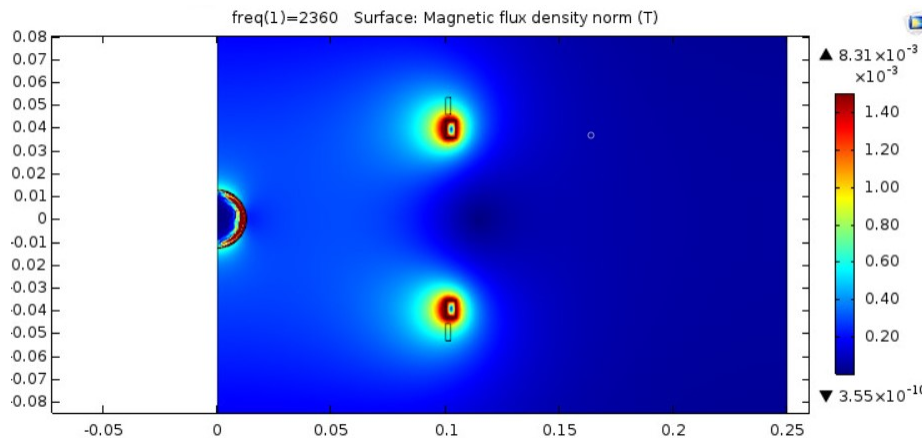


Figure 8: Magnetic Flux of EPS Windings

In COMSOL's AC/DC module, the magnetic field and electric circuit physics were utilized in order to replicate electromagnetic phenomena. Maxwell's equations and Ampere's Law are used to model the magnetic field, utilizing the following values: \mathbf{J}_e is electric current density, \mathbf{A} is magnetic vector potential, \mathbf{B} is the magnetic flux density or magnetic field, \mathbf{D} is the displacement field, \mathbf{E} is the electric field, and \mathbf{H} is the magnetic field strength, all of which are vector fields or pseudovector fields. Magnetic vector potential \mathbf{A} can be visualized using the magnetic field \mathbf{B} : for example if the B-field is oriented in the z-direction, the vector potential \mathbf{A} rotates around the z-axis. Another way to understand it is that \mathbf{A} differentiated is \mathbf{B} , or the rate of change of \mathbf{A} is \mathbf{B} .

$$\mathbf{J}_e = (j\omega\sigma - \omega^2\epsilon_0\epsilon_r)\mathbf{A} + \nabla x(\mu_0^{-1}\mu_r^{-1}\mathbf{B}) - \sigma\mathbf{v} \times \mathbf{B} \quad (10)$$

$$\mathbf{B} = \nabla \times \mathbf{A} \quad (11)$$

$$\mathbf{B} = \mu_0\mu_r\mathbf{H} \quad (12)$$

and the equation used to model the electric field is

$$\mathbf{D} = \epsilon_0\epsilon_r\mathbf{E} \quad (13)$$

The constant ϵ_0 is the electric permittivity of free space; its value is 8.85×10^{-12} F/m, Farads per meter. Similarly, the constant μ_0 is the permeability of free space; its value is $4\pi \times 10^{-7}$ H/m, Henrys per meter. The scalars ϵ_r , μ_r and σ correspond to relative permittivity of air, relative permeability of air and iron, and electrical conductivity of the windings. The values used in COMSOL are $\epsilon_r = 1$, for air $\mu_r = 1$, for iron $\mu_r = 1000$, and for copper $\sigma = 6 \times 10^7$ S/m, siemens per meter, respectively. If the model core is changed to a different material, the relative permeability must be changed accordingly.

To model the excitation coils, the above equations are used with the addition of an equation describing \mathbf{J}_e , the current density, in terms of coil windings, where N is the number

of turns, I_{coil} is the current in the coil in amps, A is the cross-sectional area of the coil in square meters, and e_{coil} is the electric field of the coil.

$$\mathbf{J}_e = \frac{NI_{coil}}{A} \mathbf{e}_{coil} \quad (14)$$

The values used for these variables are $N = 30$ turns, $I_{coil} = 1.06$ amps, and $A = 8.23 \times 10^{-7} m^2$.

The equation used to model the pickup coils is the same, except I_{coil} is replaced with I_{cir} to represent the induced current in the pickup coils which is calculated by the software; the other values used are $A = 5.09 \times 10^{-8} m^2$, and $N = 60$ turns.

In order to corroborate theory with reality, the voltage from the pickup coils is extracted from the COMSOL model to be compared with the voltage read directly off the actual pickup coils.

2.2 MATLAB

As one the most popular modeling and simulation softwares available, MATLAB was used as the programming environment for the amplitude and phase measurement algorithm. The equations described in Section 1.1 were programmed into a script which carried out the process on each increment of displacement as the model moved through the coils. MATLAB has a spreadsheet reading capability which extracts data from an Excel file into the MATLAB script. This data corresponds to the multiple measured waveforms which change in amplitude and phase and one reference waveform which remains the same. The cross-correlation as defined above is employed in the MATLAB script in order to compare each measured waveform to the reference waveform. The simplest possible calculation is used in the script to decrease computation time. Results are tabulated in graphs and figures to express the relationship between amplitude and displacement, phase and direction, and for visual representation of any signal.

Since the MATLAB code utilizes data taken in discrete time, the equations described in

Section 1.1 must be reformulated appropriately. The discrete form of Equation 4 becomes

$$R_{xy}(0) = \frac{1}{N_0} \sum_{k=1}^{N_0} x(t)y(t) \quad (15)$$

where N_0 is the total number of data samples and $t = k\Delta t$ or a multiple of sample intervals.[12] The auto-correlations represented in Equations 5 and 6 can now be represented as

$$R_x(0) = \frac{1}{N_0} \sum_{k=1}^{N_0} x(t)^2 \quad (16)$$

$$R_y(0) = \frac{1}{N_0} \sum_{k=1}^{N_0} y(t)^2 \quad (17)$$

Equations 7 through 9 are still valid utilizing the discrete forms of cross-correlation and autocorrelation.

2.3 LABVIEW

LabVIEW is another ubiquitous software used by engineers for its wide range of applications. Two LabVIEW files were constructed for two different purposes: the purpose of the first was for programming a stepper motor to displace the spherical core axially through the model EPS, the second records data taken by an NI 9205 data acquisition module in an Excel spreadsheet for post-processing in MATLAB. In order to have a precise and accurate means for displacing the model core, a stepper motor was programmed using LabVIEW to translate 1 mm at a time. Once the core reaches each 1 mm increment, the corresponding signal is recorded and sent to a text file; each data increment was stored as a 250,000-point discrete periodic signal sampled at 50,000 samples per second (50 kS/s). In this way, a discrete signal for each 1 mm increment over a 10 mm total range can be recorded as digital data to be post-processed with the MATLAB algorithm.

CHAPTER 3

METHODOLOGY

As stated in the previous two chapters, the purpose of this research was to build a one degree of freedom model of the EPS which has digital processing abilities to calculate phase and amplitude change of the measured signal from the pickup coils with respect to the reference signal from the excitation coils. This was accomplished by construction of the physical and electrical model, using COMSOL to model the electromagnetic effects, construction of a core displacement system programmed with LabVIEW, and formulation of a digital processing algorithm in MATLAB. In this chapter, the design of the 1 degree of freedom model system will be described in detail, beginning with the coil form, followed by the conditioning circuit, and the COMSOL model. Then the data collection procedure will be detailed followed by the algorithm design.

3.1 MODEL DESIGN

The first task to be accomplished was to build the physical model including the conditioning circuit and the coils. The dimensions are detailed in the schematic in Figure 9.

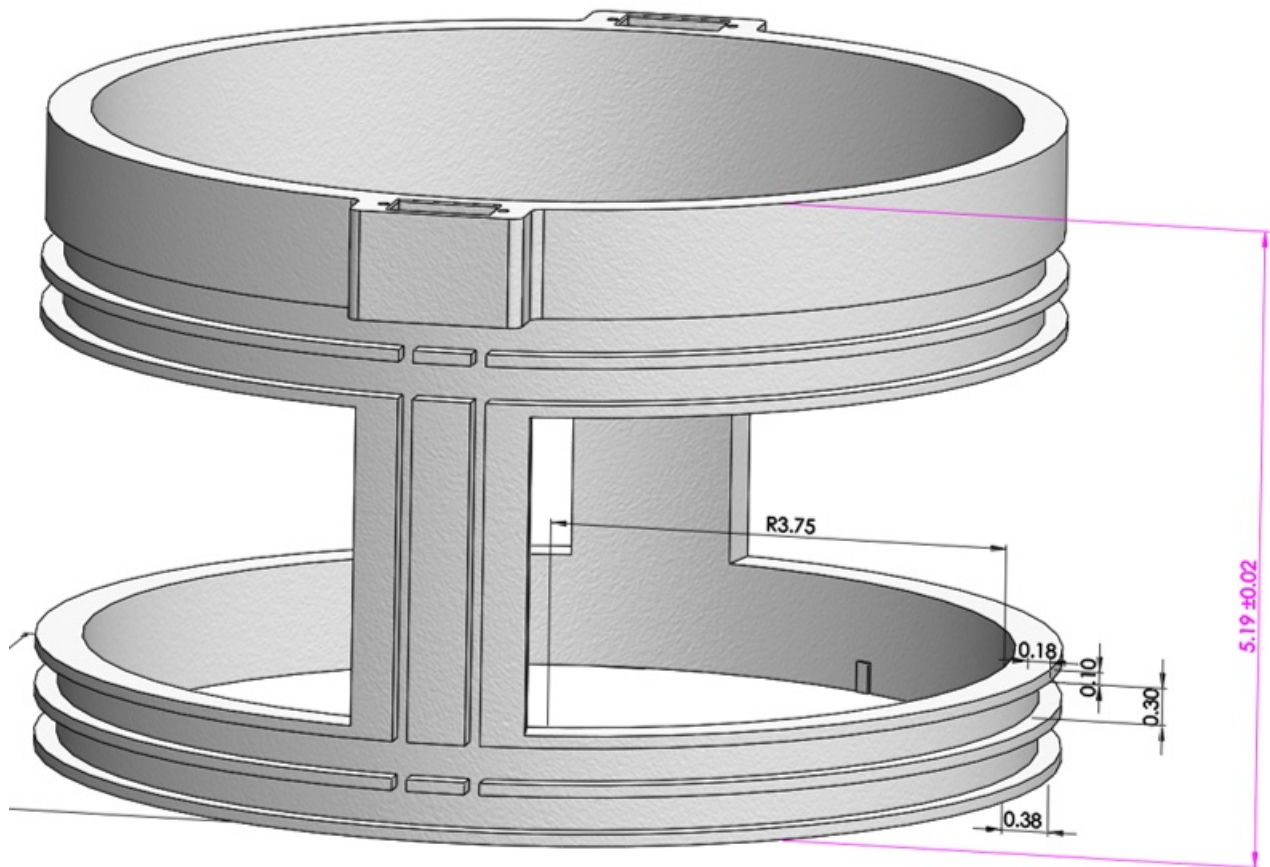


Figure 9: Schematic of 1 Degree of Freedom EPS Coil Form (in)

Since the coil form had been 3D printed previously, 18 gage enameled copper magnet wire was wound 30 times in the inner pair of grooves of the coil form to make the excitation coils and 30 gage enameled copper magnet wire was wound in two sets of 30 turns in the outer grooves of the coil form to make the pickup coils. Both 30 turn outputs are accessible enabling measurement with either 30 or 60 turns of pickup coil; one 30 turn winding is layered on top of the other. Each 30-turn pickup coil was wound in series, so the coil form has a cross-over bridge from one side to the other with a channel for the wire to lay in. There are four wire ends accessible for measurement consisting of two sets of ingoing and outgoing wires. Measurements can be taken for individual coil voltages by referencing one wire end with the corresponding cross-over wire; for LVDT operation measurements must be taken

using the wire ends.

It is important to note that since these coils were hand-wound, they are not exactly symmetrical nor do they have an ideal rectangular cross section. In order to ensure a proper nulling of the pick up coils prior to operation of the model EPS, the wires that make up the pick up coils could be adjusted while the final output signal is wired to an oscilloscope.

Next, the electronics package must be considered. A signal must be produced and conditioned to have the proper parameters for the model EPS. This requires utilizing a commercial off the shelf universal LVDT signal conditioning circuit to power the excitation coils; a system of operational amplifiers and filters adjusts the signal going into the excitation coils and coming out of the pickup coils of the EPS.

3.1.1 CONDITIONING CIRCUIT

The conditioning circuit consists of multiple parts in order to provide a signal to the EPS excitation windings with enough power and also to boost the signal measured from the pickup coils. In order to provide a signal for the coils a source signal, or driving signal, was provided by an Analog Devices EVAL-CN0301 LVDT signal conditioner, a commercial off the shelf LVDT electronics system. The excitation signal is boosted by an OPA548 operational amplifier to provide enough voltage and current for the excitation windings. A power resistor and capacitor (3 Ohms and 4.375 micro Farads respectively) are wired in series with the model EPS to form a resonant circuit. The EPS output from the pickup coils is then sent to another operational amplifier, a TL082cp, which amplifies difference of the two output lines from the EPS. A bandpass filter, the LTC1060, is the final step in the process which eliminates extraneous frequencies. A diagram of the entire system is included below.

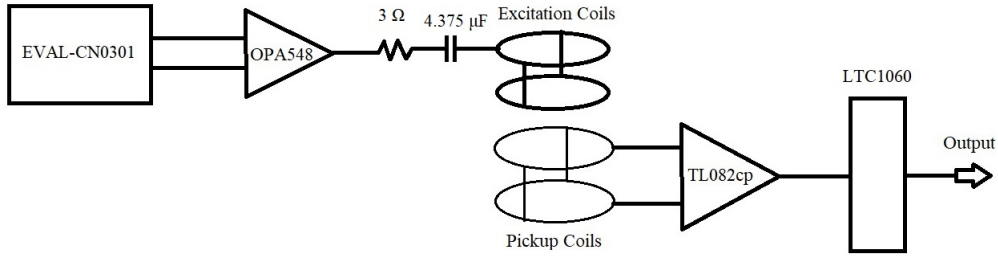


Figure 10: Schematic of Entire Signal Conditioning System

The EVAL-CN0301 conditioning board was chosen to provide the driving signal due to the convenient provision of a suitable excitation carrier signal: 2.36 kHz at 3.84 Vrms and the possibility of analog signal conditioning. This frequency is easily scalable to the 20 kHz full-scale EPS.

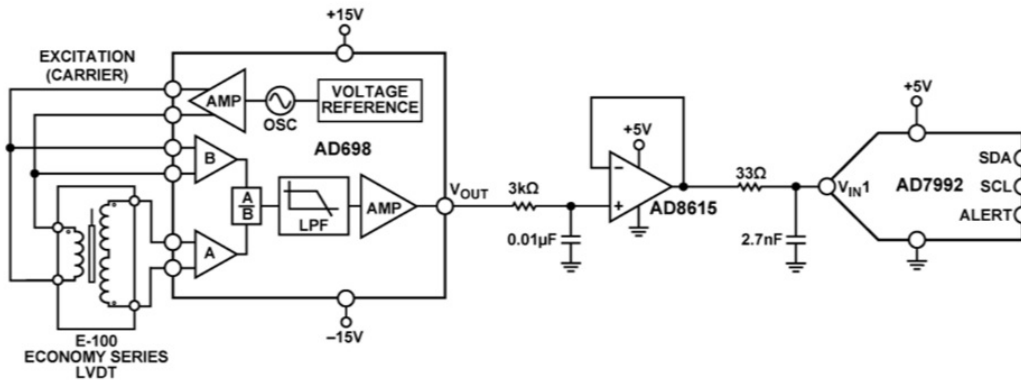


Figure 11: Schematic of EVAL-CN0301 LVDT Conditioning Circuit Board

However, the output of this board is meant for a much smaller LVDT, the E-100 economy series, hence the current must be boosted for the much larger model EPS with an operational amplifier. The full circuit diagram is included in figure 12.

An OPA548 operational amplifier was chosen due to its capability for high-current operation. The power supplied to the OPA548 is ± 15 V, limited to 1.3 amps. There are a few specifications that must be followed in order to ensure that the amplifier operates correctly. The power supply wiring should have a low series impedance and be bypassed in series with low value capacitors; additionally there should be a $0.1\mu\text{F}$ capacitor placed between both

the positive and negative voltage supplies to ground and located as close to the OPA548 as possible. A 14.7 k Ω resistor joining the third pin and the negative power pin limits the current to around 1 amp; this is necessary because the OPA548 has a current range from 0 to 5 amps. While the OPA548 is protected against over-temperature conditions the amplifier has a tendency to heat up quickly, therefore a heat sink was attached to the metallic flange. Since the excitation signal of the EVAL-CN0301 consists of two excitation signals, the amplifier was set up in differential input configuration. The OPA548 produces 1.06 amps peak current at output and with a 1 V peak-to-peak input in the chosen configuration resulting in around 13 Vpp output. The signal feeds an RLC circuit achieving resonance with a 3 Ω shunt resistor and a 4.375 μ F capacitor; the excitation coils have an inductance of 1.597 mH. The equation for sizing the capacitor to achieve resonance is

$$f_r = \frac{1}{2\pi\sqrt{LC}} \quad (18)$$

This increases the voltage to ± 24 V (48 Vpp) which is then sent into the excitation coils.

The net voltage output of the pickup coils is of a much smaller magnitude compared to the excitation voltage; the net output from the pickup coils is a few millivolts. When the model core is not centered in the coils the magnetic flux coupling results in a small voltage asymmetry between the two pickup coils. The difference between the voltage in each pickup coil results in the net voltage output sent back to the signal conditioning circuit. Theoretically the current through the pickup coils would be zero if the model EPS acted as an ideal transformer, the actual current through the model's pickup coils is 0.37 milliamps. The signal output from the pickup coils is read differentially. For the 30 turn configuration the ingoing end and outgoing end of the same wire are referenced to each other; for 60 turns the ingoing wire of the bottom pickup coil layer is referenced to the outgoing wire of the top pickup coil layer. The signal output from the pickup coils is 10-15 mV for the null condition and around 30 mV when the iron sphere core is at the extremes of the range of

displacements.

One channel of a TL082cp in basic differential amplifier configuration was utilized to capture the net signal; the gain was set to ten to allow for the differential reading to be an order of magnitude larger and hence more easily observed. However, before the signal can be ready for processing it must be sent to the filter due to noise contamination.

A bandpass filter is appropriate for this application because the exact frequency of the EPS is known beforehand, therefore all other frequencies can be rejected. The bandpass filter chosen was a switched-capacitor LTC1060 building block filter utilizing the Butterworth bandpass function. This filter has two 2nd order sections requiring four resistors each for operation. The nominal frequency of the filter was chosen to be $f_{0BP} = 2.36$ kHz, the frequency of the carrier signal. A bandwidth of 200 Hz was chosen requiring $\frac{f_0}{BW} = 10$; this relation indicates on Table 1 of the Reference 13 that the passband frequencies are required to be $f_{01} = 2.28$ kHz and $f_{02} = 2.44$ kHz.[13] The filter requires a clock frequency, f_c which was chosen to be 100 kHz. It is important to note that this clock frequency should be 2 Vpp with a dc offset of 1 V; therefore the minimum value of the clock signal is 0 V and the maximum is 2 V. Again using Table 1 of Reference 13 the Q, a dimensionless value which relates to the bandwidth, is found to be $Q_1 = Q_2 = 14.2$. Given that $K = 2.03$ from the same table, the individual bandpass gains H_{0BP} can be found since K is the product of the individual bandpass gains. The first bandpass gain was chosen as $H_{0BP1} = 1$ which set the second gain as $H_{0BP2} = 2.03$. The overall gain of the filter is therefore 2.03.

Using the four frequencies, the Q value, and the H values determined above, the resistor values can be found using a few mathematical relations. The datasheet of the LTC1060 provides the 2nd order functions of each mode of operation; the equation given is

$$f_0 = \frac{f_c}{50} \sqrt{\frac{R_2}{R_4}} \quad (19)$$

This provides the first relationship between resistor 2 and 4. This same equation can be

used for the passband frequencies as well, providing two equations with two unknowns.

$$f_{01} = \frac{f_c}{50} \sqrt{\frac{R2}{R4}} \quad (20)$$

$$f_{02} = \frac{f_c}{50} \sqrt{\frac{R2}{R4}} \quad (21)$$

Substituting in the known frequencies, these relationships simplify to

$$\frac{R2}{R4} = 1.3 \quad (22)$$

$$\frac{R2}{R4} = 1.5 \quad (23)$$

The mode 3 configuration also provides relationships to characterize Q and H providing bandpass: [13]

$$Q = \frac{R3}{R2} \sqrt{\frac{R2}{R4}} \quad (24)$$

$$H_{0BP} = -\frac{R3}{R1} \quad (25)$$

By substituting in the two H values given above, equation 18 becomes

$$-1 = \frac{R3}{R1} \quad (26)$$

and

$$-2.03 = \frac{R3}{R1} \quad (27)$$

Since $Q_1 = Q_2 = 14.2$ and substituting in this value in addition to the values found in equations 15 and 16, equation 17 simplifies to

$$12.5 = \frac{R3}{R2} \quad (28)$$

and

$$11.6 = \frac{R3}{R2} \quad (29)$$

A bandpass filter operates between two cutoff frequencies as is shown in Figure 12; the two sections of resistors specify the lower and higher cutoff frequencies. The first system of equations requires grouping together equations 15, 19, and 22 which all correspond to the lower cutoff frequency. The second system of equations consists of equations 16, 20, and 22 which correspond to the higher cutoff frequency. Reference 13 recommends to set R2 of the first system and R4 of the second system as $10k\Omega$. [13] This results in the following resistor values:

First section	Second section
$R11 = 124k$	$R12 = 84.5k$
$R21 = 10k$	$R22 = 15k$
$R31 = 124k$	$R32 = 174k$
$R41 = 7.68k$	$R42 = 10k$

Table 1: Table of Resistor Values

The LTC1060 has some design requirements for the power supplies; ± 5 V is supplied to power the filter while Pins 7 and 8 and 13 and 14, which are the analog and digital positive and negative supply pins respectively, are tied together and bypassed by a $0.1\mu\text{F}$ disc ceramic capacitor; the capacitors reduce noise errors in the power supply lines. When operational, the filter's bandwidth was confirmed to be 3 dB with a low cutoff frequency around 2.2 kHz. This allows only signals with frequencies within the cutoff frequency range to pass through the filter, rejecting most noise of the electronic system.

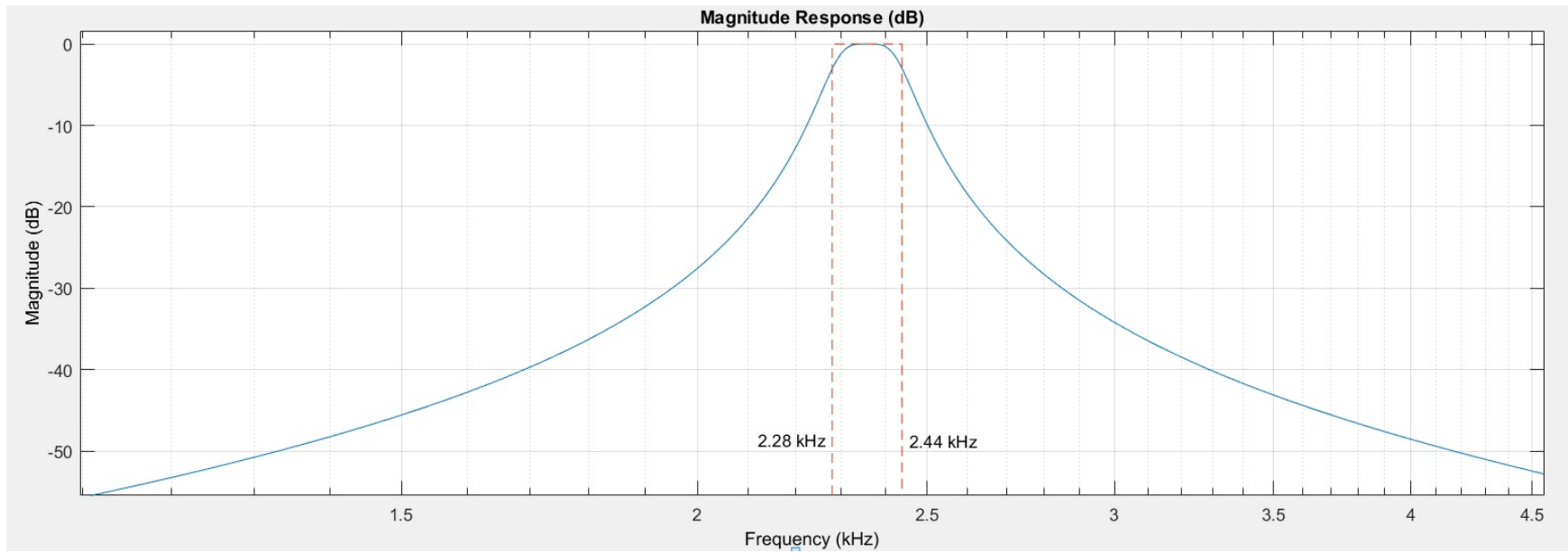
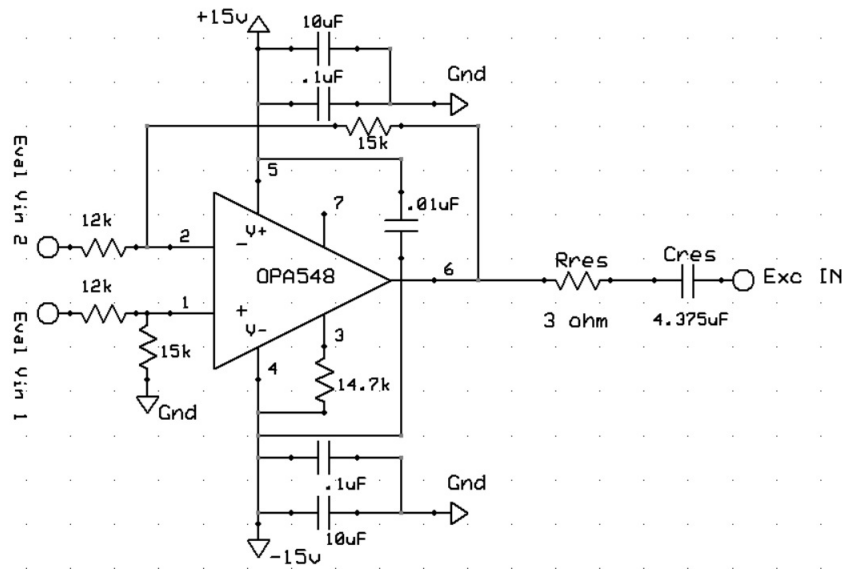
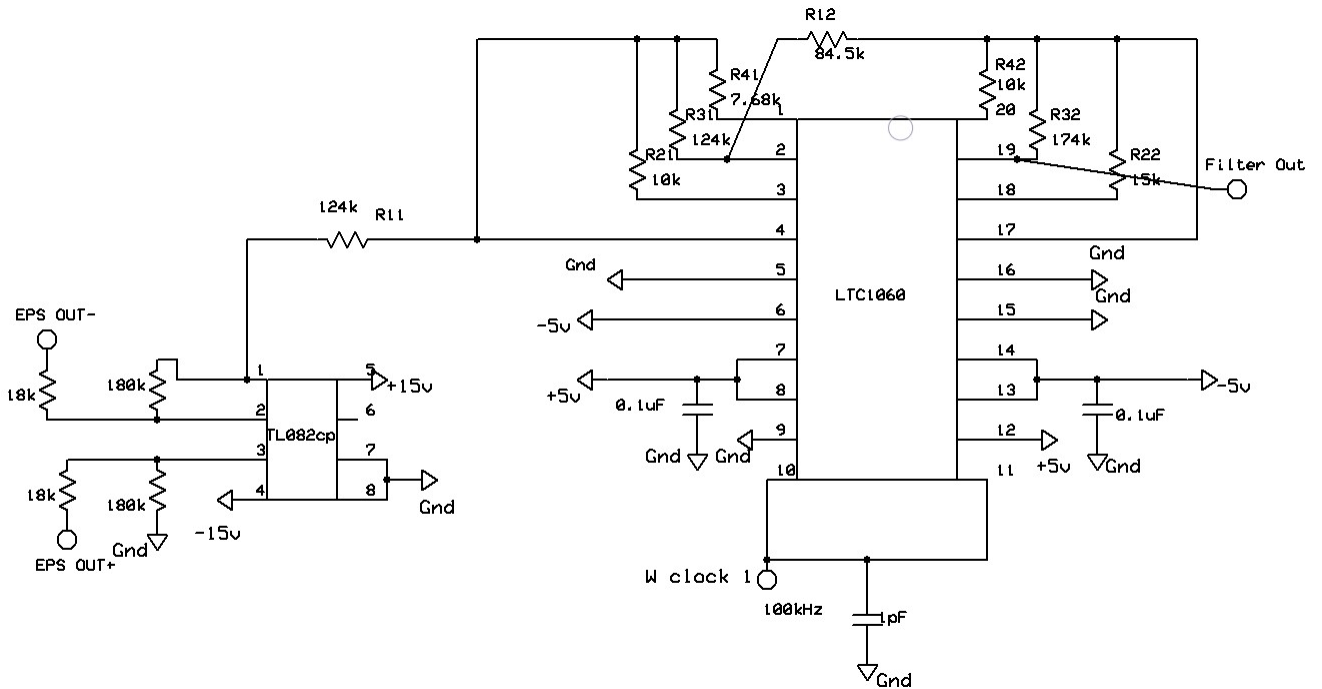


Figure 12: Frequency Response of LTC1060

Figure 12 below is a schematic of the entire conditioning circuit.



((a)) Pre-Excitation Amplifier



((b)) Post-Pickup Amplifier and Filter

Figure 12: Schematic of Conditioning Circuit

Once the design was proven functional in a breadboard, a printed circuit board was designed using the open sourced software ExpressPCB. There were multiple factors to consider

in the design: inputs and outputs must be kept as far from the power supply lines as possible, most capacitors must be placed as close as possible to the chips, and all power supply lines must be 0.05 inches in width to accommodate the voltage and current required. Figure 13 is the conditioning circuit board designed.

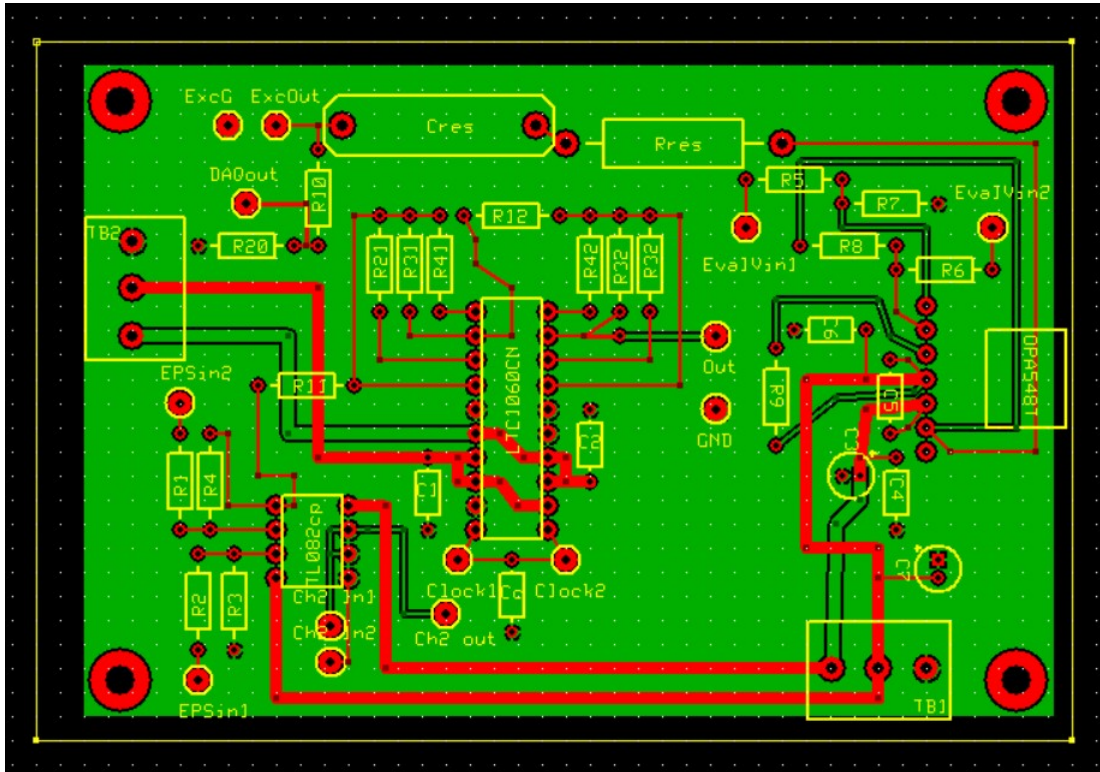


Figure 13: EPS Conditioning Circuit PCB Design

3.1.2 COMSOL MODEL

Using COMSOL multiphysics, an idealized model was developed to compare to the physical model. By calculating the voltages induced in the pickup coils using the equations above in 2.1 on a finite element model, COMSOL provides a theoretical comparison to the physical results. As described earlier, both the magnetic field and electric circuit physics are employed in this model to provide as much information from the physical model as possible. The excitation and pickup coil geometry is input by the user as parameters of the system and the materials of the coils and the core are specified as copper and iron respectively. Each

coil set has different settings due to their different functions; since the excitation coils have a set current and voltage these are provided to the excitation coil model. However the pickup coils receive their voltage and current via coupling from the excitation coils; these values must be calculated by the software. Figure 14 is a cross section of the 3D cylindrical rendering made by COMSOL; the excitation coils are the larger, interior rectangular cross sections and the pickup coils are the smaller, outer rectangular cross sections, both are symmetrically arranged above and below the cylindrical model core.

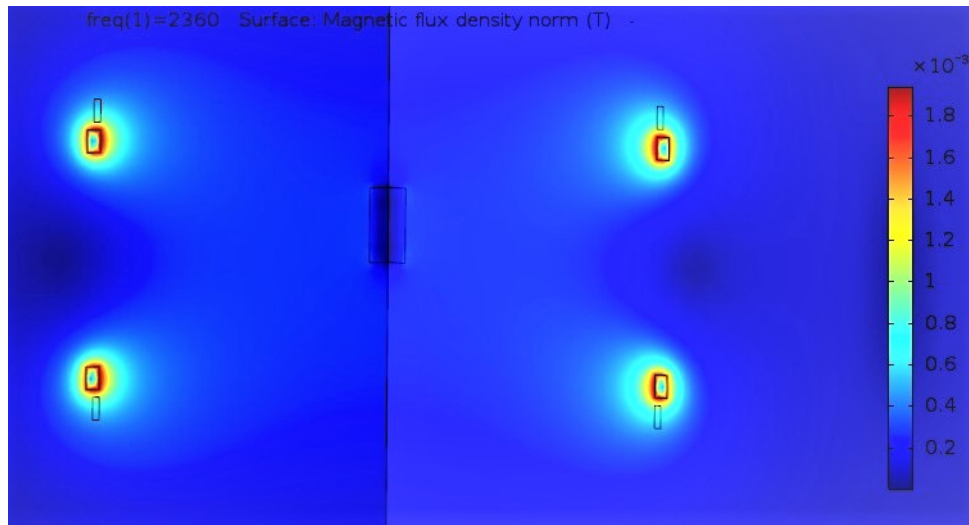


Figure 14: Surface Magnetic Flux Density Norm (T) of Cross-section of 1D EPS in COMSOL

The results of model evaluation provide the voltage in each pick up coil considering one coil to be the 60 turns on one side of the axial origin and the other coil to be the 60 turns on the other side - in Figure 14 the coils above and below the model core. By subtracting the voltages in each coil, the total output of the pickup coils can be ascertained. When the iron core is modeled at the exact center of the coils, the voltage on each coil is the same resulting in zero output signal. When the iron core is at maximum displacement corresponding to the distance of each pickup coil, the voltage on each coil is a complex number which is broken down into magnitude and phase. The voltage magnitude is found with the equation

$$c = \sqrt{a^2 + b^2} \quad (30)$$

where a is the real component and b is the imaginary component of the complex number output in COMSOL. The phase angle is found by computing the angle between the real and imaginary components using

$$\theta = \arctan \frac{b}{a} \quad (31)$$

In order to confirm that phase shifts exhibited by the output of the 1D EPS are material dependent, multiple materials were tested as the model core. The main model core utilized was a 1.25 inch steel sphere; this was the core used for all data measurements. The two other materials used for comparison were an aluminum cylinder and a copper disc.

3.2 DATA COLLECTION PROCEDURE

A data acquisition module was used to convert the analog output signals from the excitation and pickup coils into corresponding digital signals: the reference and measured signals. An NI 9205 A/D connector with 250 kS/s (kilosamples per second) maximum sample rate and 16-Bit resolution was connected to the analog outputs of the conditioning circuit. Since the NI 9205 has a maximum range of 12 V peak to peak, the 48 V peak to peak reference or excitation signal was scaled to 8 V_{pp} for data acquisition using a voltage divider; the resistor values used were 15 k Ω and 75 k Ω . The measured signal ranges from 0 V to 300 mV_{pp} during displacement. To avoid interaction between the A/D and the filter feedback paths, a separate TL082cp amplifier was added to the filter output, operated in single-supply mode with a gain of 1. The scaled reference signal and the conditioned measured signal are sent as ground-referenced differential pairs to two channels of the NI 9205. A LabView code samples each signal at a sample rate of 50 kHz, more than twenty times the Nyquist frequency. During processing it is preferable to ensure an integer number of waves in each record so as to avoid windowing distortion. This kind of distortion occurs when an infinite duration signal is truncated into a finite duration signal; the best way to avoid distortion is to ensure that the beginning of the finite signal aligns well with its end. Record lengths of 250,000 samples were recorded for evaluating the performance matrices of the digitization algorithm. The

digital voltage data is sent to a text file which documents the signal each time the program is run.

Procedure for data collection is as follows: the core is manually centered within the model coils to produce as close to null voltage as possible, then using the core displacement system the core is displaced 5 mm in either direction. In this case the core was displaced to the left or thick side of the EPS, or away from the stepper motor. This provides the first data point; the reference signal and the pickup signal produced by the core at -5 mm are recorded in Excel using the LabView file. The core displacement system then moves the core 1 mm forwards, taking another data point and recording each signal; this is repeated until 11 data points are collected. The information provided by each data point includes the constant reference signal of 8 V and the measured pickup signal. As the core moves from maximum displacement to zero and back to maximum, a corresponding change in amplitude is evident in the pickup signal. The phase change of the measured signal corresponds to the orientation of the core with respect to the origin of the axis of the model EPS. For raw signal data graphs, see Appendix E.

3.3 ALGORITHM DESIGN

To process the digital data collected, it was necessary to design a digital algorithm using the principles of cross-correlation as detailed in Section 2.2. The inputs of the algorithm are the measured and reference signals collected from the processed pickup signal and the excitation signal respectively. A simulation algorithm was designed in parallel with the data processing algorithm to allow for a control version utilizing ideal signals. The simulation and measured data algorithms are compared to test functionality and to experiment with different processing conditions and techniques.

3.3.1 SIMULATION ALGORITHM

Both the simulation and measured data algorithms were programmed in MATLAB, in most respects identical to each other. The main difference is that the simulation algorithm

uses simulated signals, idealized sine and cosine signals produced by MATLAB functions, instead of digital data. A sine function with added noise represents the measured signal while another sine wave represents the reference signal. The simulated measured signal is given a changing amplitude to represent model displacement: a range from -1 to 0 to 1 V represents the phase shift at the origin as a change in sign. This scale change occurs over 41 simulated data points which represent incremental displacement of the model core. A specified phase difference due to core characteristics (see Chapter 3.1.2) of 50 degrees is given in the script preamble. Two main purposes of this script are to verify that the range of the processed phase difference equals the given phase difference and that the processed amplitudes of each simulated signal equals the corresponding specified amplitudes of the simulated sine functions. The full MATLAB script is given in Appendix D.1.

The theoretical principles used to process the amplitude and phase information are specified in Section 2.2 as auto- and cross-correlation. To extract amplitude information from both simulated signals at each data point, discretized equations 16 and 17 are utilized. To process the cross correlation of the simulated signals equation 15 must be utilized since it is the discrete form. Equation 9 is then used to find the phase difference between the two signals at each data point. These operations are performed for each data point representing an incremental displacement of the model through the coils. The amplitude ratio and phase change are then plotted for analysis over the scale range. Graphs of the simulated signals are included in Appendix D-2. In Figure 15, the four graphs represent the processed data: R represents the cross-correlated data of both signals, X represents the auto-correlation of the reference signal data, and Y represents the auto-correlation of the measured signal data. The cross-correlation is at a maximum where the signals match, at the extremes of the displacement points. The auto-correlation of the reference signal, X, is constant because the signal does not change over time; however the auto-correlation of the measured signal, Y, does change incrementally as the core displacement changes. The maximum amplitude occurs at the extreme displacement points where the original signal has the greatest ampli-

tude. The amplitudes in the second graph are calculated using equations 7 and 8 once the autocorrelations have been found.

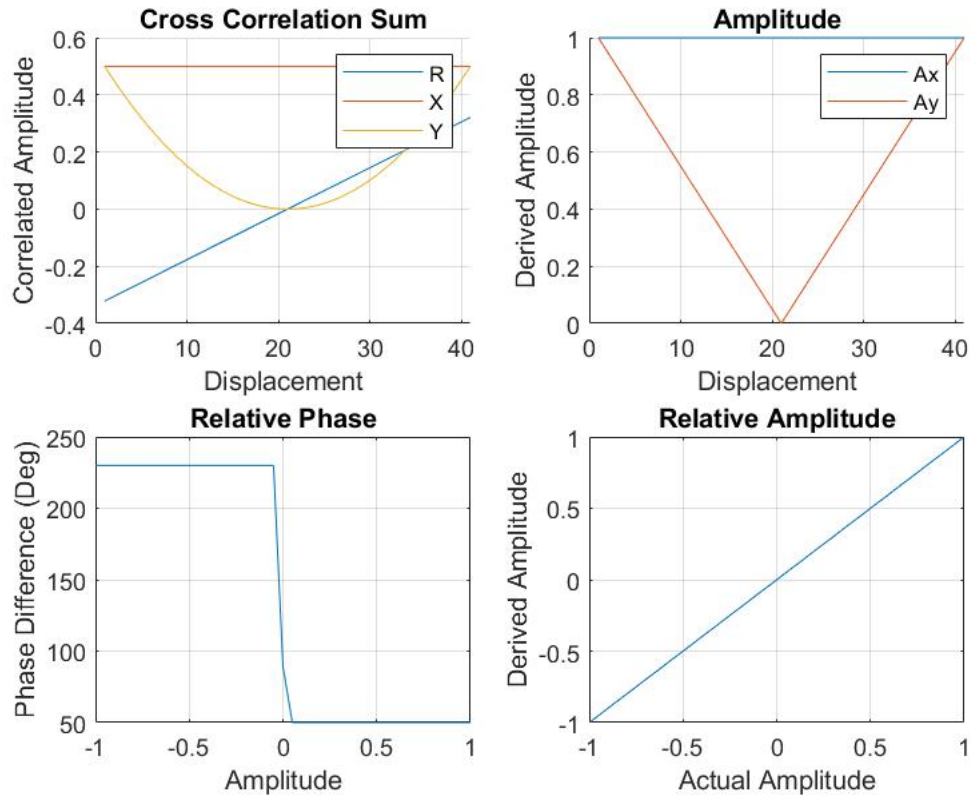


Figure 15: Graphs of Simulated Amplitude Ratio and Phase Difference

A zero-point has been omitted from this processed data in the relative phase graph due to the indeterminate phase calculation at that point. The relative phase calculation in equation 9 includes the cross-correlation value in the numerator of the inverse cosine. Since the middle point signal has an amplitude of zero, the cross correlation at that point is also zero. However, this only occurs when there are perfect, noiseless signals with no error. Since this will not be the case with actual signals, this phenomenon will not occur when processing real data.

The relative phase for the first twenty data points is -130 degrees (230 degrees) and shifts at the middle point to +50 degrees for the last twenty data points. In this run of the simulation algorithm, 50 degrees was the specified phase shift so the results match

expectations. The mechanism of the phase calculation first subtracts 180 degrees from the specified phase shift of 50 degrees resulting in a 130 degree phase lag.

These results are accurate since they represent the relationships between the signals correctly, however it has been noted that the range of the phase angle now corresponds to the opposite direction of motion as the analog system. This difference is easily taken into account for future experiments and calculations.

Once the simulation algorithm was operational it was necessary to add noise, change the sample rate to test for discretization error, change the sample length, and to change the number of bits to investigate quantization error; all these processes are required to test the algorithm for robustness when converting a real-time continuous signal into a discrete, digitized signal at a finite rate. Quantization error occurs in analog to digital conversion due to the approximation of a continuous signal into discrete values utilizing quantization levels; the voltage is changed from continuous to discrete levels. The more bits utilized by an analog-to-digital converter results in more levels and less quantization error. Discretization error also occurs due to analog to digital conversion; because of sampling the continuous signal is converted into a discrete-time signal. In this case, the continuous time signal is replaced with a sequence of discrete samples. In order to represent the signals properly, the Nyquist theorem suggests that the sample rate must be above two times the frequency of the continuous signal, however the standard practice is to use ten times the frequency of the carrier signal. It is advantageous to this thesis that the exact frequency of the system is known beforehand; this is also the case with the full-scale EPS and MSBS which operate at 20 kHz. The fidelity of the sampled signal increases as the sample rate increases, therefore it is important to sample at as high a sample rate as possible. By decreasing the sample rate, fewer data points are taken from the original signal; this results in the possibility of error upon reconstruction of the signal.[5]

3.3.1.1 NOISE ANALYSIS

To test the simulated algorithm for sensitivity to noise, two different sources of noise

were simulated: random noise and current ripple or discrete frequency noise. Current ripple noise is another source of unsynchronized noise produced by an AC power source. In the real MSBS and EPS system, the power supply ripple is 51 kHz. Both current ripple and random noise was added onto the simulated signal and the relative amplitude values for each data point were calculated. Each noise error case was run 10 times.

The algorithm responds relatively well to random noise; the amplitude of noise signals was chosen to be 10% and 20% of the maximum displacement of the pickup signal, which is 1 V. Therefore, the 10% noise signal has a peak amplitude of 0.1 V, the 20% noise signal has an amplitude of 0.2 V. The mean of the relative amplitude for each error case's 10 runs is shown in the figure 16. When random noise reaches 0.1 V, the relative amplitude exhibits a slight nonlinearity around the origin. When increased to 0.2 V, the linearity of the relationship breaks down around the origin. This phenomena is shown in Figure 16.

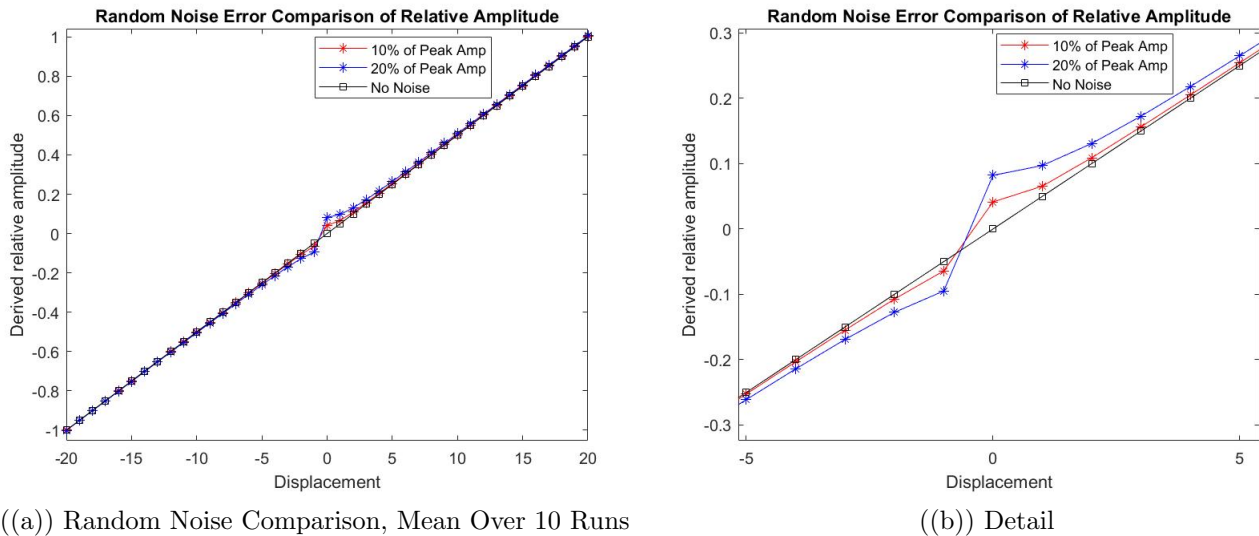


Figure 16: Comparison of Random Noise: 10% & 20% of Maximum Pickup Amplitude

The standard deviation of each noise comparison was found between the 10 runs with each added noise case (10% and 20%); this was performed on a point-by-point basis and is shown in Figure 17.

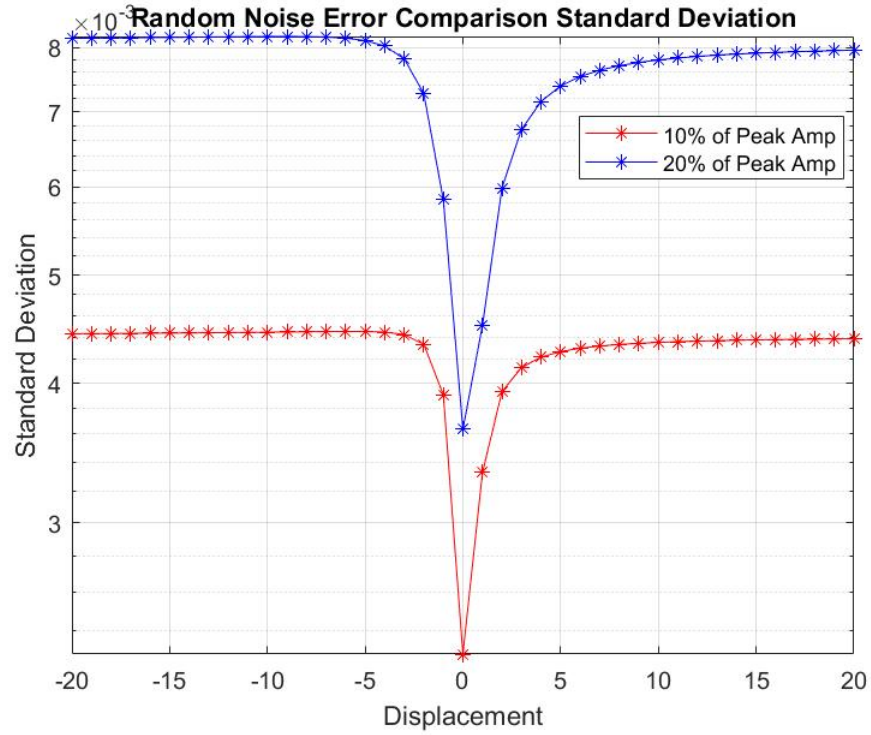


Figure 17: Random Noise Level Standard Deviation (V)

The deviation was calculated between the means of each random noise case and the noiseless run. The greatest error occurs at the center point as expected.

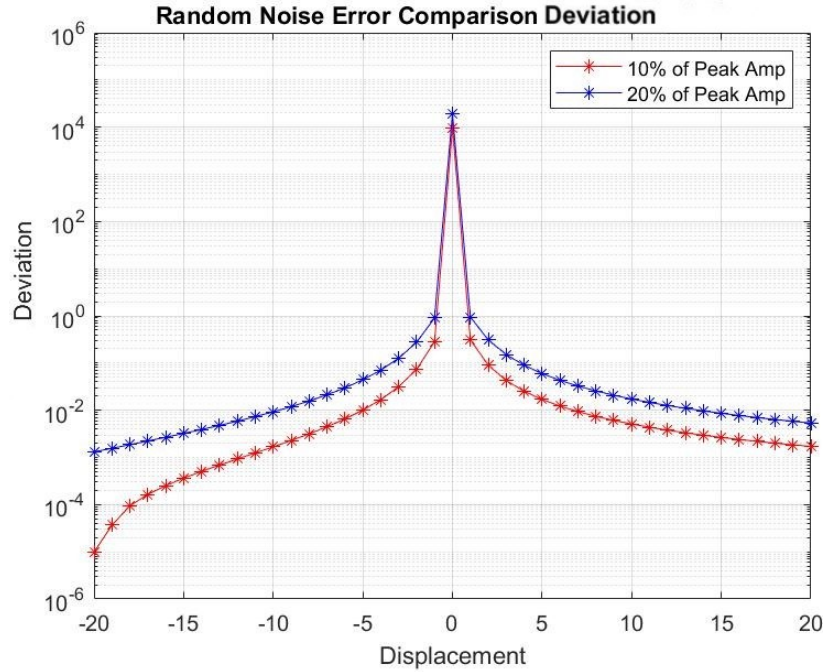


Figure 18: Random Noise Level Deviation

The other source of noise to consider is the current ripple noise; it has a specified frequency ratio of 2.55, the ratio of the real MSBS EPS system's current ripple frequency to operating frequency ($\frac{51kHz}{20kHz} = 2.55$). The algorithm breaks down when ripple has a peak amplitude of 0.1 V, therefore the two cases are 5% and 10% of maximum peak amplitude; the current ripple has a greater effect on the algorithm than random noise. The breakdown of the relative amplitude linearity is illustrated by the detail graph in Figure 19.

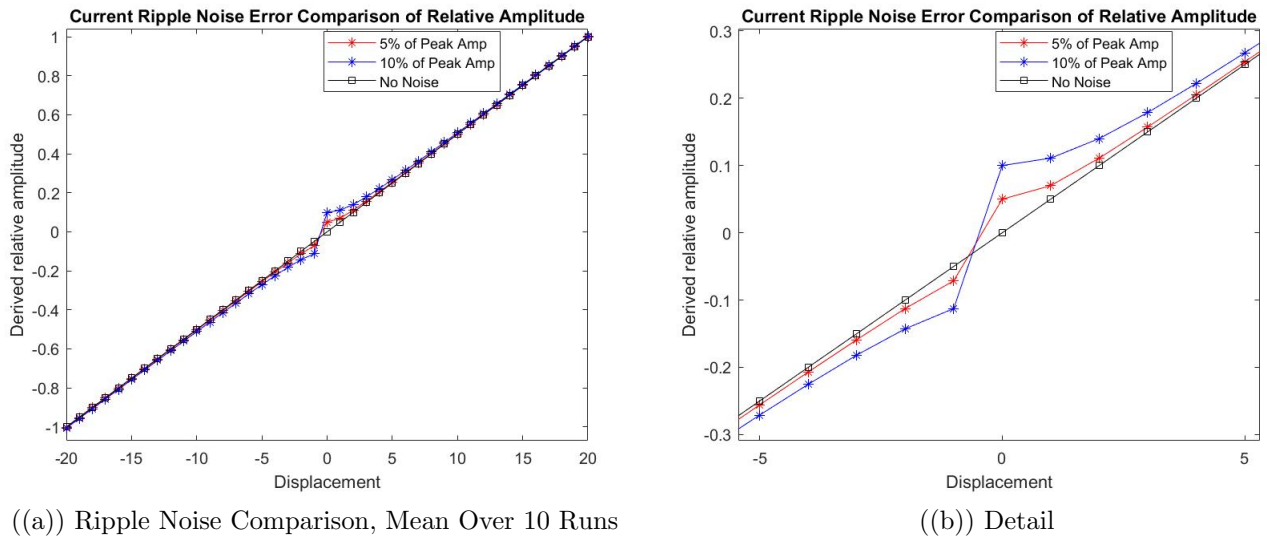


Figure 19: Comparison of Ripple Noise: 0.05 V peak and 0.1 V Peak Amplitude

The standard deviation of the 10 runs was found; the deviation of each run compared with the others is a very low order of magnitude. Some of the computed points are missing because they achieved exactly zero deviation. This signifies very little variance between each run when current ripple noise is present.

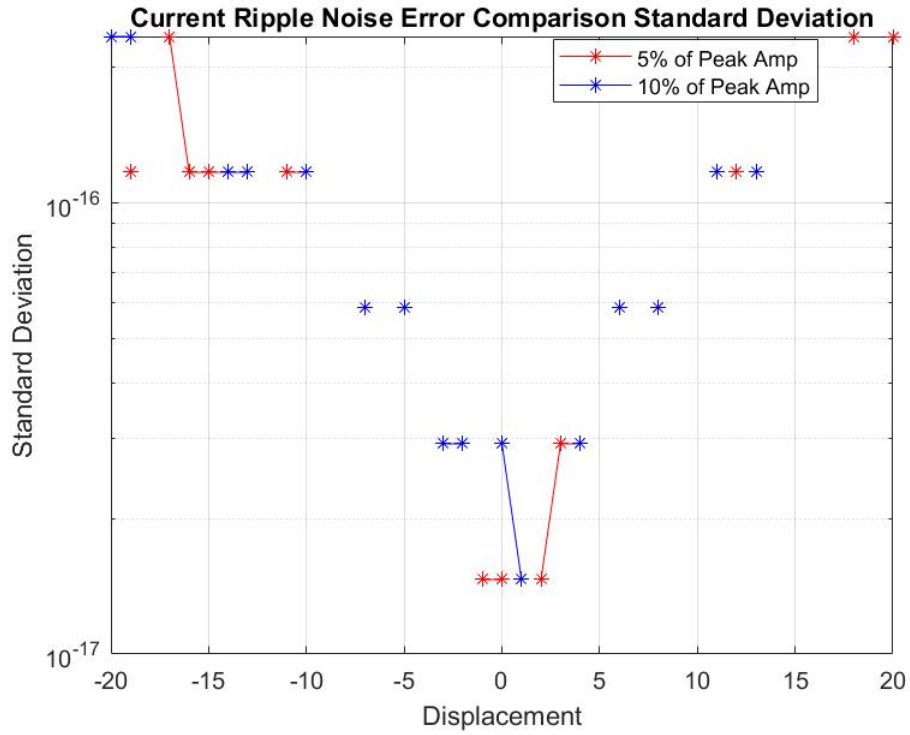


Figure 20: Current Ripple Noise Level Standard Deviation (V)

The deviation displays a similar trend to the random noise cases. Figure 21 illustrates how 0.1 V peak of current ripple noise results in the same order of magnitude of error as 0.2 V of random noise.

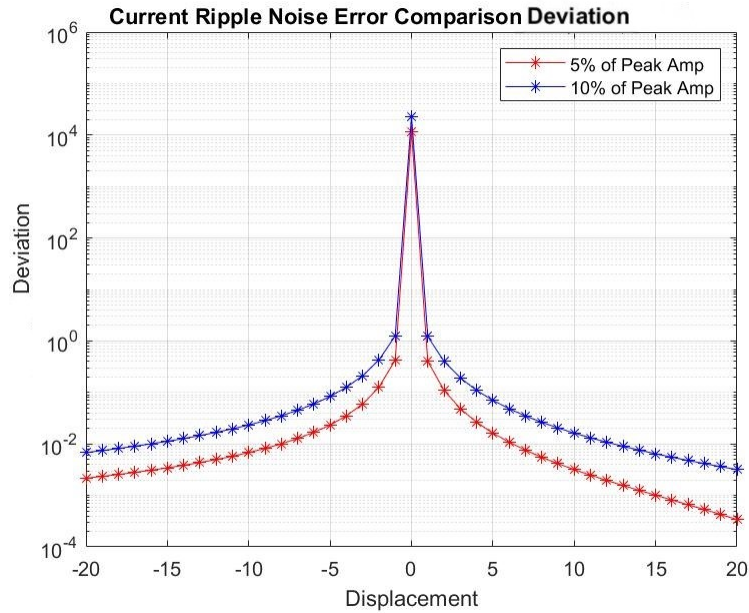


Figure 21: Current Ripple Noise Level Deviation

The last step in noise analysis is to mix the two sources of unsynchronized noise. This was accomplished by adding 5% (0.05 V) current ripple noise and 5% random noise to the pickup signal; the next case adds 5% current ripple noise and 10% (0.1 V) random noise. The third case adds 10% current ripple noise and 5% random noise, and the final case adds 10% current ripple noise and 10% random noise. All cases were run 10 times and the mean relative amplitude was calculated for each case.

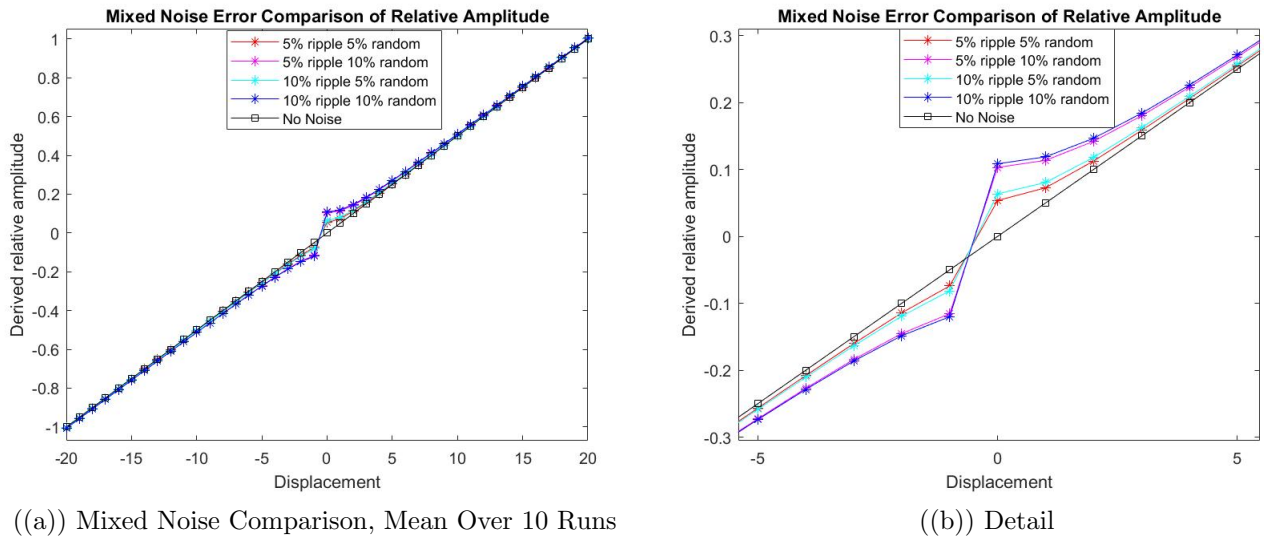


Figure 22: Noise Comparison: Random & Current Ripple Mixed: 0.05 V Peak & 0.1 V Peak

Again, the standard deviation of the 10 runs was found and are compared to each other in Figure 23.

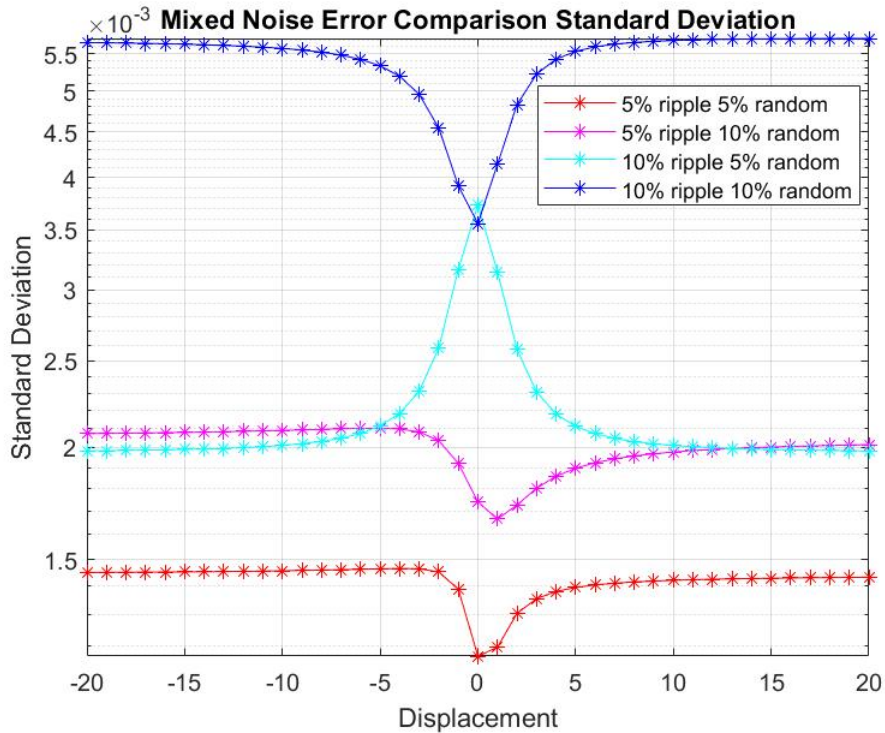


Figure 23: Mixed Noise Level Standard Deviation (V)

The deviation of each case from the noiseless case was calculated as well. It is evident that the random noise has a greater effect on the algorithm when both noise sources are added.

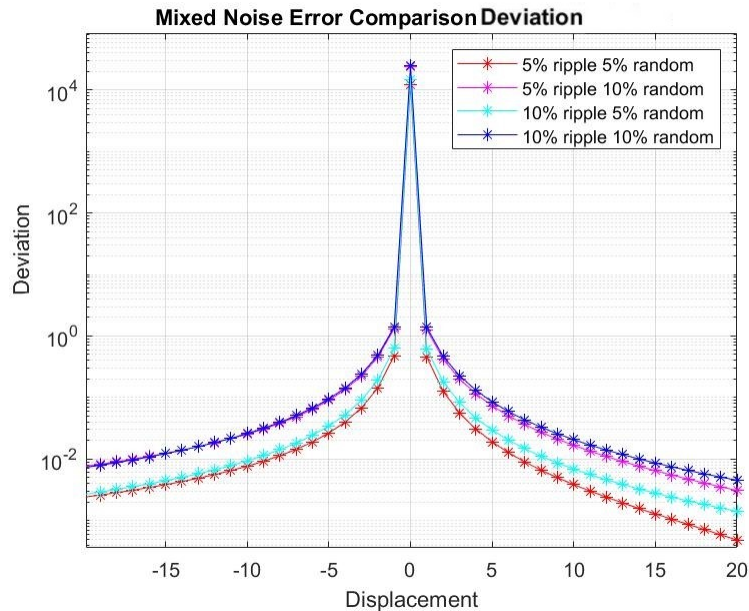


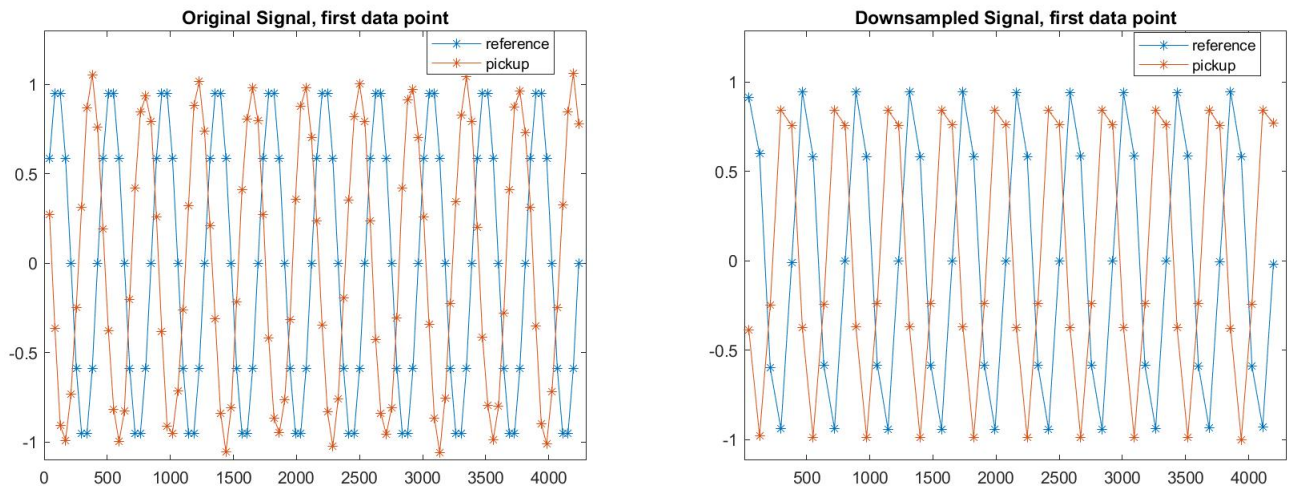
Figure 24: Mixed Noise Level Deviation

The ripple noise is no longer the dominant error factor when both noise types are added to the pickup signal; this is encouraging since the peak amplitude of the ripple noise can be larger than the pickup signal.

3.3.1.2 SAMPLE RATE ANALYSIS

To test for discretization error, the sample rate was reduced from its original value. The simulation algorithm's baseline sample rate is defined by multiplying the number of complete cycles by ten; since the reference number of cycles is ten this results in a sample number of 100 points per ten cycles. Sample sequences at half and one quarter the original rates were developed by downsampling the original data sequence. When testing the algorithm for discretization error it withstood sample rate decreases down to one fourth of the original sample rate, or 25 points per ten cycles.

When the sample rate was reduced by half for a total of 50 points per ten cycles, there was hardly a difference in the amplitude or phase calculations. The sinusoidal signal contains half the points of the original and retains its frequency, but the fidelity of the signal is greatly reduced as is evident in Figure 25.

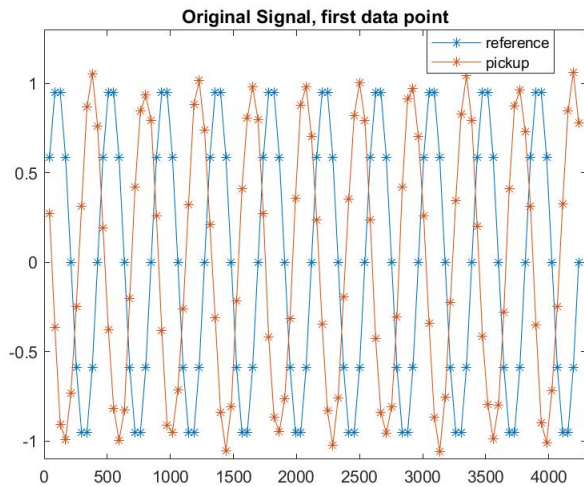


((a)) 100 samples per 10 cycles

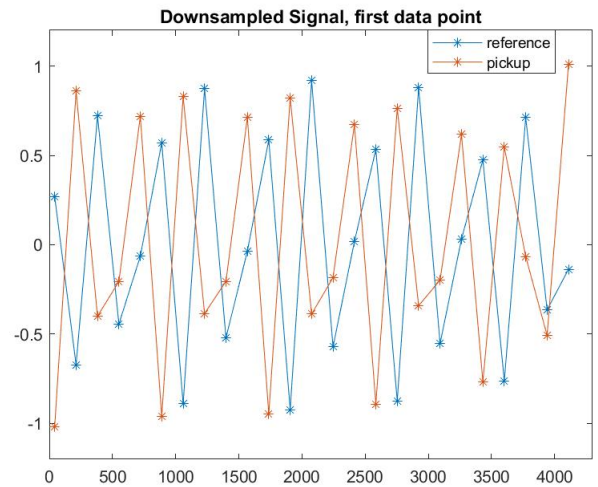
((b)) 50 samples per 10 cycles

Figure 25: Graphs of Signals: Downsampling at Half Sample Rate Comparison

When attempting to find the breaking point of the algorithm, it was found that a sample rate of one fifth of the original would accomplish this due to the violation of the Nyquist theorem. At one fourth of the sample rate, the algorithm can still function as intended. For comparison, Figure 26 shows the original signals versus the downsampled signals and Figure 27 shows the processed amplitudes and their deviations down to one fourth the sample rate.

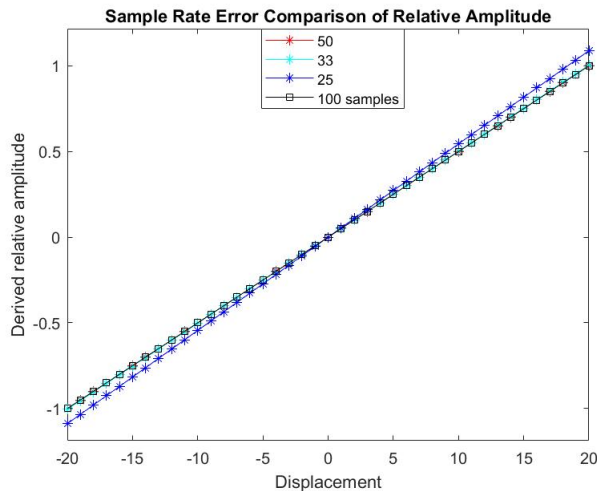


((a)) 100 samples per 10 cycles

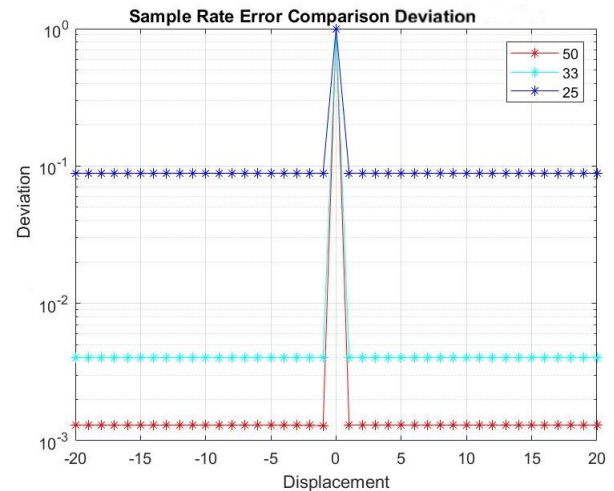


((b)) 25 samples per 10 cycles

Figure 26: Graphs of Signals: Downsampling at One Fourth Sample Rate Comparison



((a)) Sample Rate Comparison (10 cycles)



((b)) Deviation

Figure 27: Comparison of Different Sample Rates

Instead of exhibiting nonlinearity around the center point, a reduction of sample rate past 25 samples per 10 cycles alters the slope of the relative amplitude graph past acceptable margins. This phenomenon begins to occur at 25 samples per 10 cycles and can be seen in the first graph of Figure 27. However, at 25 samples per 10 cycles the percent deviation remains just below the 10^{-1} order of magnitude threshold. It is encouraging that the simulated data

algorithm can withstand a one-fourth reduction of sample rate, however current and future technology will enable the algorithm to run at high sample rates.

3.3.1.3 RECORD LENGTH ANALYSIS

Changing the record length had no effect even down to a minimum of one cycle. As is evident in Figure 28, all processing occurs just as well with one period of signal as it did with ten periods of signal.

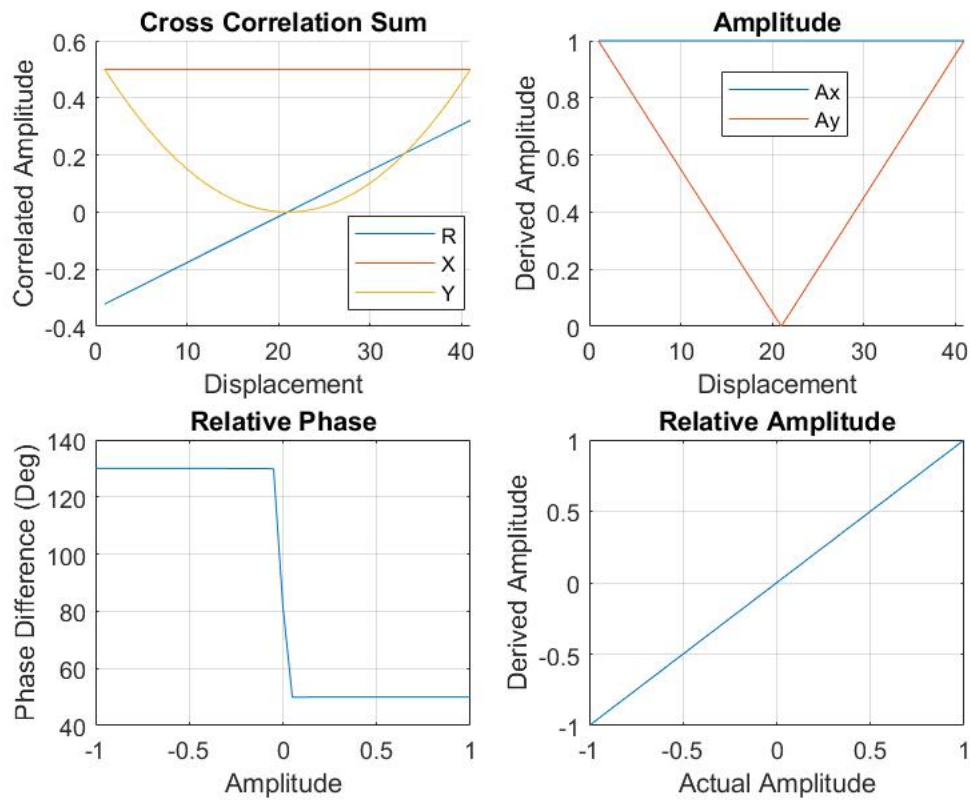


Figure 28: Graphs of Amplitude Ratio and Phase Difference: Record Length One Cycle

This performance cannot be expected from a digitized and hence discretized data set, or once noise is added.

3.3.1.4 QUANTIZATION ANALYSIS

To test the algorithm for quantization error, the number of bits of simulated data were changed until the minimum number of required bits was found. Quantization error was

ascertained by varying the bits inherent in the data acquisition; by varying the number of bits the signal is divided into different numbers of levels according to the equation

$$M = 2^N \quad (32)$$

where M is the number of levels and N is the number of bits. The more levels are included, a more complete picture of the signal can be reproduced digitally. For example, a 2-bit length will result in only four levels of discrete values for the signal. For illustrative purposes, a 3-bit quantized signal with eight levels is shown below.

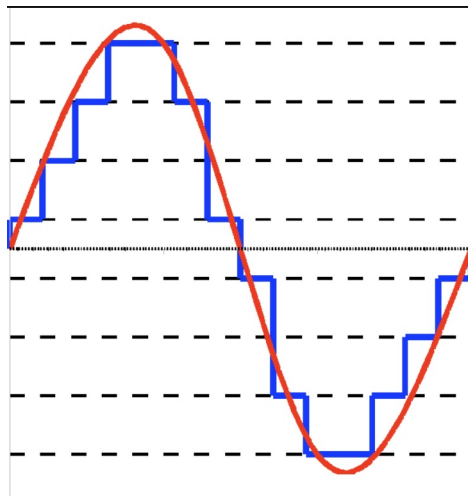


Figure 29: 3-bit Quantized Signal with 8 Levels

In order to find the minimum bits necessary the simulation was run with 8, 10, 12, and 16 bits to represent different levels of quantization error with 16 bits representing nominally zero error. The results are shown in Figure 30; a detailed view is also included.

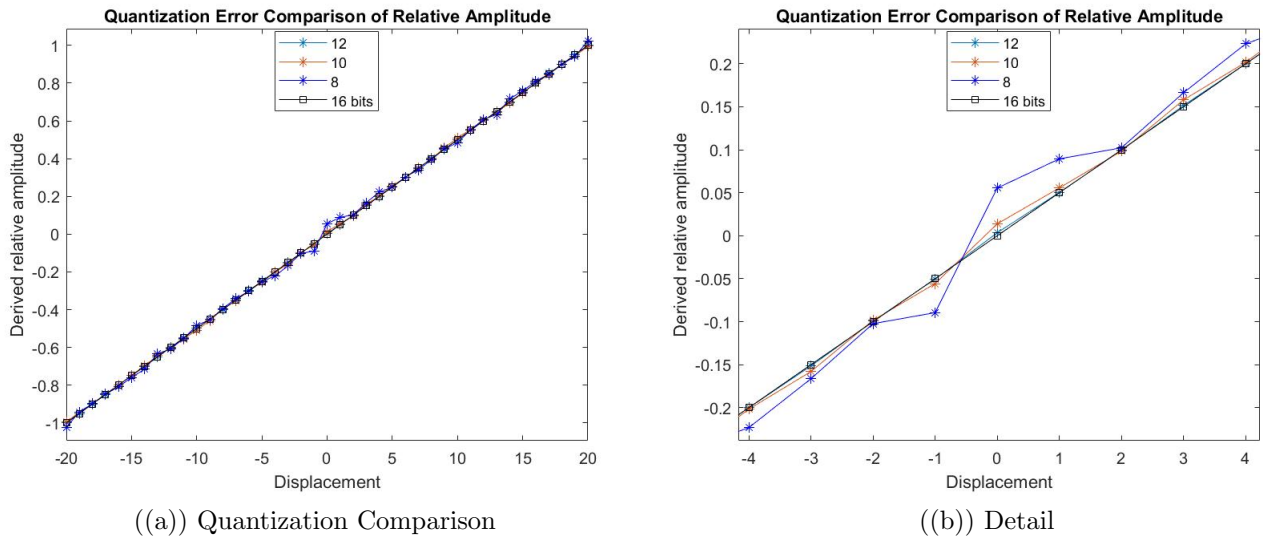


Figure 30: Comparison of Bit Levels

For the purposes of this simulation, it appears that 10 bits will be enough to process amplitude ratio information with only a small amount of error. The center of displacement is the area of primary concern; since the ratio involves a correlation with a signal very close to zero the highest potential for error exists in this range. For the 10 bit simulation, there is a small amount of deviation from the 16 bit ideal case. However, for higher fidelity analysis 12 bits are recommended for processing since it deviates very little from the 16 bit ideal case. A graph of the deviation of each case from the ideal case is included in Figure 31 below.

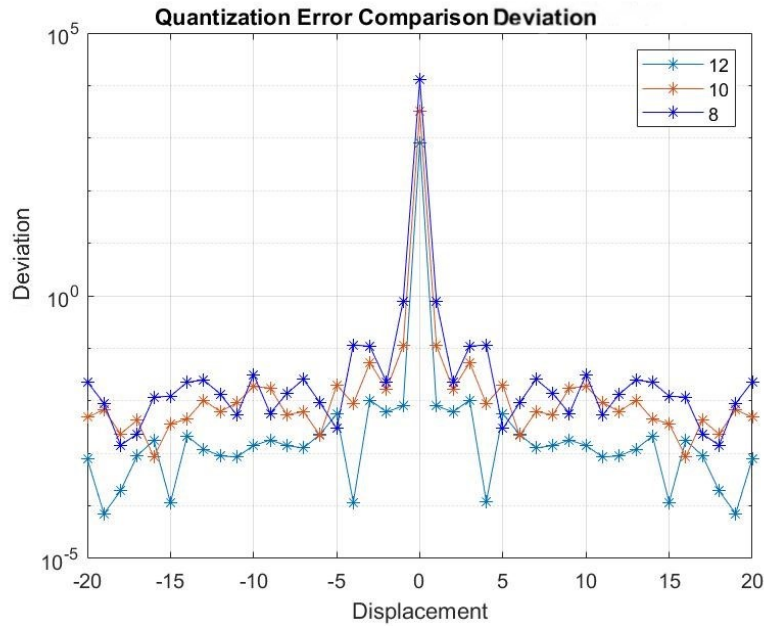


Figure 31: Deviation of Bit Values

For the purposes of this simulation, 10 bits would be the minimum required to maintain algorithm fidelity. The percent deviation of the 10 bit case from the 16 bit case is less than 10^{-1} percent, and the 12 bit case is one further order of magnitude less.

3.3.1.5 MIXED ERROR SOURCE ANALYSIS

When combining error sources the quantization, decrease in sample rate, and increase in noise all must be made to vary. The 25 sample per 10 cycle case was omitted due to the shifting of the slope. Since noisy signals are expected, the comparison was made with a constant noise level at 10% of maximum peak amplitude (0.1 V) for both random noise and current ripple.

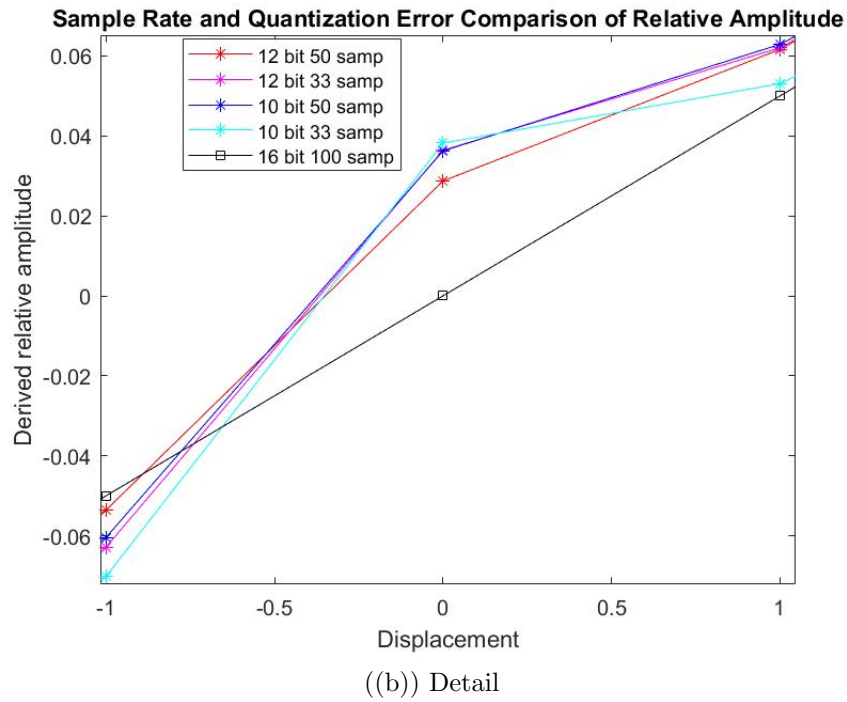
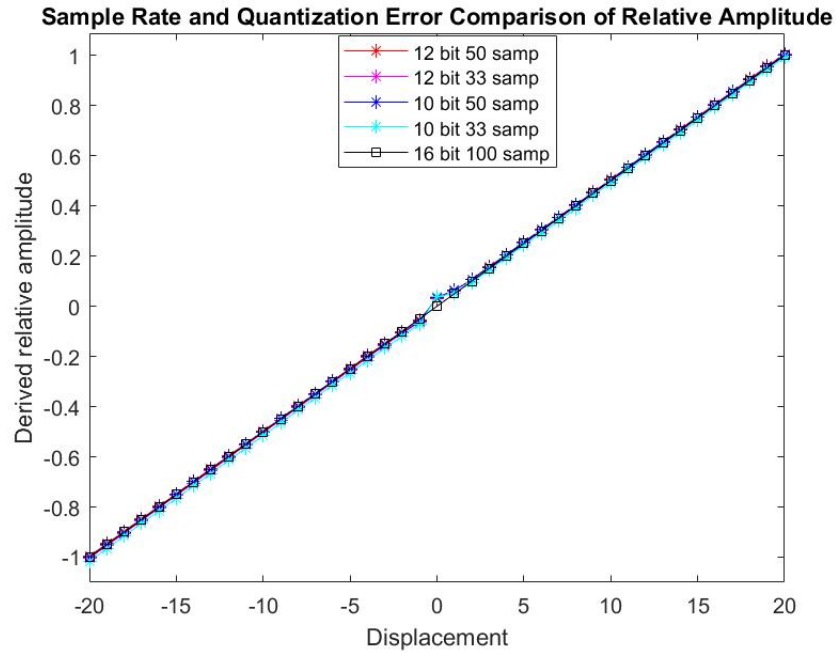


Figure 32: Comparison of Quantization and Downsampling Errors: 0.1 V Noise

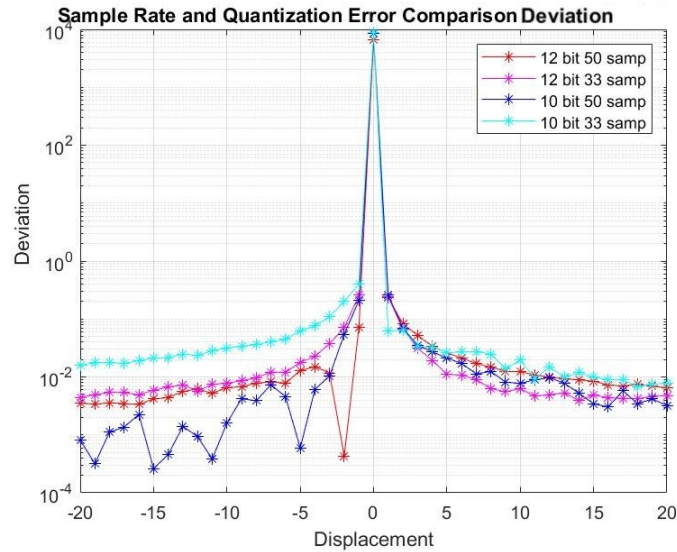


Figure 33: Deviation of Quantization and Downsampling Errors at 0.1 V Noise

The case with the highest deviation at the center point is the 10 bit, 33 samples per 10 cycles case as is evident in Figure 32. The 12 bit 33 sample case and the 10 bit 50 sample case exhibit almost the same deviation. When considering the deviation over the entire displacement, the 10 bit 33 sample case exhibits the most deviation for most locations, as is evident in Figure 33.

Based on this analysis, if the sample rate is expected to decrease significantly 12 bits are recommended. If the sample rate is not expected to go below half of the idealized rate, then 10 bits are sufficient.

CHAPTER 4

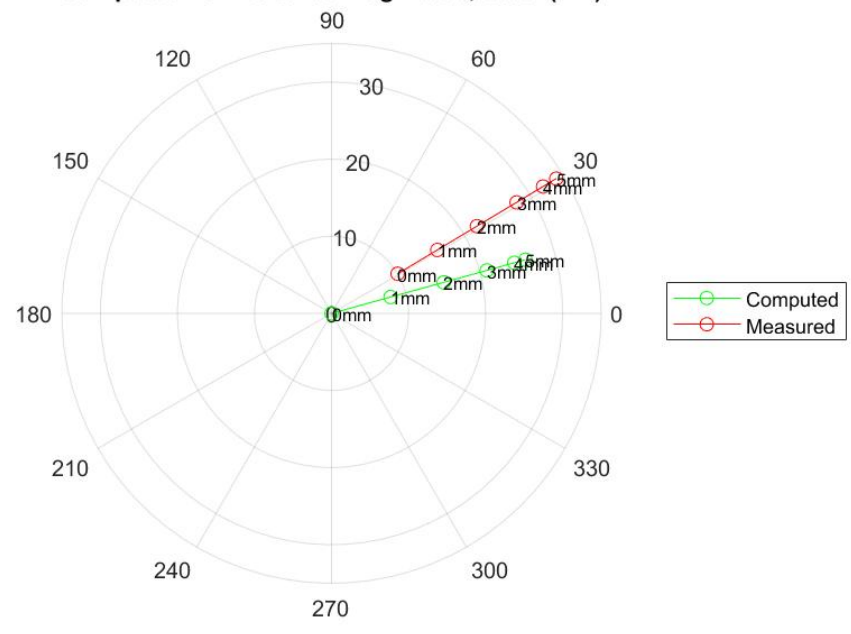
RESULTS

In the previous three chapters the theoretical formulations and simulations pertaining to the model EPS have been detailed. In this chapter, the results comparing theory with reality will be explored. First is an explanation of the comparison of COMSOL results with measured voltages. This is followed by details of the digital processing algorithm and how it handles real, discretized signals.

4.1 COMSOL ANALYSIS

In addition to the 1.25 inch steel sphere, two other materials used for comparison were a 1.25 inch tall aluminum cylinder and a 1.25 inch wide copper disc. The voltage magnitude was measured directly off each pickup coil in order to compare to the theoretical values in COMSOL. The phase angle was ascertained by utilizing the same data acquisition process as the steel sphere and calculating the phase angle with the EPS conditioning algorithm. Two polar plots are included below to demonstrate the difference between theory and reality of the three materials. In Figure 34, each point on the plot is labeled as it displaces from 0 to 5 mm.

Computed vs Measured Magnitude, Steel (mV)



Computed vs Measured Magnitude, Aluminum & Copper (mV)

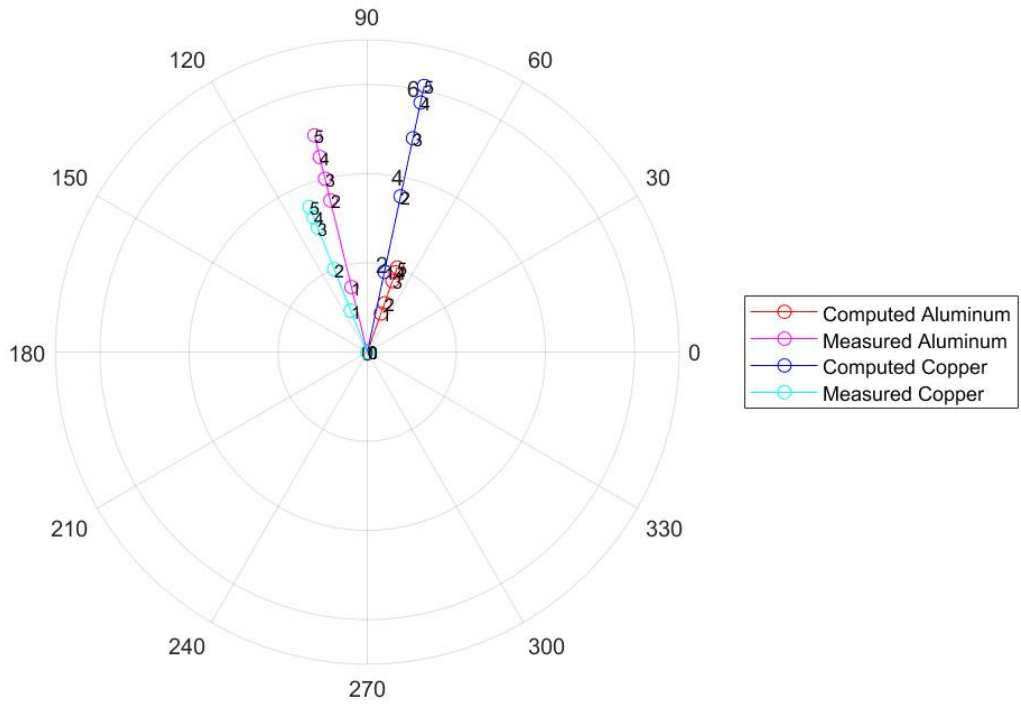


Figure 34: Comparison of Core Materials

The steel sphere was modeled as iron in COMSOL and the phase angle between excitation and output was found to be 0.27 radians or 15.47 degrees, more or less independent of core position. The aluminum cylinder exhibits a phase angle of 1.23 radians or 70.47 degrees and the copper disc exhibits a similar phase angle of 1.36 radians or 77.92 degrees. The measured voltage values were taken using a sensitive oscilloscope and moving the core 1 mm at a time. The measured phase values were ascertained with the digital processing algorithm.

When comparing the measured voltage to the theoretical voltage of the steel ball, the most notable difference is that the signal does not start at 0 V, the origin, at the center point. This is due to the very sensitive nature of the model EPS; while efforts were made to isolate any metallic objects from the model EPS, it is inevitable that a background signal is present. Since the EPS operates in an electromagnetic environment with metallic equipment in its vicinity, the presence of a background signal is expected. The background signal results in a small signal at the center where there should be no signal, or a null point; the MSBS EPS system at Langley has an analog nulling system for the purpose of nulling any background signals. The lack of a null point in the measurements is made apparent in Figure 35 which compares the theoretical COMSOL voltage magnitude to the measured voltage magnitude; the background signal also causes nonlinearity around the origin. However, the total lengths of the lines in Figure 34 are comparable and the phase values of the computed versus measured data are within 15 degrees of each other.

The measured and theoretical voltages of the aluminum cylinder and copper disc are quite similar considering the small size of the values. Since the amplitudes of the voltage for these two materials are so small, the first measured data point was assumed to be zero for consistency; however if the accuracy of the oscilloscope was higher the first measured data point would likely be slightly greater than zero. The phase values of computed versus measured data are within 35 degrees of each other.

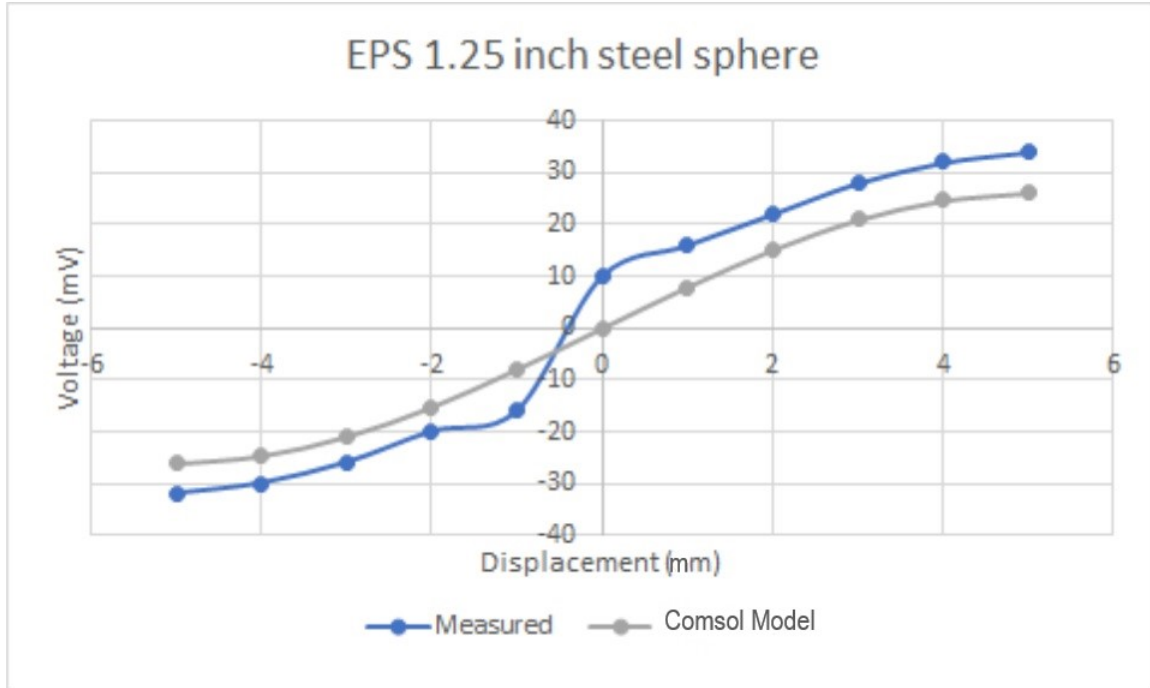


Figure 35: Comparison of Steel Voltage Magnitude: COMSOL vs. Measured

While the base voltage on the pickup coils is slightly lower in reality than in theory as calculated by COMSOL, the measured value is within 40 percent of the theorized value for most points. The deviation increases around the center point because accurate measurement at such small voltages is difficult, even with a sensitive oscilloscope.

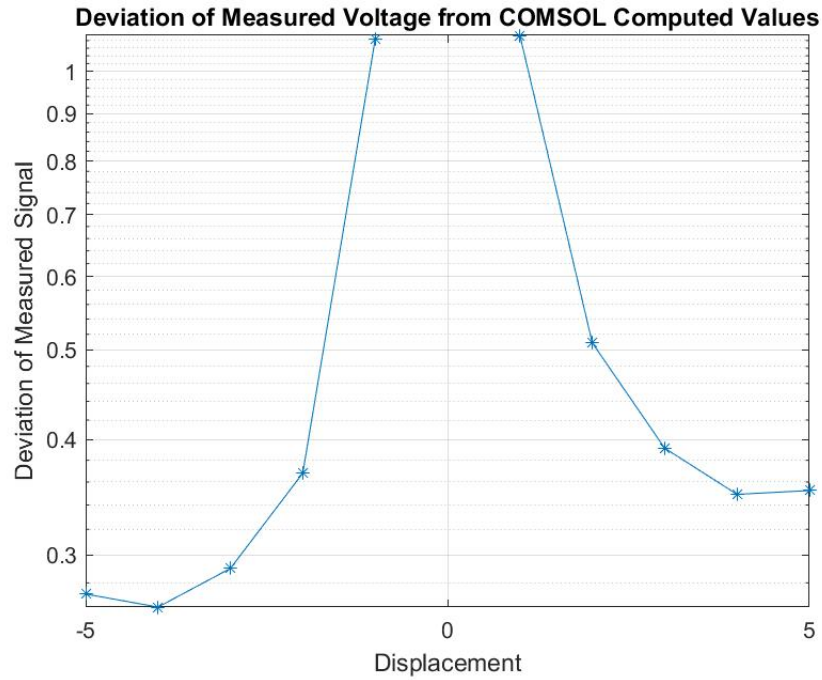


Figure 36: Deviation of Steel Voltage Magnitude: COMSOL vs. Measured

Since the theoretical measurement in COMSOL at the center point is 0 V, the deviation calculated for the center point is infinite and does not appear on the graph in Figure 36.

4.2 DIGITAL PROCESSING ALGORITHM

Since it has been established that the simulated algorithm has been proven to operate within the specified constraints, the same approach was applied for the algorithm to be implemented in real time. The objective is to formulate the algorithm in an efficient manner suitable for real-time execution on dedicated hardware, thereby minimizing latency while each consecutive set of samples are being processed. The processing algorithm is another MATLAB script using the same discrete sum representation of the cross-correlation of the reference and measured signals utilizing equations 15, 16, and 17. The discrete sum representations of the auto-correlations of each signal are used to find the amplitude at each data point. The full MATLAB script is given in Appendix D.3.

LabVIEW was used to collect discrete data at a rate of 50 kHz for 250,000 samples per data point; the total number of points for each 11 positions was 2,750,000 points. After these eleven pairs of signals from each data point were collected into a text file the data is uploaded into the MATLAB script for post-processing. The script calculates the cross-correlation of the reference and measured signals and the autocorrelations of each signal for each data point as the core displaces 1 mm at a time for a total displacement of 10 mm and a total of 11 data points for analysis. The amplitudes of each signal are calculated using the corresponding auto-correlation at each point; the phase difference is calculated using equation 9 at each point. The desired effect of this processing is for the cross-correlation to be at a positive maximum at one extreme of the displacement and at a negative maximum at the other extreme. The auto-correlation of the reference signal should remain nearly constant, therefore the amplitude of the reference signal should remain nearly constant at around 4 Vpp. The auto-correlation of the measured signal should be at a maximum at the extremes of displacement and a minimum at the null center, therefore the amplitude of the measured signal should do the same ranging from 600 mVpp to less than 50 mVpp.

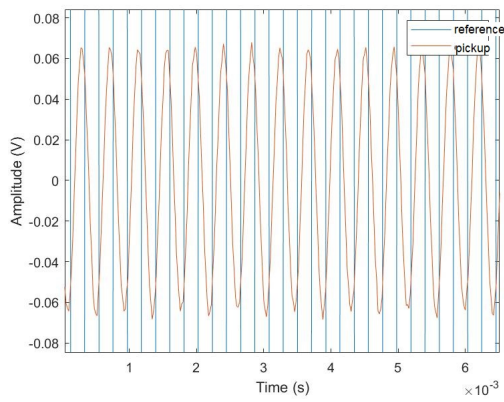
In the next section, the details and analysis of the digital processing algorithm are explained. This algorithm uses the data acquired from the NI-DAQ 9205, stored in a text file

as digital, discrete data. This algorithm was then tested for functionality within the same constraints and using the same techniques as the simulation algorithm.

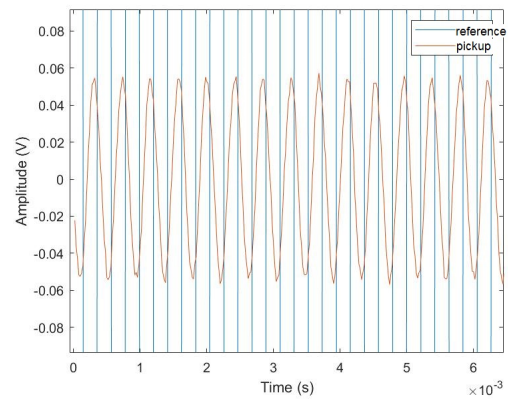
4.3 DATA ANALYSIS

The analog-to-digital conversion of signal data occurring in the NI-DAQ 9205 required some adjustments before the MATLAB processing script could use the data. The raw data had a slight negative offset on the pickup signal; see Appendix E for graphs of raw signals. The offset was removed using MATLAB's **detrend** function; for future reference if the acquired data does not exhibit an offset then this part of the code may be left out. The **detrend** function computes and subtracts the mean value from each time-domain signal.

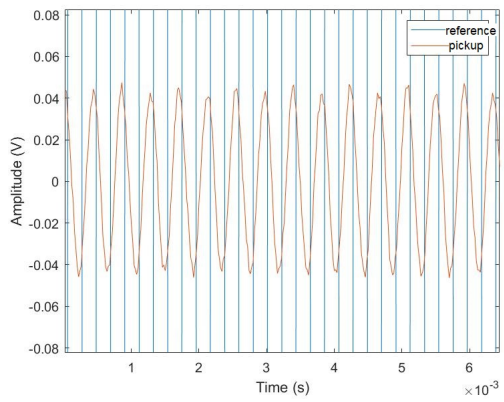
As the model core was made to displace from the connector end of the EPS to the other end it was necessary to use negative displacement units for half of the axis and positive displacement units for the other half. The negative displacement units correspond to the first five data points approaching the null center point and the positive displacement units correspond to the last five data points moving away from the null center. The signals for each of the 11 points are displayed in Figures 37 and 38. For ease of viewing, the graphs that display amplitude of the measured signal are shown in Figure 37: the red signal is the pickup signal, which varies in amplitude; the blue signal is the reference signal which remains constant in amplitude. In Figure 38 the peaks of both signals, and therefore the phase difference between them, are displayed.



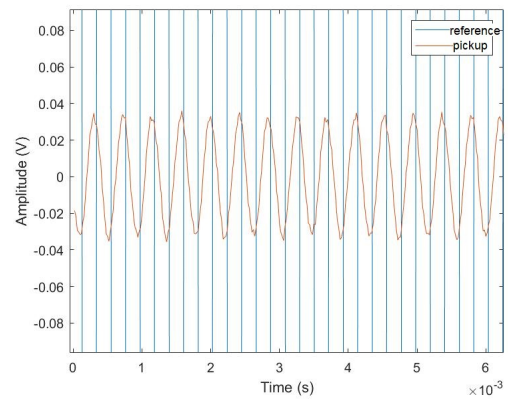
((a)) Core at -5 mm



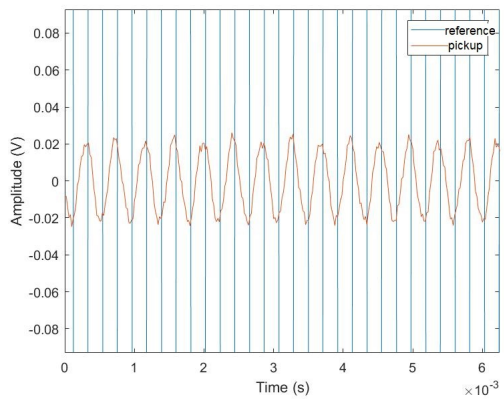
((b)) Core at -4 mm



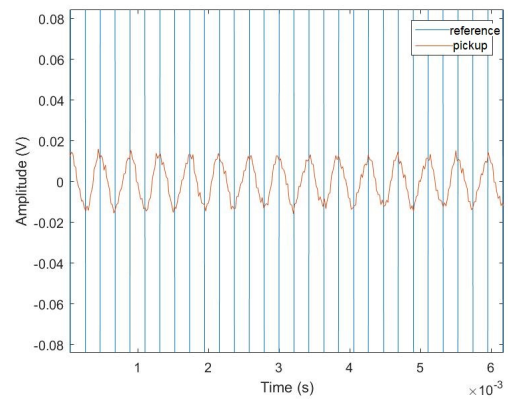
((c)) Core at -3 mm



((d)) Core at -2 mm

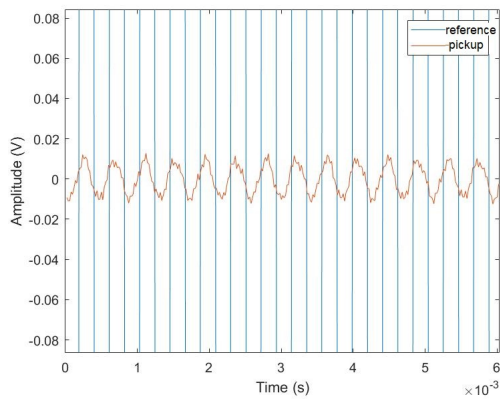


((e)) Core at -1 mm

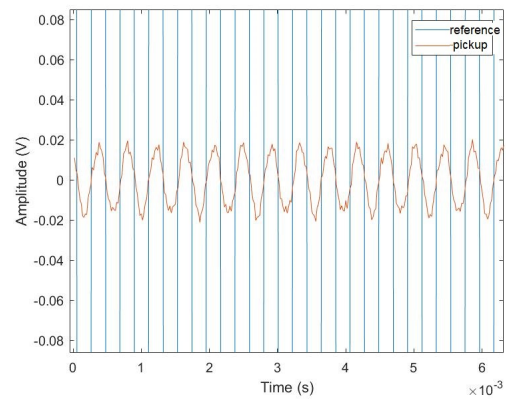


((f)) Core at 0 mm

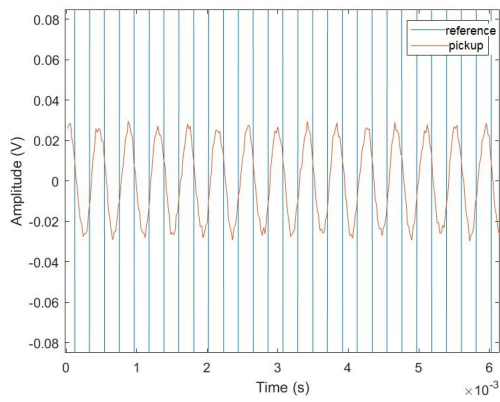
Figure 37: Data graphs as model core displaces 11 mm, Pickup Amplitude Detail



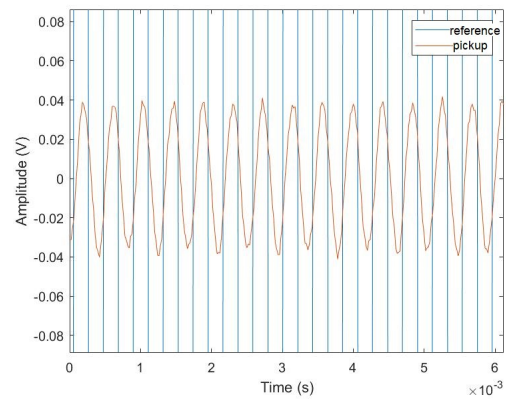
((g)) Core at 1 mm



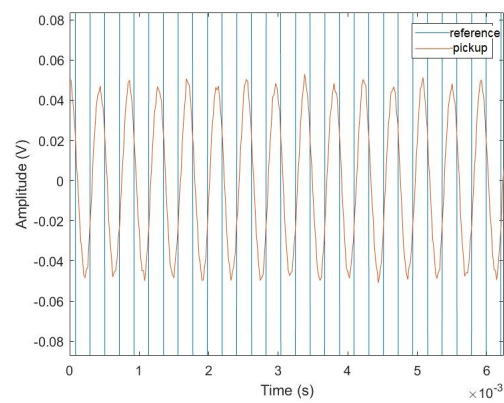
((h)) Core at 2 mm



((i)) Core at 3 mm

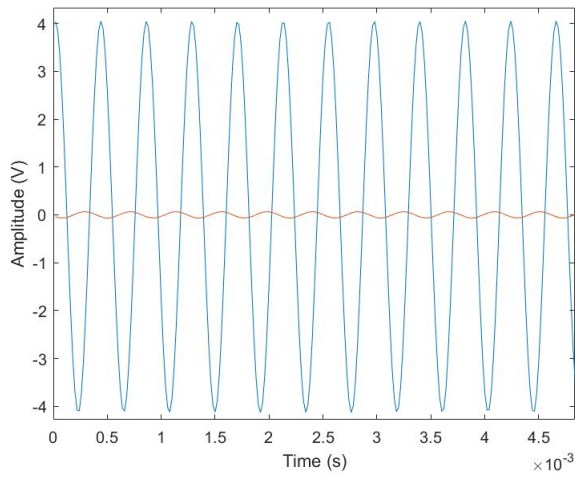


((j)) Core at 4 mm

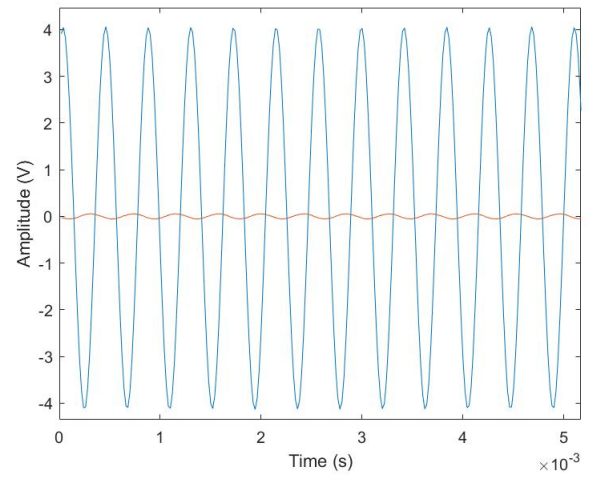


((k)) Core at 5 mm

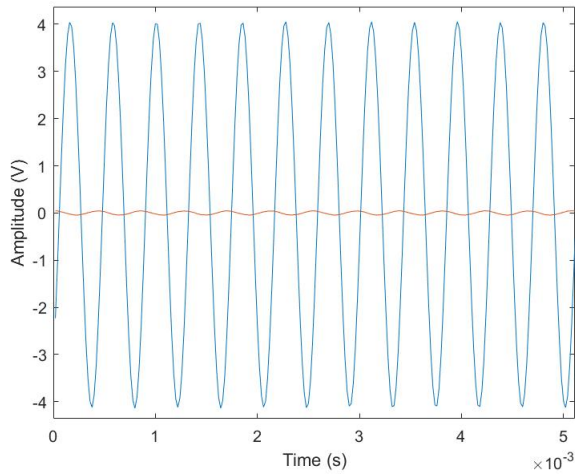
Figure 37: Data graphs as model core displaces 11 mm, Pickup Amplitude Detail



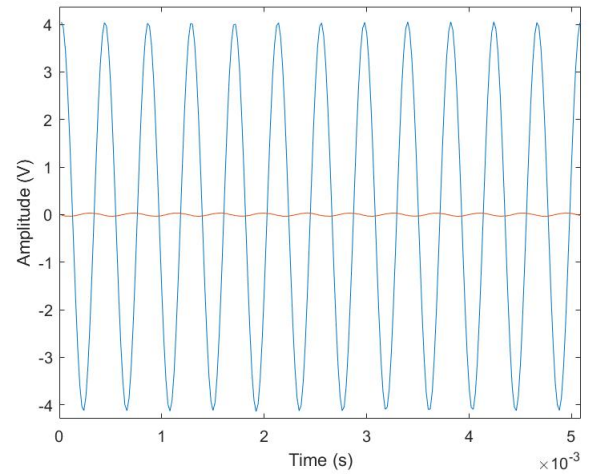
((l)) Core at -5 mm



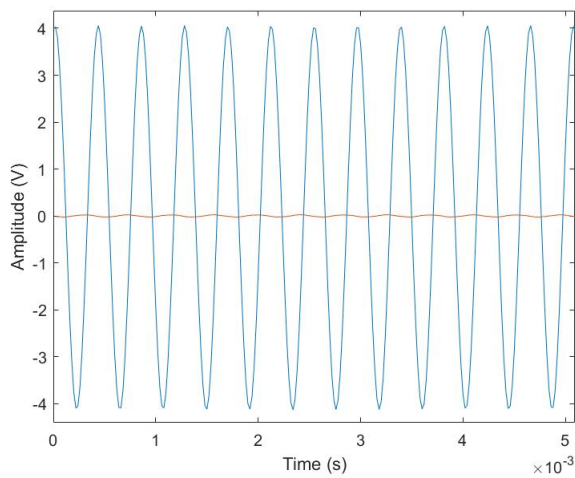
((m)) Core at -4 mm



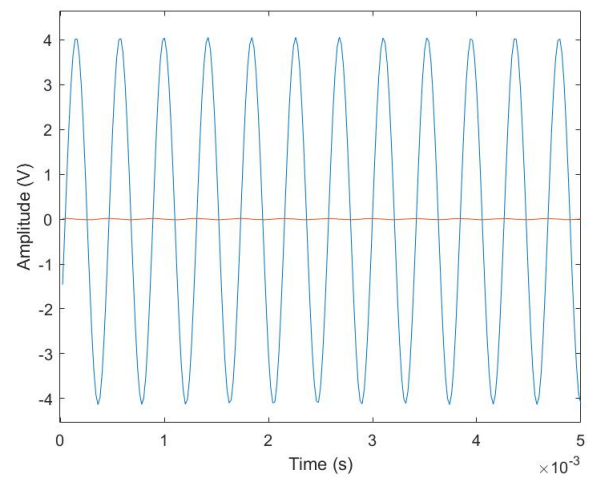
((n)) Core at -3 mm



((o)) Core at -2 mm

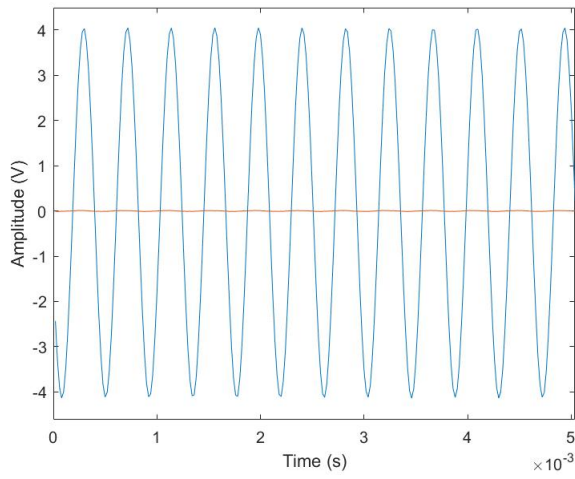


((p)) Core at -1 mm

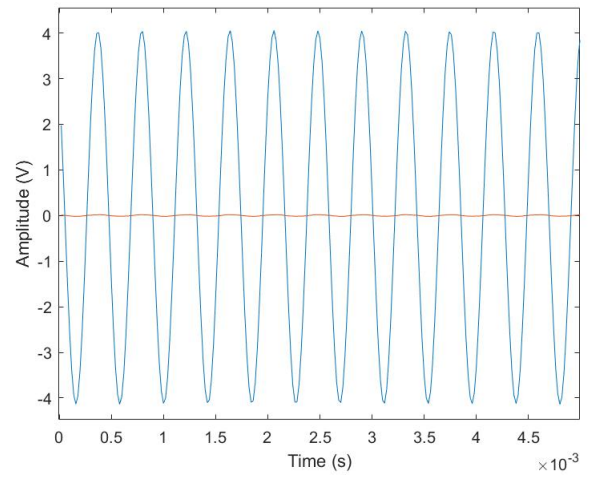


((q)) Core at 0 mm

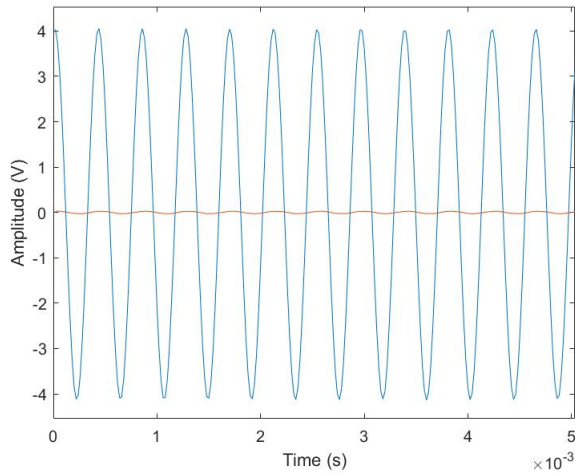
Figure 38: Data graphs as model core displaces 11 mm, Phase Difference



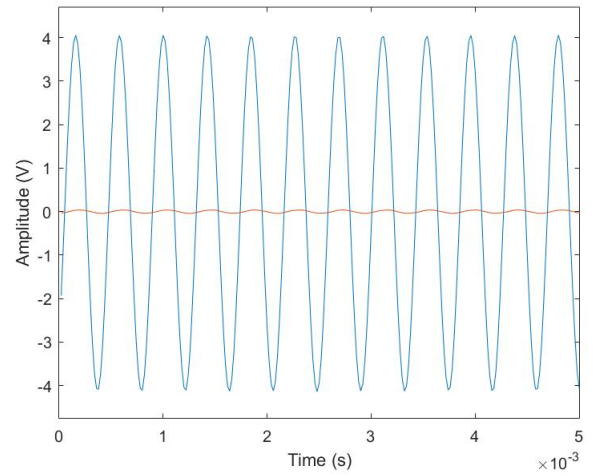
((r)) Core at 1 mm



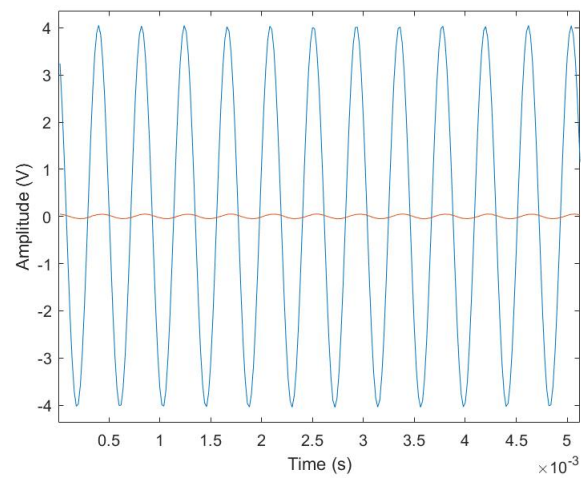
((s)) Core at 2 mm



((t)) Core at 3 mm



((u)) Core at 4 mm



((v)) Core at 5 mm

Figure 38: Data graphs as model core displaces 11 mm, Phase Difference

It is apparent that the pickup signal amplitude decreases until the null point and increases again as the model core displaces axially through the model EPS. The phase of the signal also changes from opposing in phase to aligning in phase; this causes the relative phase to change from negative to positive around the null point.

These graphical representations of each signal at the 11 data points of displacement were then used by the algorithm to determine the relative amplitudes and phases at each data point. Utilizing equations 15, 16, and 17 to find the discretized cross-correlation of both signals and the auto-correlation of each signal at every data point, the relative amplitude and phase of the measured signal with respect to the reference signal can be determined as they change with axial displacement.

4.4 ALGORITHM RESULTS

The signals in the previous section at each data point were sampled at 16 bits, 50 kHz, with 250,000 samples per position. They were processed with the MATLAB algorithm employing discrete cross correlation and autocorrelation to find the resulting relative amplitude and phase at each point. Since the reference signal has a constant amplitude and phase, the amplitude and phase of the measured signal can be determined using the relative amplitude and phase information. The relationship between relative amplitude as the model displaces axially is ideally linear with a sign change occurring at the origin.

The raw data collected required some correction before it could be processed with cross-correlation analysis. As was introduced in Section 4.1, a background signal was present on the pickup signal. At the center point location, the signal obtained had a small amplitude instead of being 0 V or a null point. It also introduced nonlinearity around the origin.

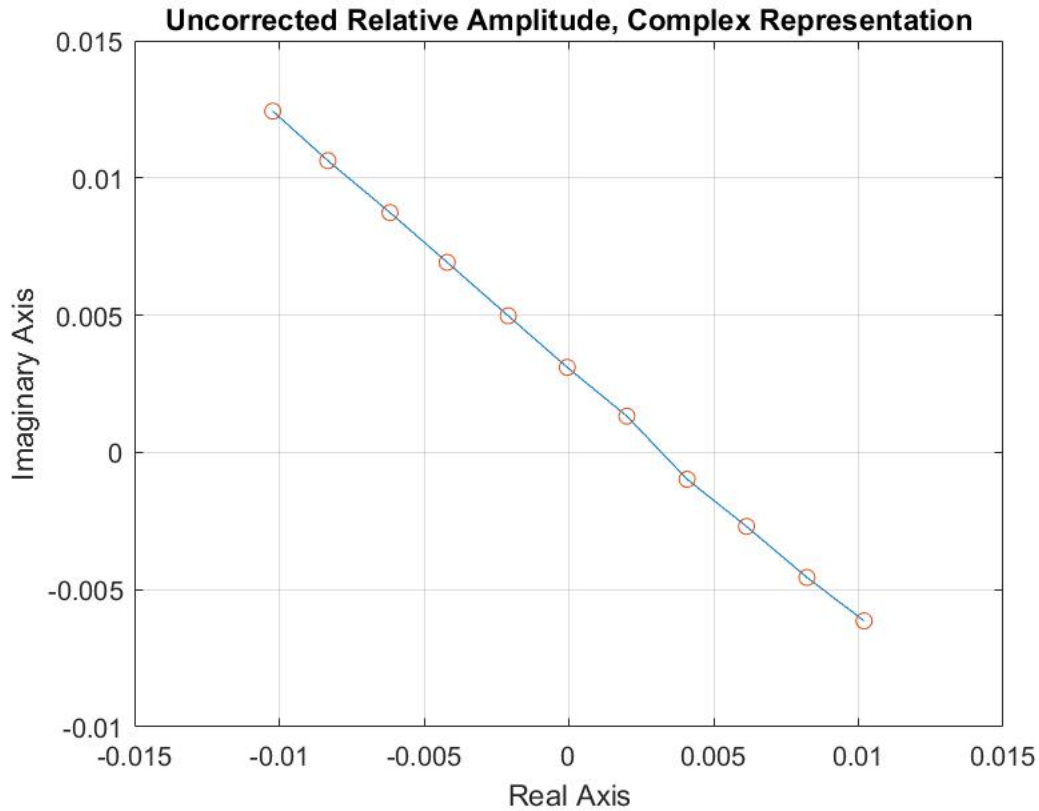


Figure 39: Raw Measured Relative Amplitude

Employing the theory of subtraction of phasors, the relative amplitude and relative phase values output by the algorithm were used to construct a series of complex numbers to represent a phasor. The midpoint of the series of complex numbers was not zero, hence the background signal was identified as the complex number at this center point. For this set of data, the value is $-0.00005332875 + 0.0030995i$. This complex number was subtracted from each of the series of complex numbers using phasor subtraction. The nulled version of figure 39 is shown in figure 40, where the background signal is removed.

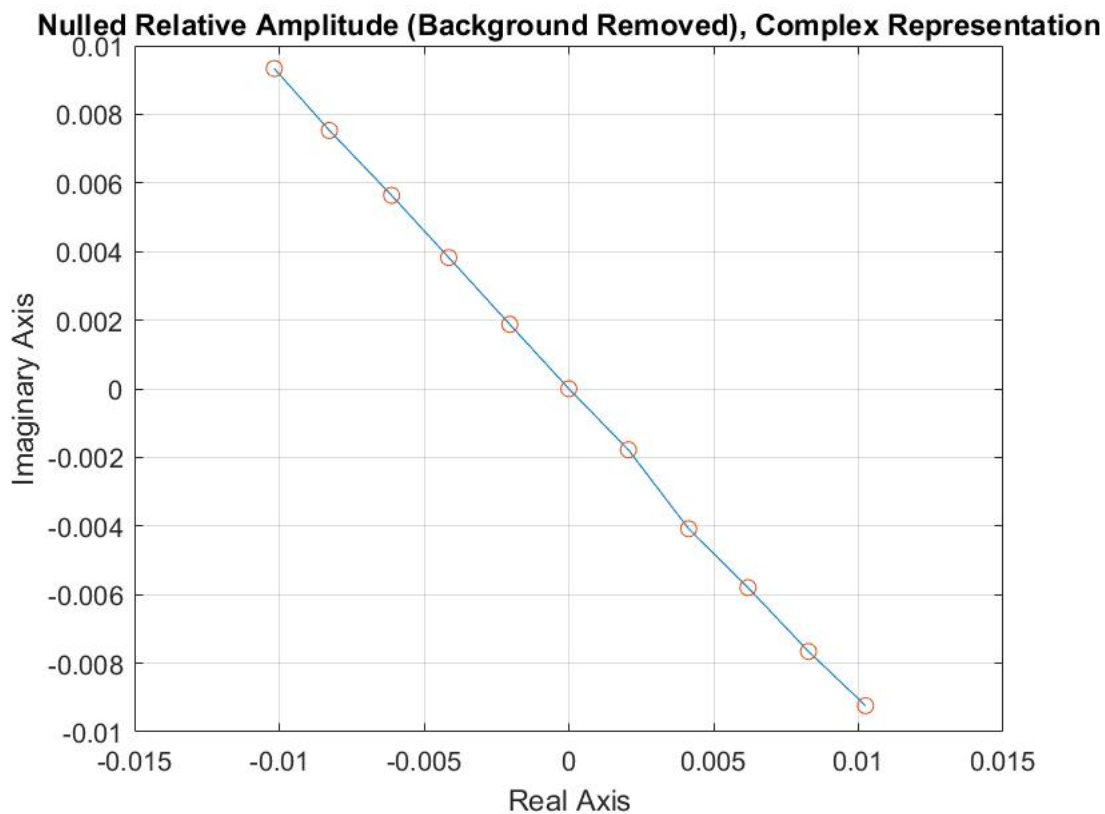


Figure 40: Nullled Relative Amplitude

A diagram representing the process is included in Figure 41.

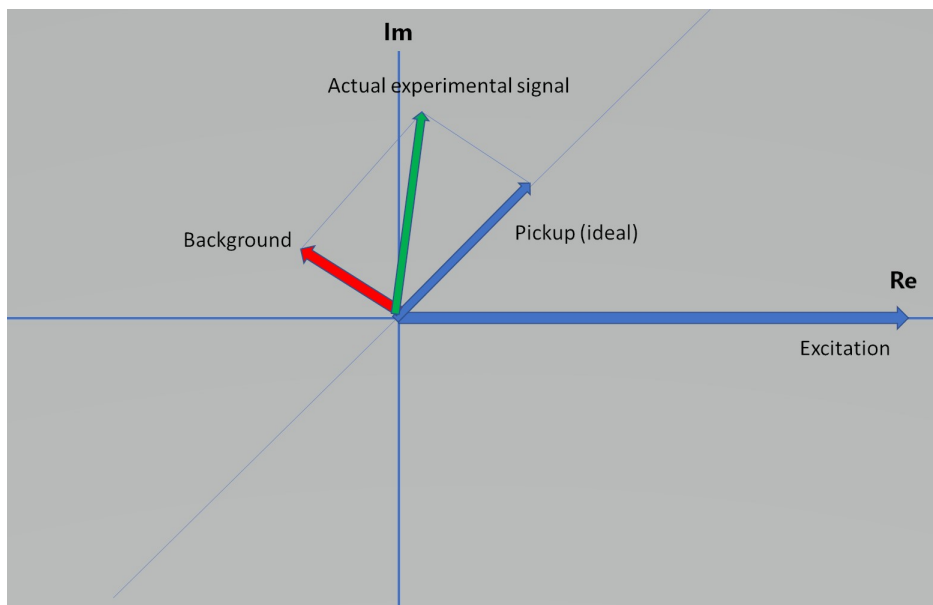
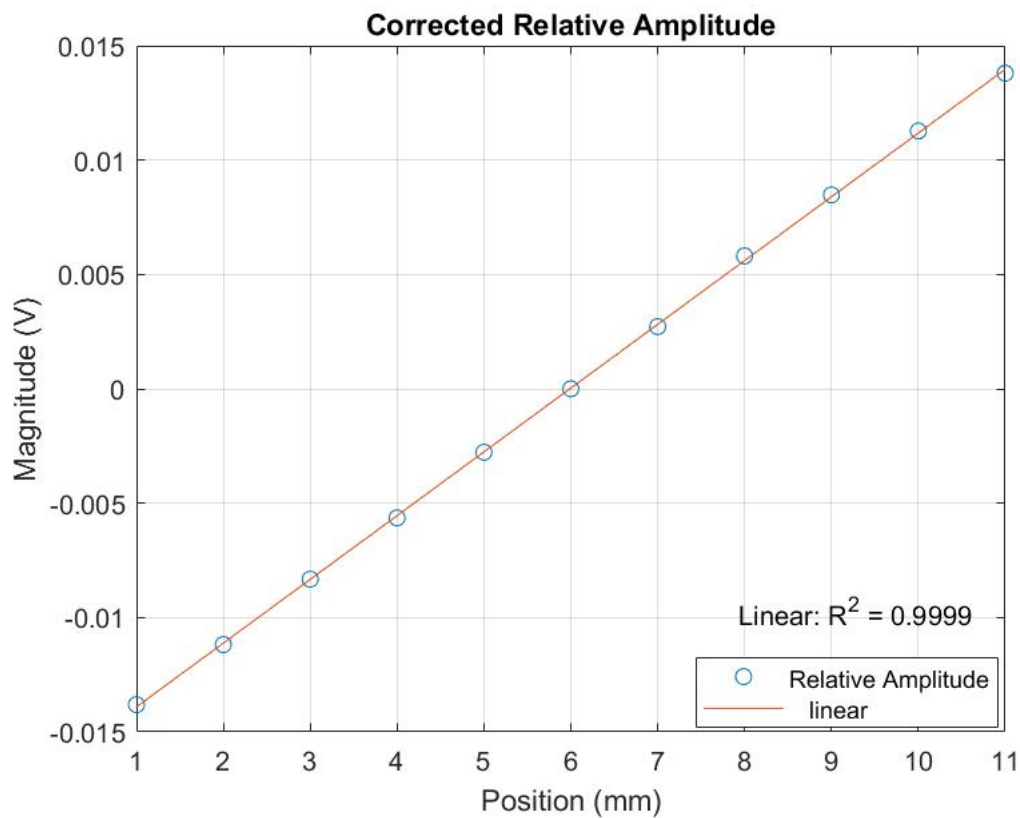


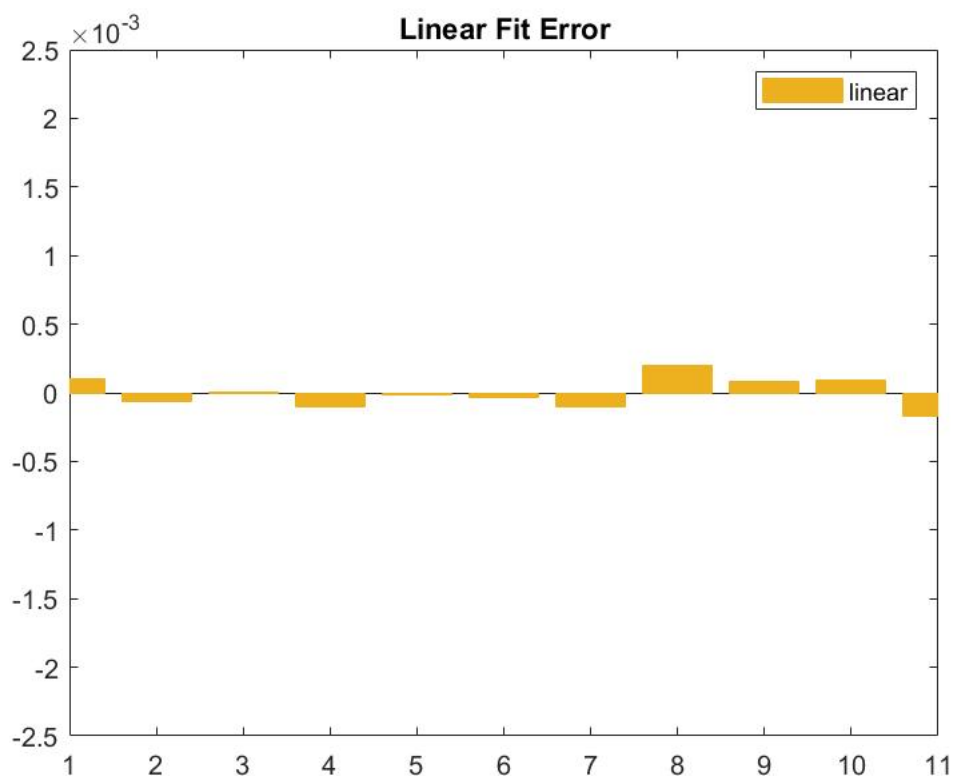
Figure 41: Phasor Representation of Background Signal Addition

In the diagram, the experimental signal is the result of the ideal pickup signal being altered by the addition of a background signal. In order to extract the ideal pickup signal, the background signal must be subtracted from the measured signal. This process is similar to the nulling process required by the analog system for EPS operation. Instead of making manual analog corrections to the pickup signal by nulling any background signals, the algorithm can identify the background signal and subtract it resulting in an ideal pickup signal for processing.

Once the background signal was removed, the resulting relative amplitude data points were plotted and a linear fit was imposed upon them. The linear fit calibration makes it clear that the displacement points in millimeters are accurate within margins specified in figure 42. It is important to verify that the derived displacement locations are an accurate portrayal of the measured displacement locations.



((a)) Relative Amplitude



((b)) Linear Fit Error

Figure 42: Corrected Relative Amplitude with Linear Fit

The MATLAB code used to find and remove the background signal is included in Appendix D.3.

The second set of corrections involve changing the relative phase and amplitude values within the algorithm's functionality. First, the relative phase must be corrected. Since the relative phase is found using equation 9, the arccosine can only output angles between 0 and 180 degrees. In order to allow for the relative phase angles to span over a 180 degree difference as expected, the last five angle measures were shifted by subtracting them from 2π . In this way, the first values of relative phase at 137 degrees and the last values at -43 degrees span a total of 180 degrees. Another correction must be made to the relative amplitude values to ensure a linear relationship. The slope of the first half of the data points must be negated in order to reflect an increase in amplitude from the furthest point to the center point. When the relative amplitude of the first five points is negated, the expected linear relationship occurs.

The results of both the cross-correlation and auto-correlations at each data point contribute to the phase difference information as specified in equation 9. The results of each autocorrelation contribute to the relative amplitude information as specified in equations 16 and 17. The resulting relative amplitude and phase difference calculated at each data point is tabulated below:

Displacement (mm)	Relative Amplitude	Phase Difference (Degrees)
-5	-0.0138	137.4456
-4	-0.0112	137.6978
-3	-0.0084	137.3174
-2	-0.0056	137.4276
-1	-0.0028	137.3061
0	0.00004	91.3156
1	0.0027	-39.8396
2	0.0058	-44.5625
3	0.0085	-43.0654
4	0.0112	-42.8268
5	0.0138	-42.0263

Table 2: Table of Relative Amplitude and Phase Difference Values

The graphical representation of this data is displayed in Figure 43.

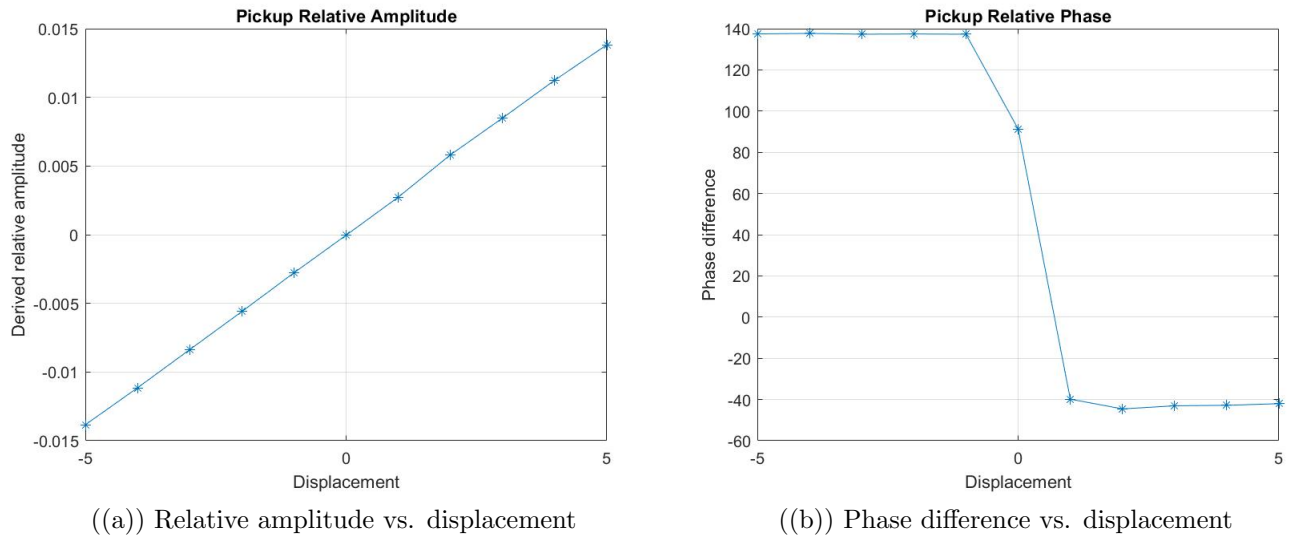


Figure 43: Phase Difference and Relative Amplitude Graphs

The phase difference values correspond with the phase difference of the signal graphs at each data point. As is evident by the graphs in Figure 43, the phase values of the first three data points are almost the same. The pickup signal leads the reference signal by around 137 degrees. Once the model core approaches closer to the midpoint, the phase difference decreases by more each time with each data point. This is consistent with the behavior of

the signals in Figure 37. The phase then decreases to 91 degrees at the midpoint, it then jumps by about 134 degrees so that the pickup signal is more in phase with the reference signal at a -43 degree difference. This is also consistent with the behavior of the signals in Figure 38.

4.5 DESCRIPTIVE ANALYSIS

Since the algorithm has been proven to be operational for the data acquired, it was necessary to test it for robustness with similar techniques utilized for the simulation algorithm. The algorithm was tested for quantization error, discretization error, noise sensitivity, and windowing error. In order to provide realistic conditions for processing, the record length was shortened to around 10 cycles long, or 250 samples, for quantization and downsampling.

4.5.1 QUANTIZATION ANALYSIS

The algorithm was computed utilizing 10, and 12 bits with 16 bits representing the ideal case. Once the quantized signals were then put into the algorithm to be processed with correlation, the amplitude ratio graphs were compared to analyze the quantization error. As was mentioned above, the center area of the amplitude ratio graph is of most concern and is where the highest concentration of quantization error will be represented. The graph of the amplitude ratios for each bit case is included in Figure 44.

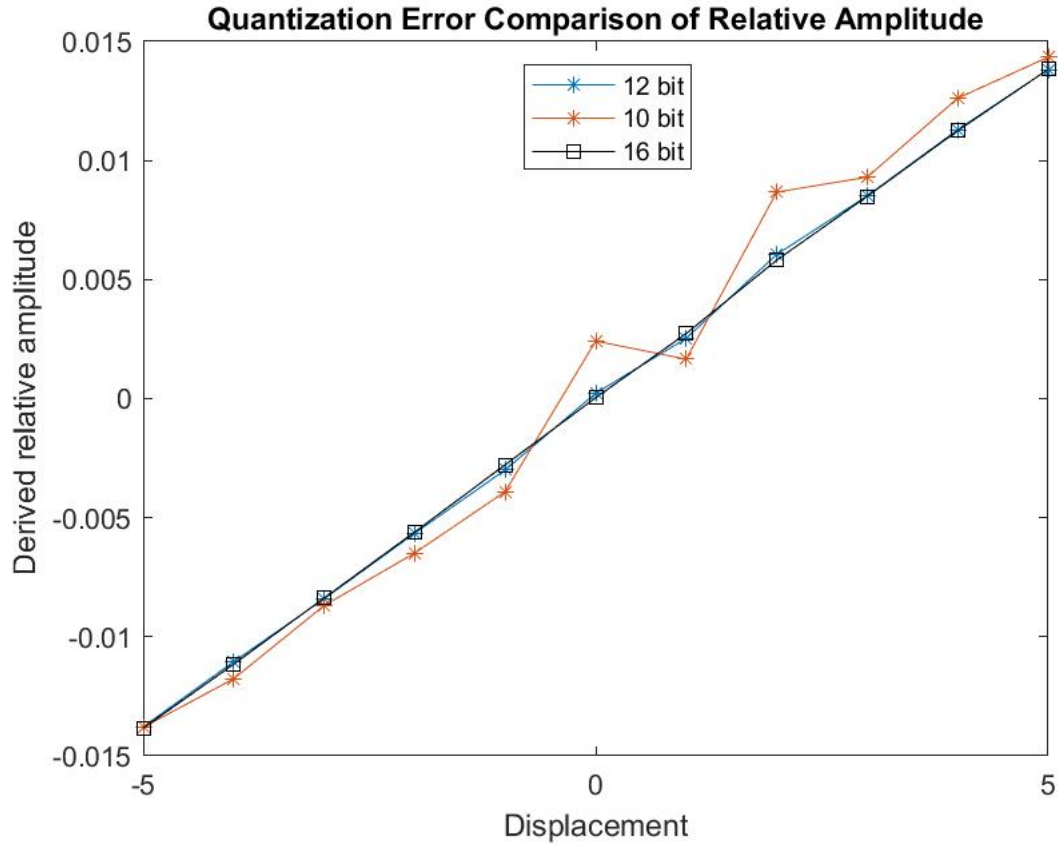


Figure 44: Quantization Error of Relative Amplitude Graphs

As is evident in Figure 44, the 10-bit quantization error introduces nonlinearity to most of the relative amplitude graph. However, the 12-bit quantization error is minimal for the entire signal. Each bit case's deviation from the 16-bit case is plotted in Figure 45. The processing algorithm was run once for each bit case, then each case's relative amplitude was compared to the 16-bit ideal case's relative amplitude on a point-by-point basis to compute the deviation for each bit case.

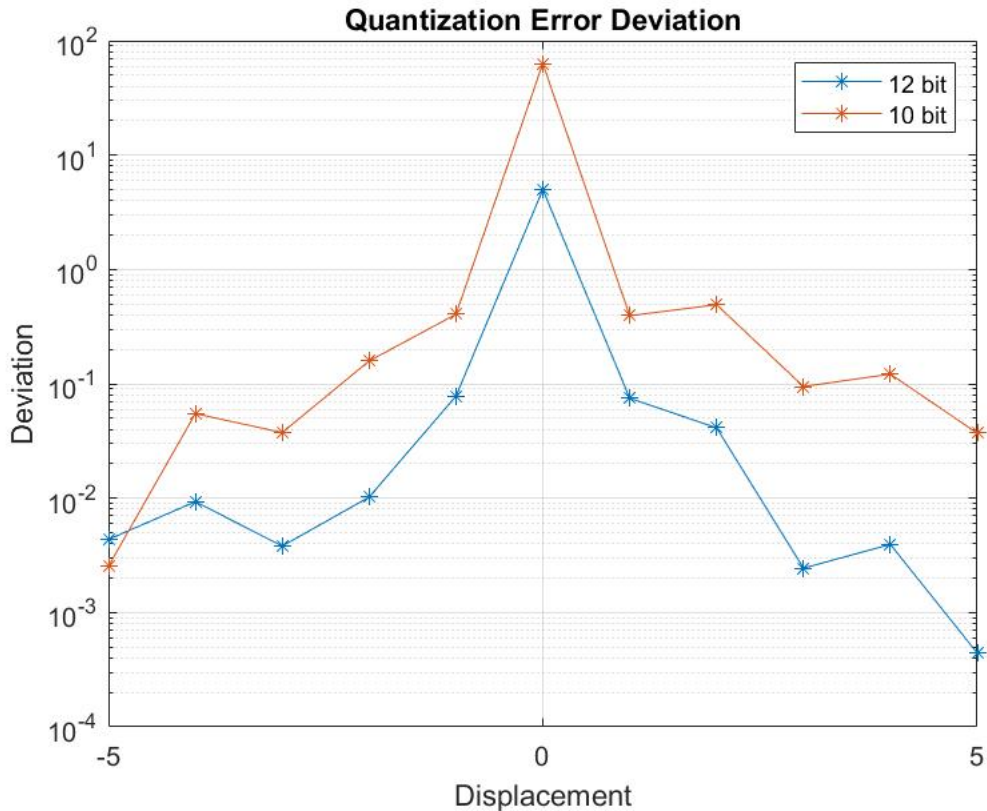
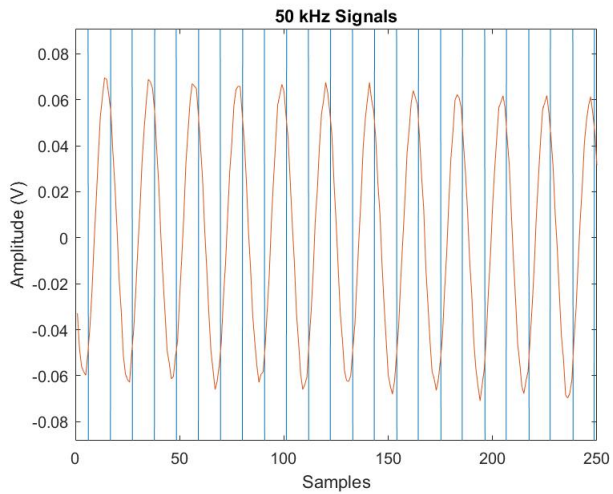


Figure 45: Deviation from 16 Bit

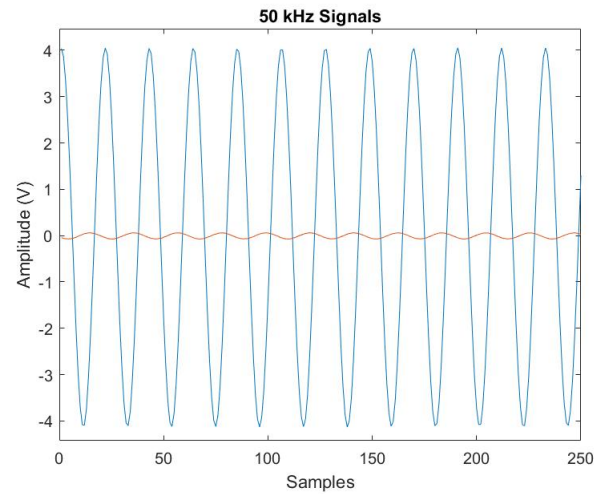
As the quantization level increases in bits, the error decreases by one order of magnitude. The 12-bit quantization outperforms the 10 bit case. For analog-to-digital conversion purposes, if a 12-bit processor is easily found it would be preferable to a 10-bit processor.

4.5.2 SAMPLE RATE ANALYSIS

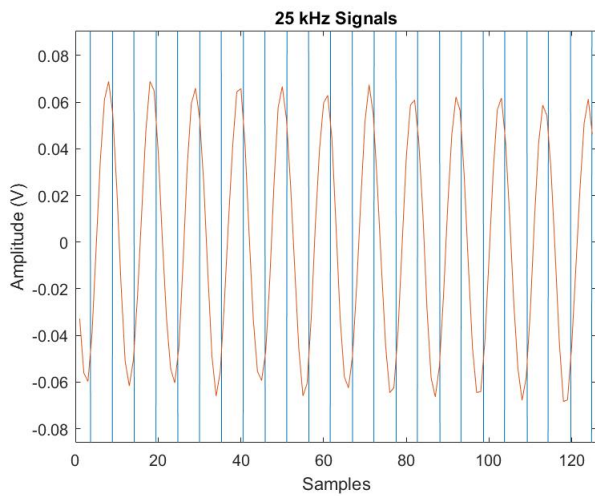
Discretization error was ascertained by decreasing the sample rate from its original rate of 50 kHz. Multiple attempts were made by decreasing the sample rate until the algorithm began to degrade. The first attempt was at half the original sample rate or 25 kHz which is slightly more than 10 times the carrier signal rate of 2.36 kHz; there was barely a difference in the results of the algorithm. Figure 46 shows the difference between the original signals and the downsampled signals at half the sample rate. Again, the blue signal is the reference signal, the red signal is the pickup signal.



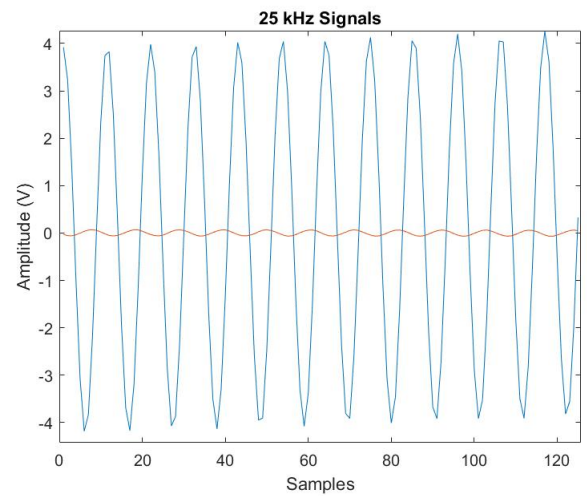
((a)) Pickup Signal (Red) Detail, 50kHz



((b)) Both Signals, 50kHz



((c)) Pickup Signal (Red) Detail, 25kHz

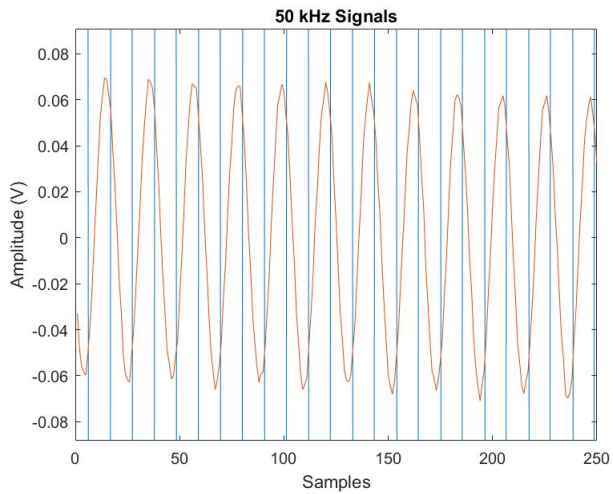


((d)) Both Signals, 25kHz

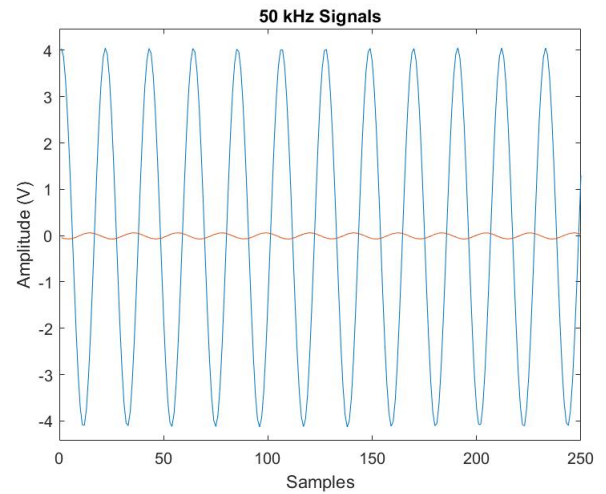
Figure 46: 50 kHz vs. 25 kHz

At this decreased sample rate, the discretization of the signals has led to minimal loss of fidelity. However, the amplitude ratio and phase difference data remains very close to the non-downsampled data.

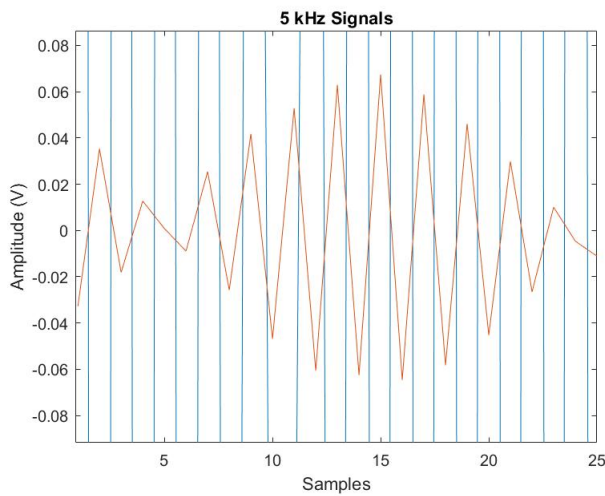
When the discretization error is increased, the algorithm breaks down at one tenth the original sample rate, or at 5 kHz. The signals sampled at 5 kHz are included in Figure 47 for comparison.



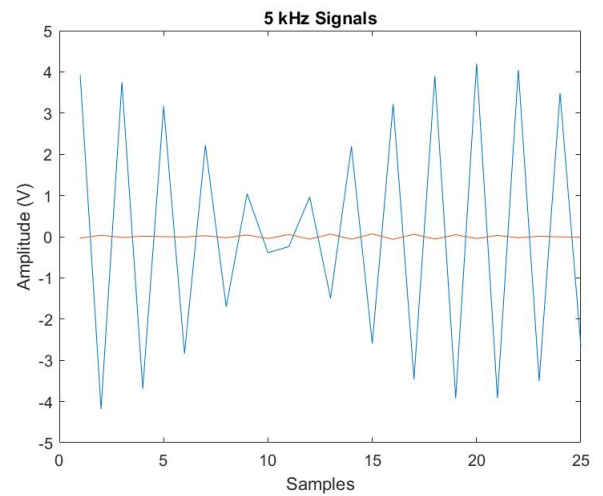
((a)) Pickup Signal (Red) Detail, 50 kHz



((b)) Both Signals, 50 kHz



((c)) Pickup Signal (Red) Detail, 5 kHz



((d)) Both Signals, 5 kHz

Figure 47: 50 kHz vs. 5 kHz Sample Rate

This breaking point at 5 kHz is consistent with the simulated data algorithm's performance. Since the simulated data was sampled at 10 times the number of total samples, corresponding to a theoretical sample rate of 23.6 kHz, the breaking point was at a sample rate of 4.72 kHz.

As a result of the decreasing sample rate, the relative amplitude graph displays a loss of linearity. A comparison of the relative amplitude graphs is included in Figure 48.

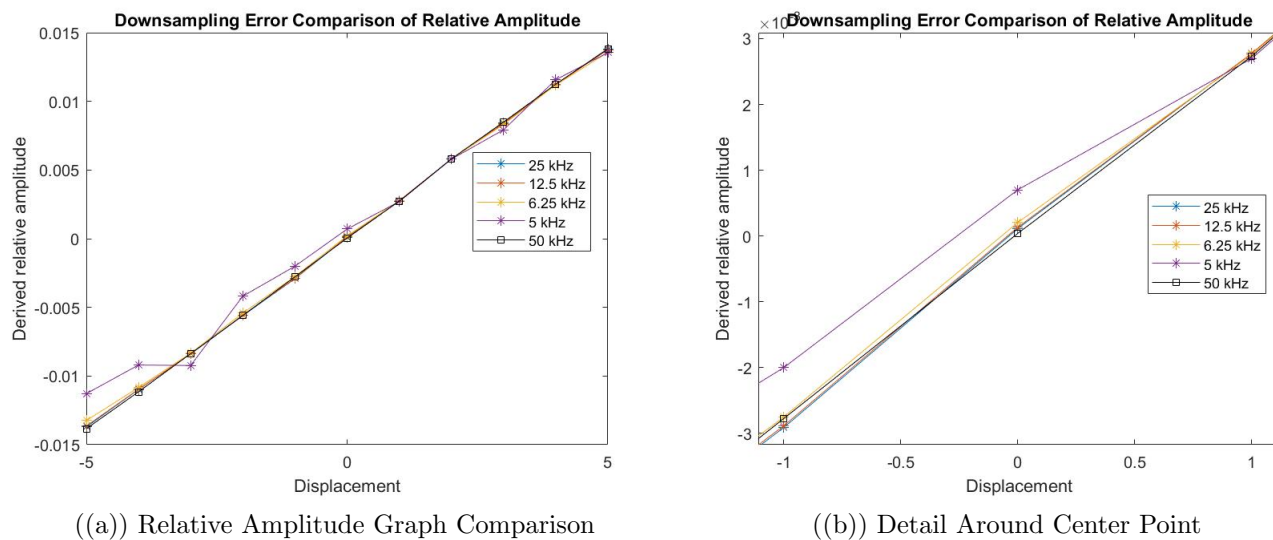


Figure 48: Downsampling Comparison of Relative Amplitude

Figure 49 shows the deviation of the relative amplitude derived at each sample rate from the 50 kHz case.

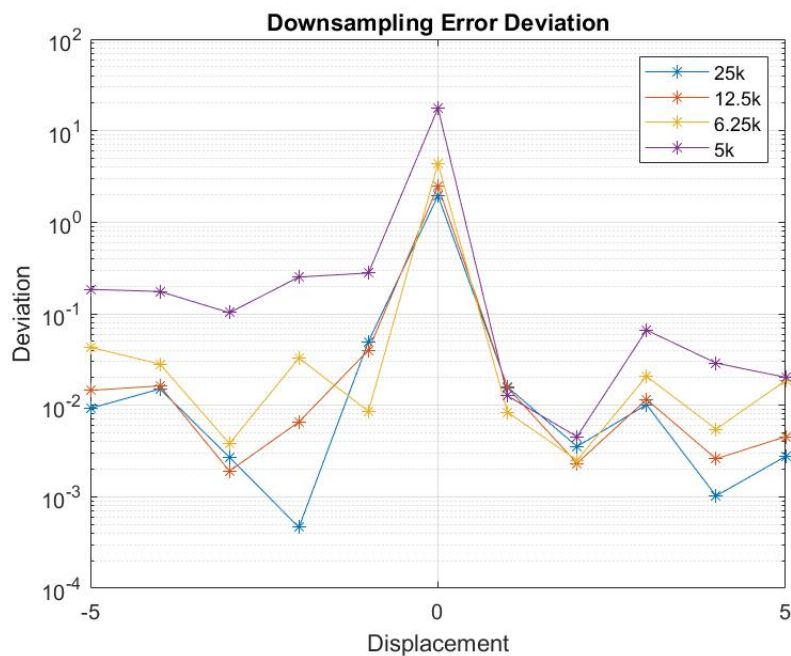


Figure 49: Downsampling Comparison Deviation

To preserve a maximum amount of fidelity, the sample rate should remain above or at 10 times the driving frequency. However, it is encouraging to note that there is minimal

discrepancy when sampled at around half the driving frequency; the deviation of the 12.5 kHz sample rate case remains under 10^{-1} for most data points.

4.5.3 RECORD LENGTH ANALYSIS

The next source of error to be explored is window length error. In Chapter 3.3.1.3, it was demonstrated that reduction in window length had no effect even down to one cycle. However, the simulation algorithm used idealized signals. Since the real algorithm utilizing data does not, this cannot be an expected result.

The original data signals were sampled at 250,000 samples per channel, therefore the total number of samples was reduced to create shorter windows of signals sampled at the same rate of 50 kHz. The shortened window signals were processed and each case's relative amplitudes are compared in Figure 50. The window lengths range from the original 250,000 samples down to 25 samples.

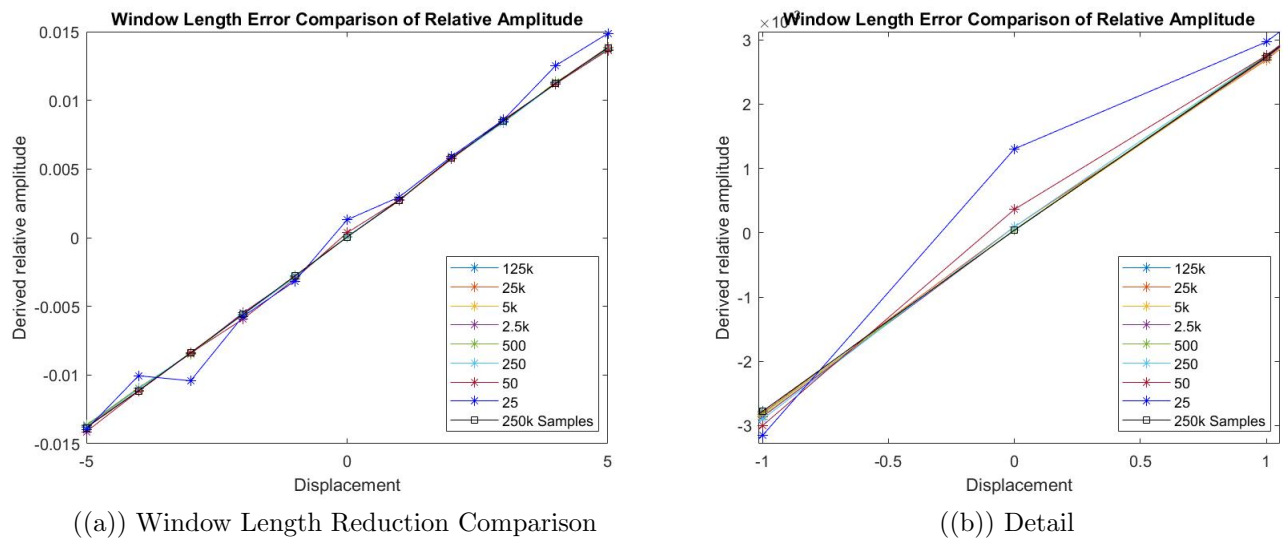


Figure 50: Window Length Comparison

The deviation of each case compared to the original window length is shown in Figure 51.

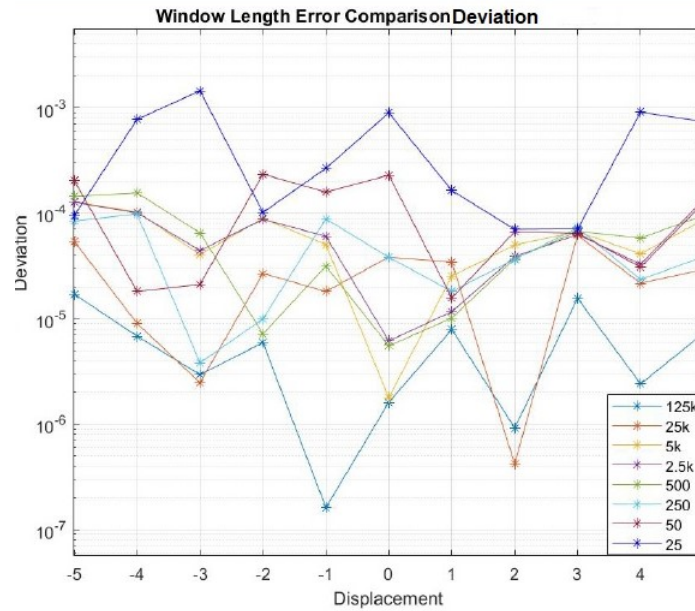


Figure 51: Window Length Comparison Deviation

The algorithm starts to break down at a window length of 25 samples, or just over one cycle of the excitation signal. However, window lengths above 250 samples have a deviation from the ideal of at or below 10^{-3} . It is recommended that window length remains at or above 250 samples per channel, or around 10 cycles of the excitation signal, for optimal precision in relative amplitude data.

4.5.4 NOISE ANALYSIS

The final source of error to be tested is noise put on the signals as they are being recorded. Since the MSBS and EPS are in an electrically noisy environment, this source of error is always a factor. In order to quantify the amount of noise on the signals as they were recorded as digital data, the 250,000 sample records were split into twelve records of 20,000 points in length. Each length was then run through the algorithm and the relative amplitudes of each set was compared. The standard deviation quantifies the difference between all twelve sets and represents the amount of noise.

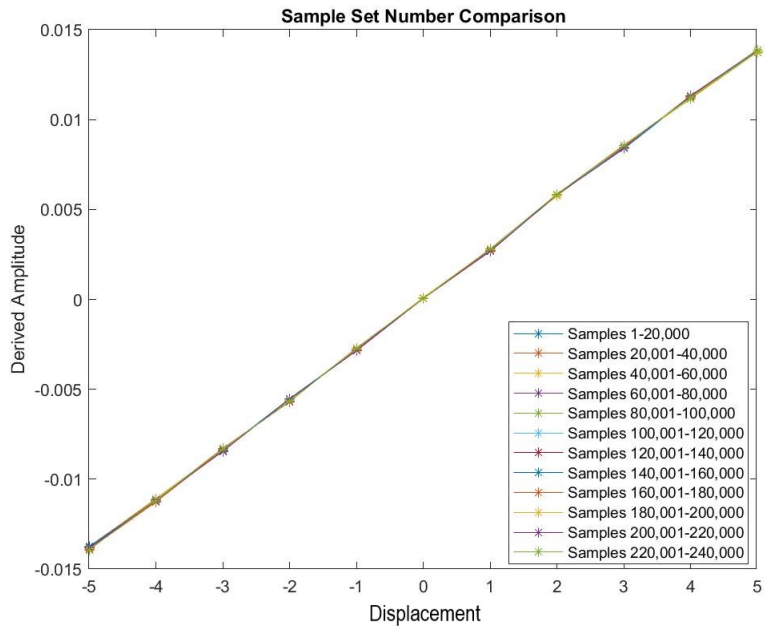


Figure 52: Set Relative Amplitude Comparison

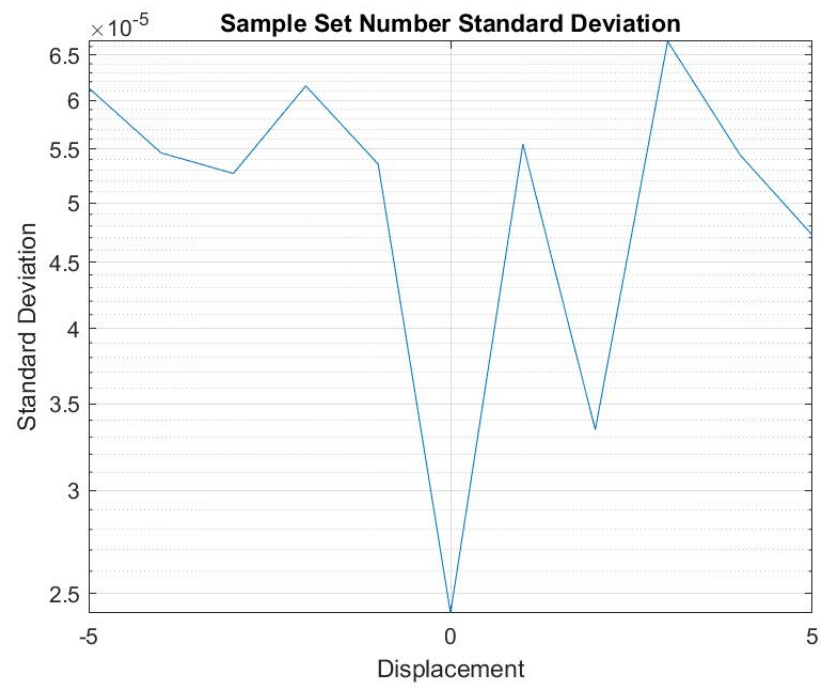


Figure 53: Set Comparison Standard Deviation (V)

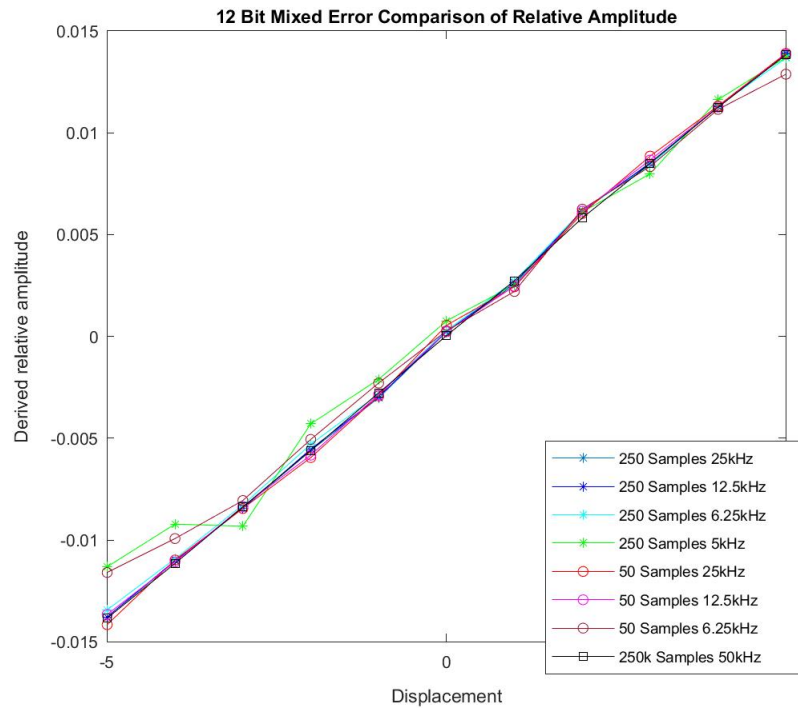
It is encouraging that the difference between each set of samples differs so little. The standard deviation of the sets taken together remains below 10^{-5} V. The noise rejection

inherent in cross correlation analysis has been made manifest.

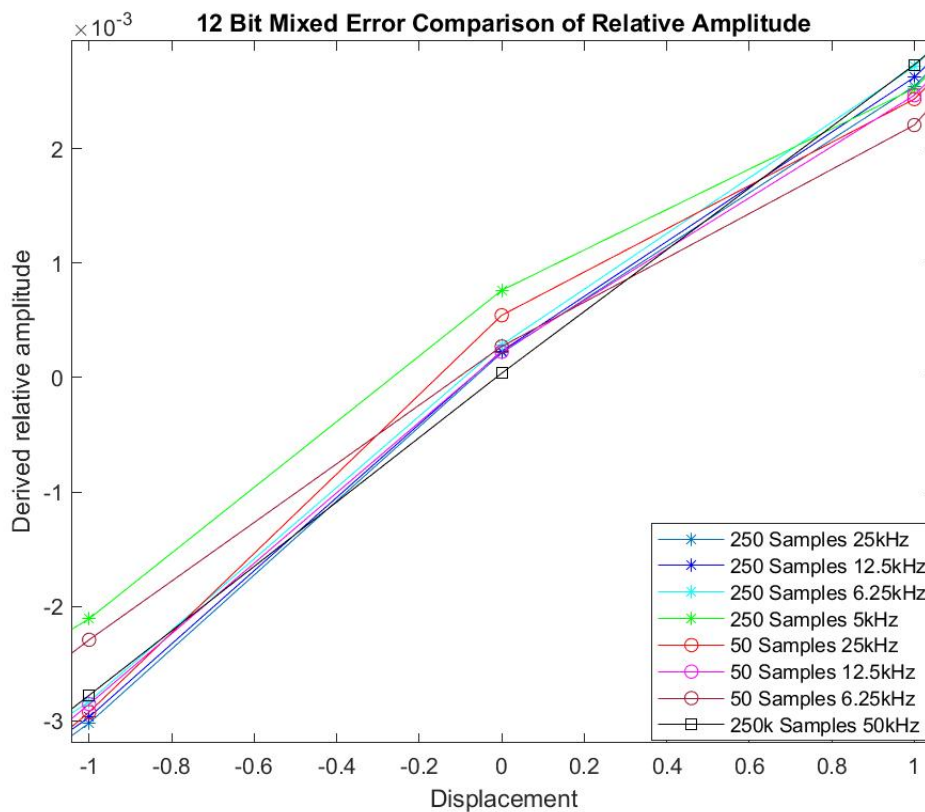
4.5.5 MIXED ERROR SOURCE ANALYSIS

The final step in the error analysis is to combine all sources of error. This was accomplished by first using the noisy 250,000 sample pair of signals, then reducing the window length of each. Then the signals are discretized and finally quantized.

When testing the algorithm to ascertain its breaking point, two cases were tested. Since the 12 bit quantization and 10 bit quantization had such marked deviation from each other, the mixed error sets were broken into 12 bit and 10 bit cases. The window length was set 250 at samples (about 10 cycles) and was then reduced to 50 samples (about 2 cycles). A graphical representation of the relative amplitudes at 12 bits is included in Figure 54. The deviation of each case from the ideal case is shown in Figure 55.



((a)) Relative Amplitude Comparison



((b)) Detail

Figure 54: Mixed Error Sources Relative Amplitude Comparison: 12 bits

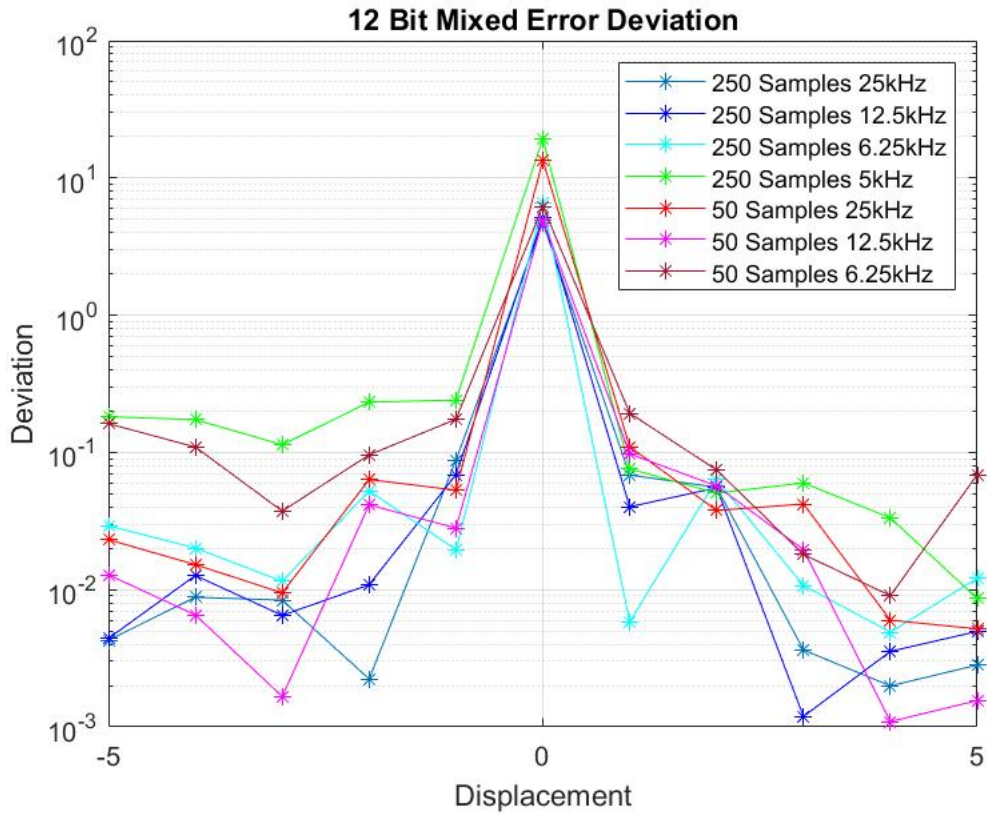


Figure 55: Mixed Error Sources Deviation: 12 bits

The 10 bit case exhibited much higher error than the 12 bit case when errors were mixed. The algorithm could only tolerate reduction of sample rate to 12.5 kHz when using 250 samples (about 10 cycles); at 50 samples (about 2 cycles) it begins to break down. A graphical representation of the relative amplitudes at 10 bits is included in Figure 56 and the deviation in Figure 57.

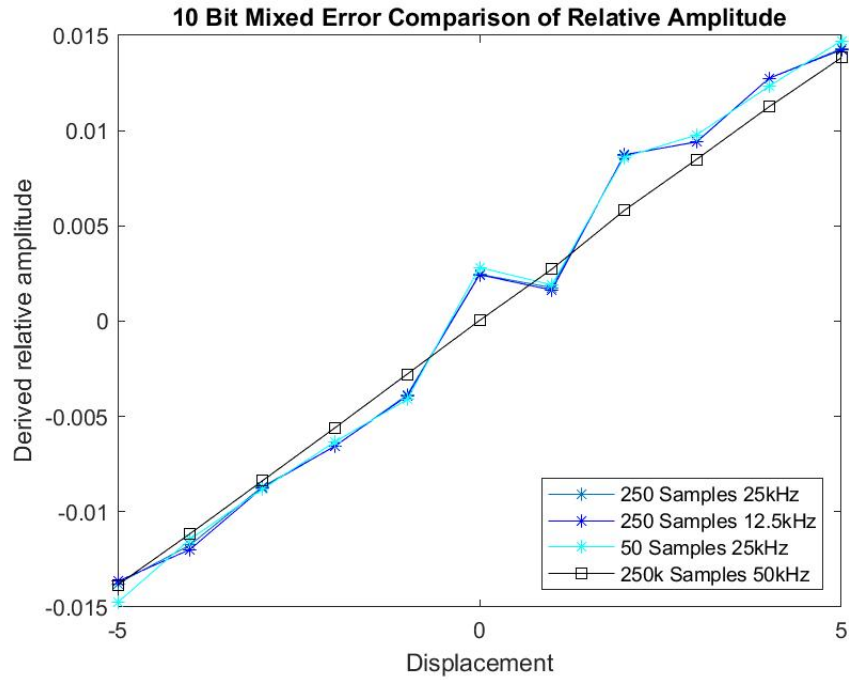


Figure 56: Mixed Error Sources Relative Amplitude Comparison: 10 bits

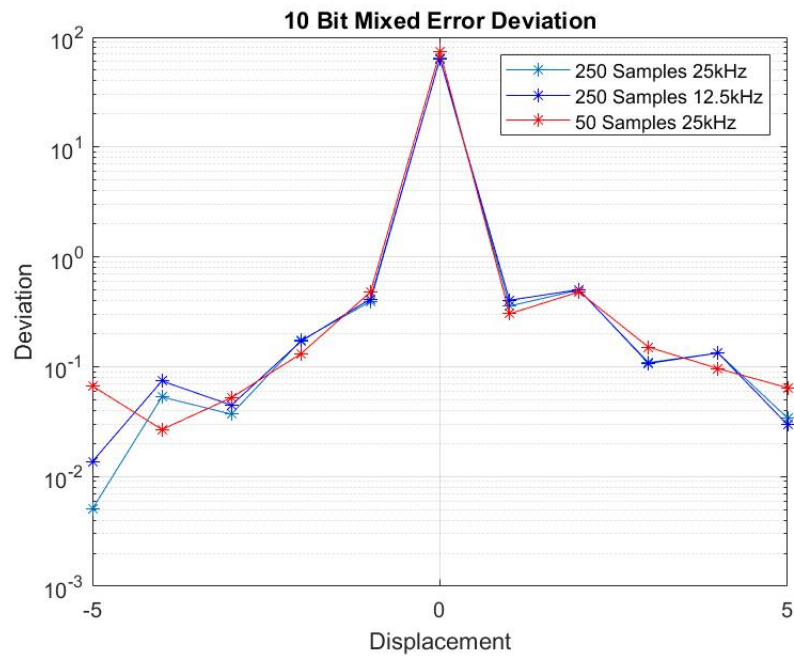


Figure 57: Mixed Error Sources Deviation: 10 bits

It is clear that the 12 bit case outperforms the 10 bit case in all circumstances. It is recommended that the window length be around 10 cycles, or 250 samples if the sample

rate is expected to decrease past one quarter of the original sample rate. In this case, since the original sample rate was 50 kHz, or around twenty times the frequency of the EPS, the minimum recommended sample rate would be 12.5 kHz if at a window length of 10 cycles.

4.6 REAL TIME CODE

The final step in the digitization is to convert the algorithm into as minimal a code as possible to allow for computation in real time. Currently, the NASA LaRC team is using a SpeedGoat real time target computer to process the EPS sensor information and allow for real time feedback control. While the SpeedGoat is not quite fast enough to process this algorithm in real time, the hardware to run this code quickly enough is imminent. The target execution time for the real time algorithm is 0.5 milliseconds. In the algorithm, an arbitrary sample size is specified by the user to allow for sequential processing where data points are added one by one and dropped off one by one. The original sample is 250,000 data points; the sample size used in the run in Figure 58 was 250 data points long. The mathematics in the code have been stripped down for the fastest possible processing time; no Matlab functions are included for this purpose. For 16 bits at 50 kHz and 250 samples, the processing time of the entire code can vary around 0.007 s.

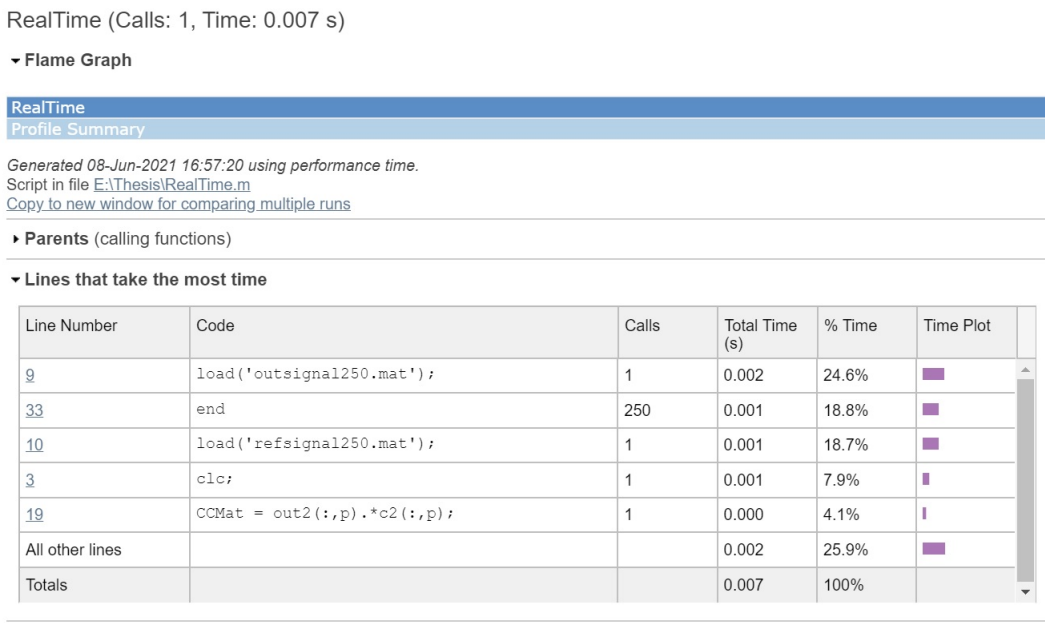


Figure 58: Time Analysis of Real Time Code

The majority of the run time was occupied by loading the digital data which is two lines of 2,750 samples; loading the data takes 0.003 s. Therefore, since loading data will not be necessary for real time processing, the actual total run time is 0.005 s.

The real time algorithm includes an indicator where the user can specify which displacement position point is desired for processing. For example, one run of the real time algorithm can be at the first displacement point and another run can be at a middle displacement point. Figure 59 is the real time code.

```

%% --- Real Time Computation
% For 250 point data cycle, one point at a time real-time processing
clc;
clear;

out2 = zeros(250, 11);
c2 = zeros(250, 11);

load('outsignal250.mat');
load('refsignal250.mat');

% Total number of points in signals loaded above
points=250;
% Position Indicator
p=1;

outMat = out2(:,p);
cMat = c2(:,p);
CCMat = out2(:,p).*c2(:,p);

outMatS = 0;
cMatS = 0;
CCMatS = 0;

%Point-by-point correlation sums
for u = 1:points
    %Pickup Auto-correlation
    outMatS = outMat(u)^2 + outMatS;
    %Reference Auto-correlation
    cMatS = cMat(u)^2 + cMatS;
    %Cross-correlation
    CCMatS = CCMat(u) + CCMatS;
end

%Auto- and Cross-correlations
ACout= (1/points)*outMatS;
ACc= (1/points)*cMatS;
CC= (1/points)*CCMatS;

%Amplitudes
Ac= sqrt(2*ACc);
Aout= sqrt(2*ACout);

relamp= Aout/Ac;
theta= acos(CC/sqrt(ACout*ACc));

```

Figure 59: Real Time Code

While the run time of 0.005 s is one order of magnitude larger than the target of 5 ms, the real time code was run on a regular home PC. The factor of ten is acceptable considering custom hardware will be purchased to allow for faster processing than was allowable on the home PC.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

A one degree of freedom model of the unique MIT-constructed EPS has been demonstrated along with its digital processing algorithm. The sensor physically performs as expected; when a magnetizable conductive model displaces axially through the model EPS amplitude and phase shift of a driving signal is evident. Theoretical data obtained from COMSOL AC/DC multiphysics software confirmed the operational status of the EPS. The conditioning circuitry receives input from the 2.36 kHz driving signal from a commercial off the shelf LVDT driving circuitboard and successfully adapts it to the scale of the model EPS. Finally, the digital processing algorithm has been demonstrated to extract amplitude and phase information. It has been proven to be stable when multiple error sources have been applied; the recommended minimum for reliable performance is 12 bits, a sample rate of 12.5 kHz, and a record length over 250 samples. It has also been demonstrated that the algorithm can be applied to a real time active feedback control mechanism when all functions are replaced by simple mathematics.

5.2 FUTURE WORK

This study has been a proof of concept of a one degree of freedom physical and digital model EPS. As such, much work remains to be completed by the NASA LaRC team. The one degree of freedom model must be expanded into six degrees of freedom to be useful to the MIT built EPS. This will be accomplished by expanding both the conditioning circuitry and the digital algorithm. Next, the conditioning circuitry must be scaled up to 20 kHz, the operating frequency of the MIT EPS. Finally, the real-time digital processing algorithm must be optimized for speed and performance to reject corruption from multiple error sources. The end result will be a fully digitized MSBS and EPS system with active feedback control.

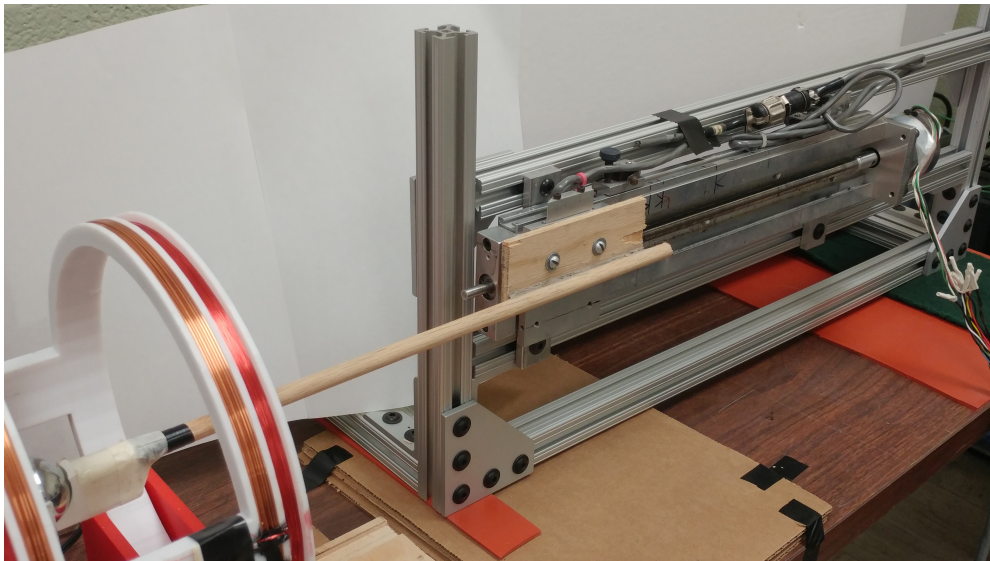
REFERENCES

- 1 R.P. Boyden, *A Review of Magnetic Suspension and Balance Systems*. (1988) 15th Aerodynamic Testing Conference, American Institute of Aeronautics and Astronautics.
- 2 R.P. Boyden and P. Tcheng, *Status of Magnetic Suspension Technology*. (1986) Langley Symposium on Aerodynamics, Vol. 1, pp. 261-277.
- 3 C. Britcher, *A Retrospective on the First Two Eras of Magnetic Suspension and Balance System Research*. (2015) Proceedings, International Conference on Flow Dynamics, Vol 12.
- 4 T. Schott, T. Jordan, and T. Daniels, *Present Status of the MIT/NASA Langley 6 inch MSBS*. (1992) Proceedings, International Symposium on Magnetic Suspension Technology, Part 2. NASA CP-3152-PT-2, pp. 741-764.
- 5 R.P. Boyden, R.A. Kilgore, P. Tcheng, and C.P. Britcher, *Super Magnets for Large Tunnels*. (1988) Aerospace America, Vol. 26, pp. 36-38,40.
- 6 W.R. Towler, *Electromagnetic Position Sensor for a Magnetically Supported Model in a Wind Tunnel*. (1971) Proceedings, 2nd International Symposium on Electro-Magnetic Suspension, pp. Q4-Q7.
- 7 T. Stephens, *Design, Construction, and Evaluation of a Magnetic Suspension and Balance System for Wind Tunnels*. (1969) NASA CR-66903, MIT-TR-136, Chapter IV & Appendix B, pp. 61-79, 143-157.
- 8 T.S. Daniels and J.S. Tripp, *Improvements to an Electromagnetic Position Sensor for a Magnetic Suspension Tunnel*. (1988) Proceedings, 34th International Instrumentation Symposium, Instrument Society of America, pp. 65-70.
- 9 C. Britcher, M. Schoenenberger, *Feasibility of Dynamic Stability Measurements of Planetary Entry Capsules Using MSBS*. (2015) International Conference on Flow Dynamics, Vol 12.
- 10 C. Britcher, M. Weinmann, T. Schott, and M. Schoenenberger, *Further Development of an Electromagnetic Position Sensor for a Wind Tunnel MSBS*. (2018) Fifteenth International Conference on Flow Dynamics.
- 11 S.W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*. (1997) California Technical Publishing, USA.
- 12 Sun, Xianwu, et al. *New Phase Difference Measurement Method for Non-Integer Number of Signal Periods Based on Multiple Cross-Correlations*. The Open Automation and Control Systems Journal, vol. 7, no. 1, 2015, pp. 1537–1541., doi:10.2174/1874444301507011537.
- 13 N. Sevastopoulos and R. Markell, *A Simple Method of Designing Multiple Order All Pole Bandpass Filters by Cascading 2nd Order Sections, Application Note 27A*. (1988) Linear Technology Corporation.
- 14 *TB6600 Stepper Motor Driver User Guide*. DFRobot.com. Version V1.2.

APPENDIX A - Core Displacement System

To ensure that the core displaces in accurate increments a system was constructed to move the core 1 mm at a time using a stepper motor programmed with LabView. An aluminum structure supports the stepper motor which is attached to a wooden pole terminating in a 3D printed cradle for a 1 inch diameter steel ball; all materials used were chosen for their low electrical and magnetic conductivity.

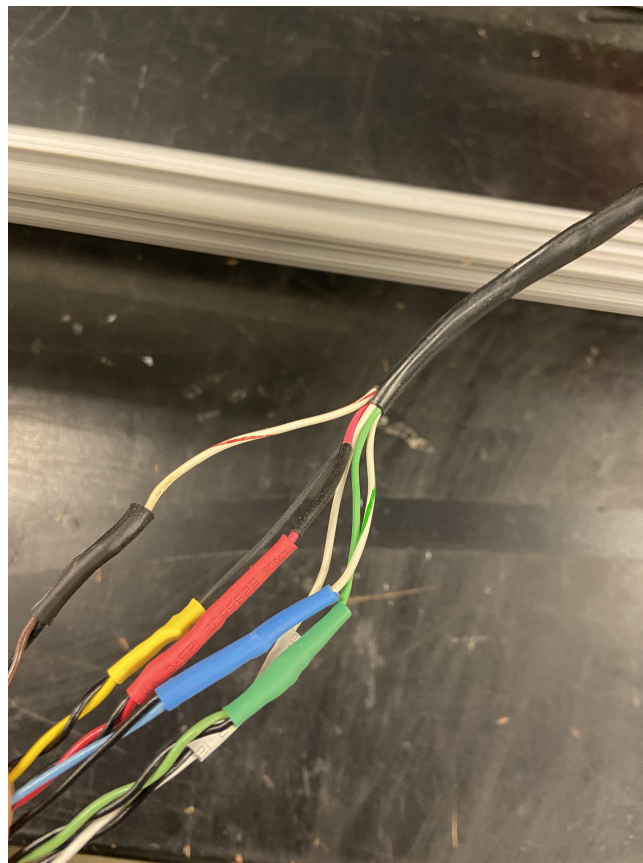
A-1: Photo of Core Displacement System



The components used for the core displacement system were a Superior Electric Slo-Syn M061-LS08 step motor, a Mean Well PS-65-R16VAI power supply, and a TB6600 Stepper Motor Driver microcontroller. The Slo-Syn motor requires 3.8A at 1.25V DC to move 200 steps per revolution. A step motor converts digital signal pulses into shaft rotation providing displacement; in order to send a digital signal pulse to the motor the driver and microcontroller must be wired into a LabView controller, a Measurement Computing USB-1208LS PMD. The LabView code then outputs a square wave at a user-specified frequency, each signal pulse causes a step corresponding to 1.8 degrees of rotation.

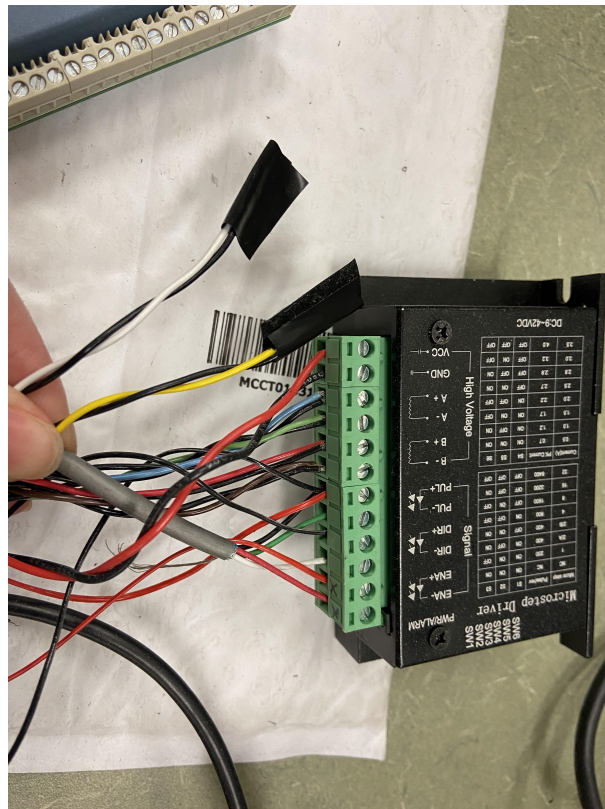
Power was supplied to the Slo-Syn motor using a Mean Well PS-65-R16VAI power supply and control is provided by the TB6600 microcontroller [14]. There are six power lead wires from the Slo-Syn motor which are color coded: red solid, red striped, green solid, green striped, white, and black. The motor is composed of two motor coils, the A and B coils, each of which has two corresponding wires. The red solid and striped wires correspond to the A+ and A- ends of the A coil, the green solid and striped wires correspond to the B+ and B- ends of the B coil. These four wires, in addition to the white and black wires, are connected to a cable with six color coded wires each with a black ground reference. The connections are made as follows: red solid to red, red striped to brown, green solid to green, green striped to blue, white to white, and black to yellow.

A-2: Photo of Motor Power Lead Wires



The microcontroller only requires four wires for operation, namely the wires corresponding to each end of both the A and B coils. Hence, a 2-phase 6-wire motor wiring was used with the yellow and white wires excluded.

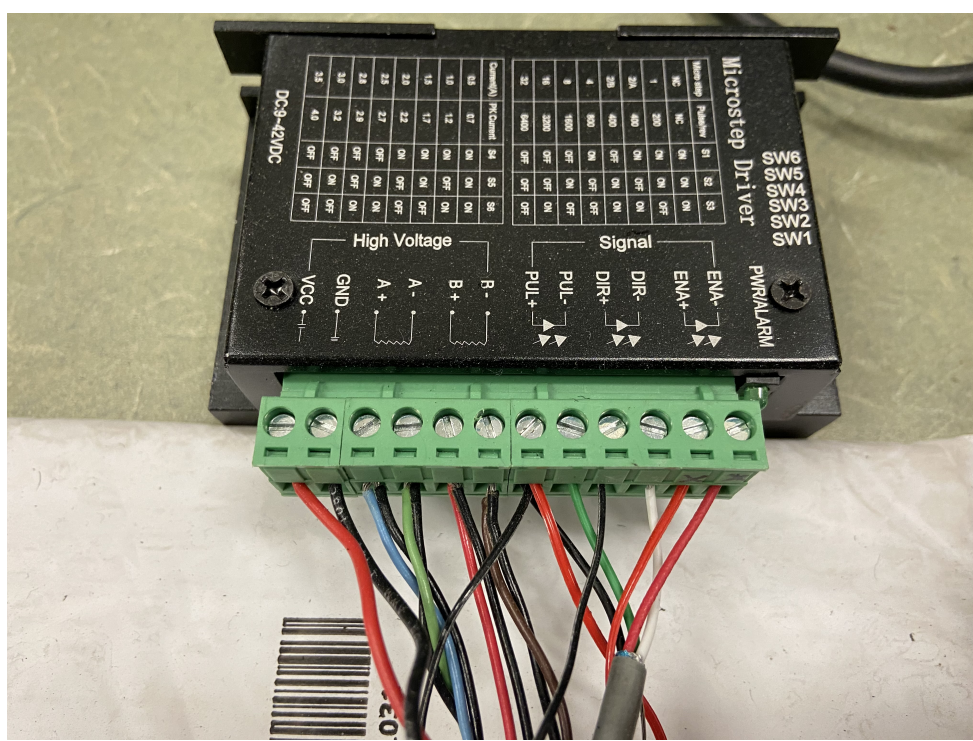
A-3: Photo of Motor Driver Wiring



The common-anode connection was used to connect the driver to the microcontroller excluding the EN terminal which allows for off-line function. The pulse and direction information are sent from a LabView program to the MC PMD. The common-anode connection entails utilizing the pulse wire from pin 21 of the PMD as the pulse minus terminal on the motor driver. The direction wire from pin 22 connects with the direction minus terminal on the driver. The ground wire from the PMD is wired to the pulse plus terminal on the driver. The red wire which is unconnected to the PMD is wired to the enable minus terminal on the driver. Two more connections are necessary: a red wire connects the enable plus terminal to

the pulse plus terminal, and a black wire connects the pulse plus terminal to the direction plus terminal.

A-4: Photo of Motor Driver Wiring

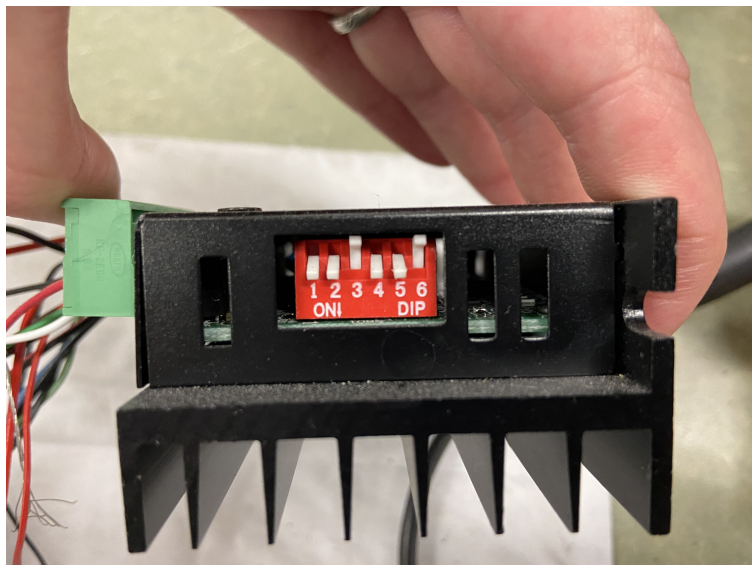


A-5: Photo of LabView PMD



Additionally the motor driver has six DIP switches; the first three signify the micro step and pulse per revolution and the last three signify the current. The switch settings selected were S1 off, S2 off, and S3 on to signify 16 micro steps and 3200 pulses per revolution.

A-5: Photo of Motor Driver DIP Switches



Since the Slo-Syn motor has a 1.8 degree step angle, the final step angle would be $1.8/16 = 0.1125$ degrees. The other three switches were selected to be S4 on, S5 off, and S6 on to signify current control at 1A.

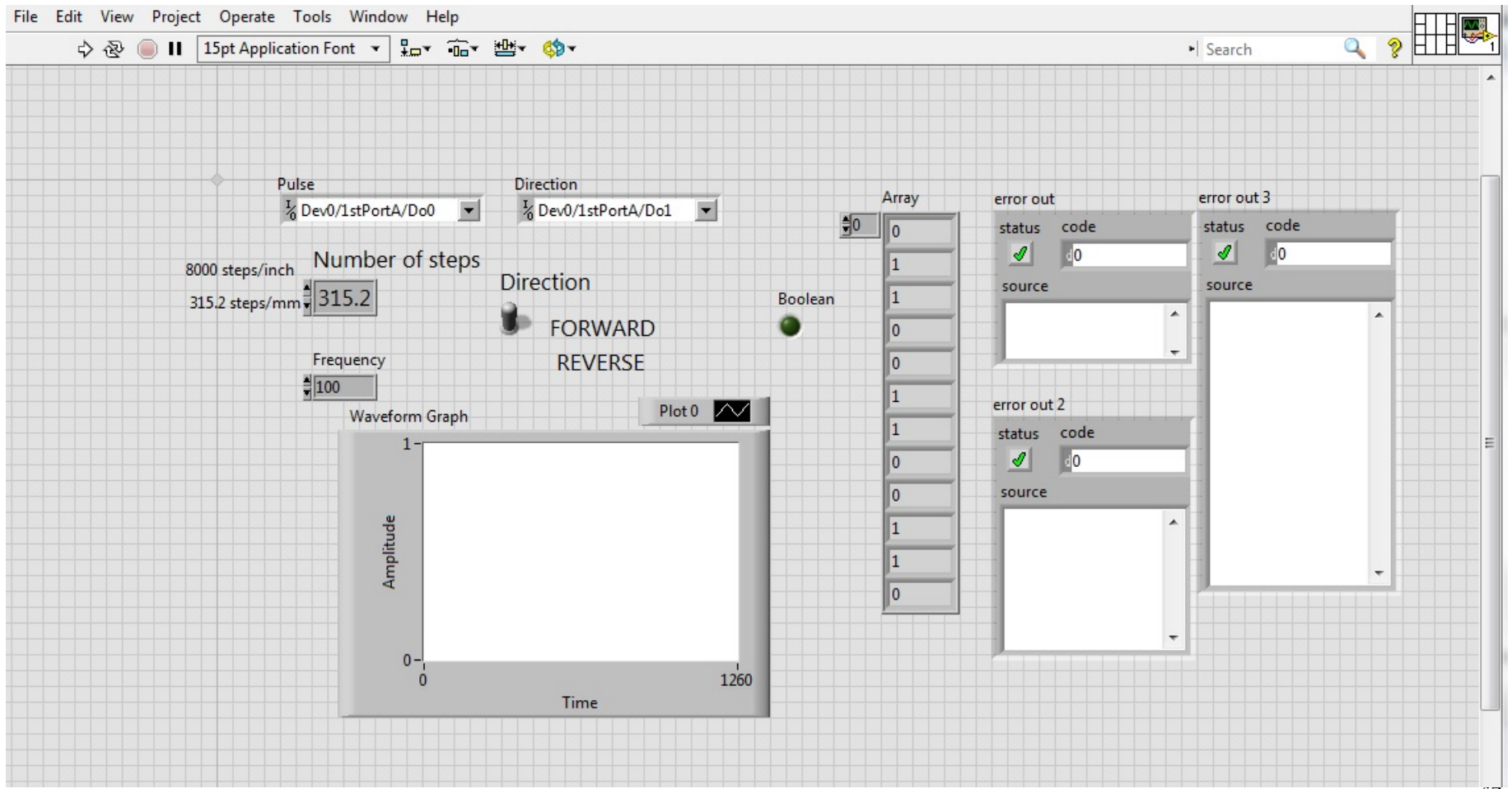
It is essential for proper operation of the microcontroller that the data acquisition device be powered on and the LabView program be open before power is supplied to the driver. The clock from the LabView program must be on and sending a signal to the driver before the power can be turned on. As mentioned previously, there are two pins of the PMD which supply relevant information to the microcontroller from LabView: pins 21 and 22. Pin 21 sends the proper pulse information from LabView into the pulse plus terminal of the microcontroller. Pin 22 sends the proper direction information from LabView into the direction plus terminal of the microcontroller. Once the pulse and direction information has been supplied by LabView, the information was sent into the microcontroller to power the step motor. The degrees of rotation were converted into displacement, the desired displacement of 1 mm at a time was achieved. For details about the LabView program, please refer to Appendix C-2.

APPENDIX B - Measurement Details

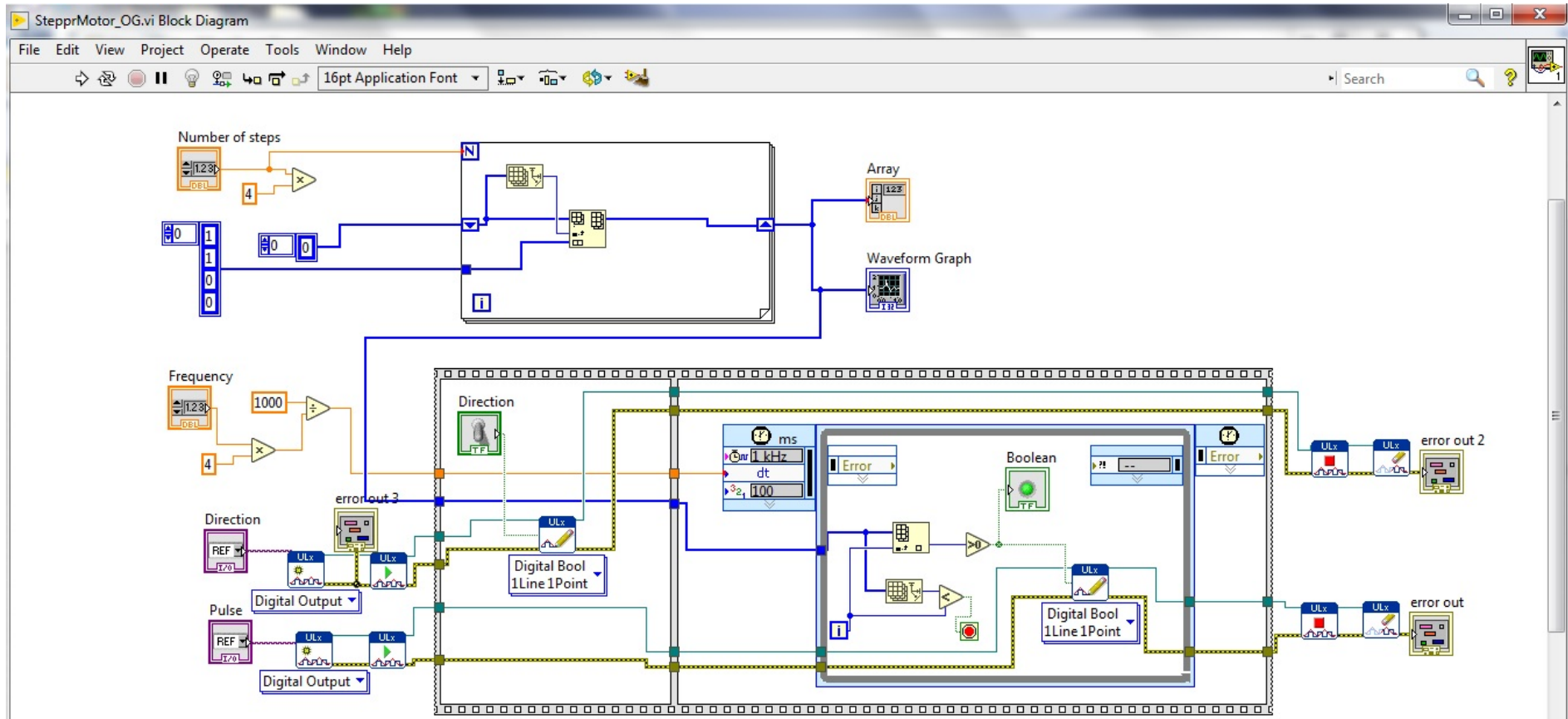
To measure the voltage output of the model EPS, the signal on the pickup coils was obtained using an oscilloscope with precise voltage measurement capabilities; since the driving frequency is 2.36 kHz a voltmeter could not be used due to its maximum frequency allowance of 1 kHz. The voltage on each coil was measured to be about 8.6 V, therefore a change on the mV scale would not be noticeable on the oscilloscope. The voltage measurements were taken for each coil set including both coils wound in series; the steel sphere was placed first at one end of the model EPS and then the other. With the sphere at the top of the model EPS, the first coil set measured 52 mV and the second set measured 32 mV, totalling 84 mV. With the sphere at the bottom of the model EPS, the first coil set measured 28 mV and the second set measured 60 mV, totalling 88 mV.

APPENDIX C - LabView Codes

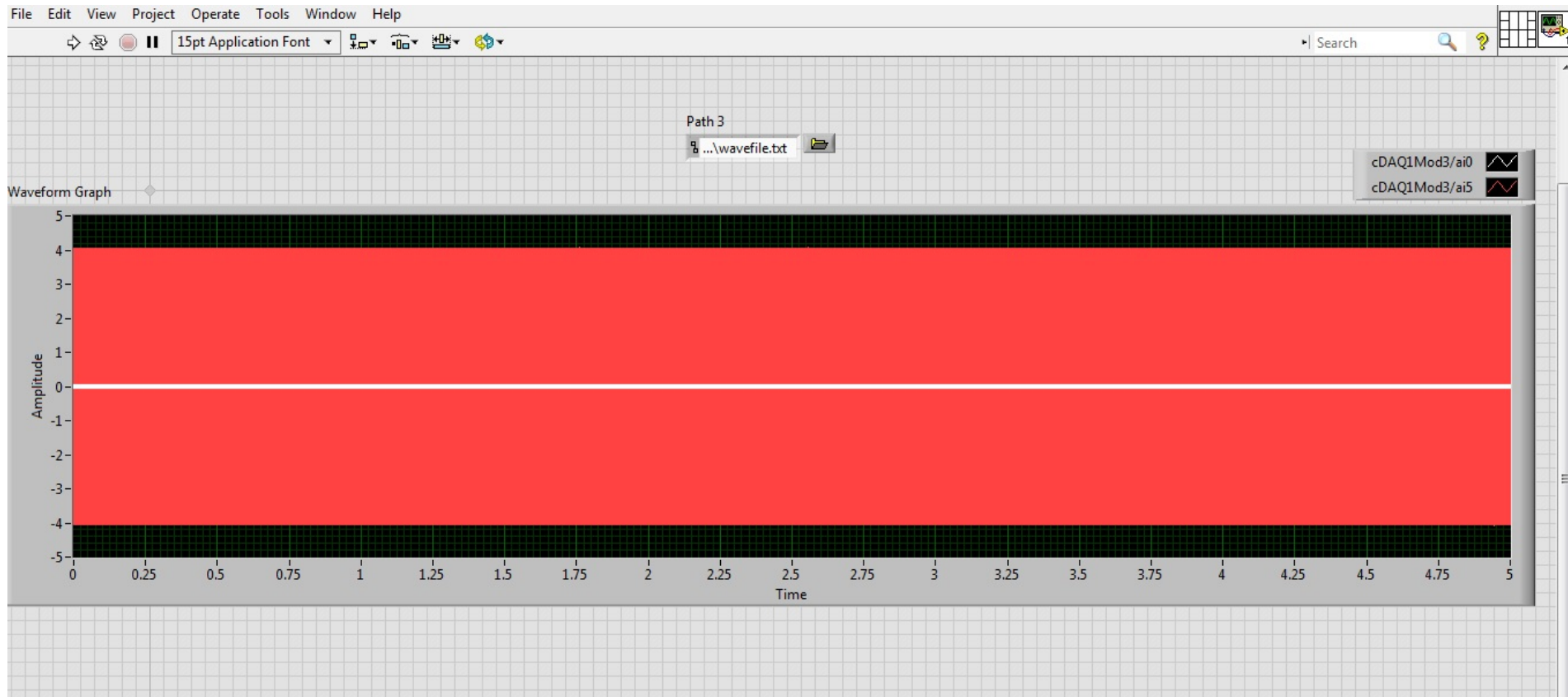
C-1: LabView Step Motor Driver Control Panel



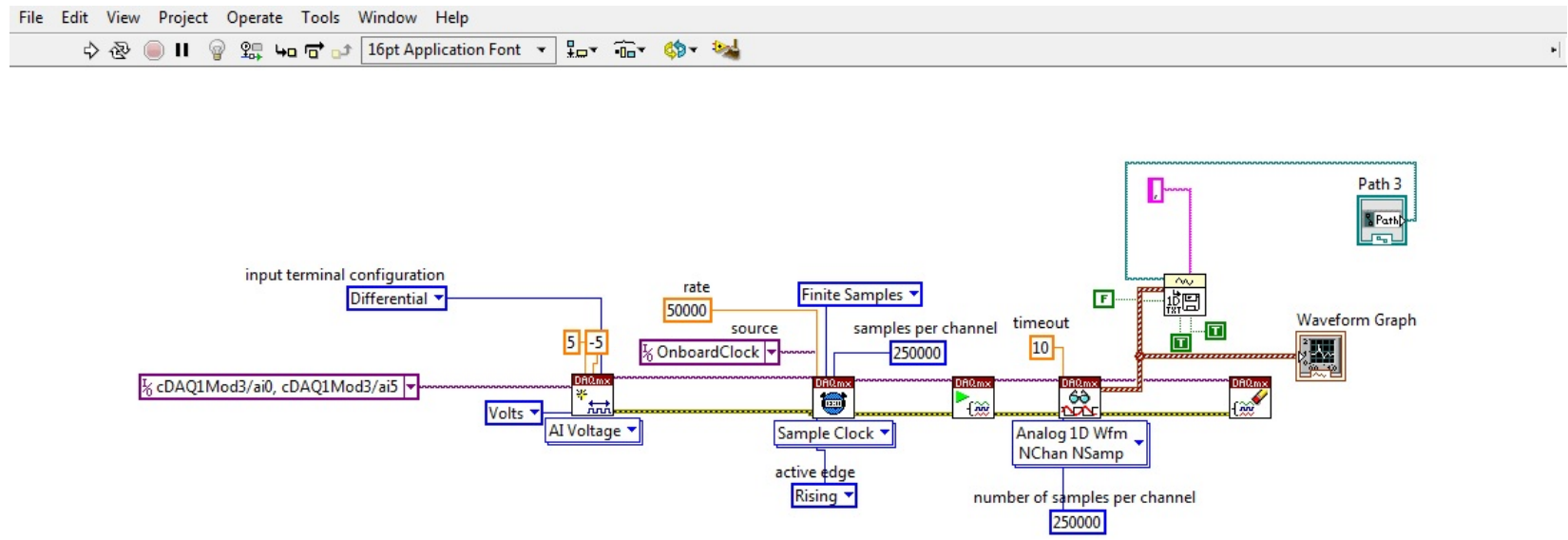
C-2: LabView Step Motor Driver Block Diagram



C-3: LabView Data Collection Block Diagram



C-4: LabView Data Collection Block Diagram



APPENDIX D - Algorithm MATLAB Code

D-1: MATLAB Simulation Scripts

Contents

- [Script for cross-correlation based EPS demodulation](#)
- [Constants](#)
- [Main Code](#)
- [To plot first displacement point reference and pickup signals](#)
- [Analysis plots](#)

Script for cross-correlation based EPS demodulation

SIMULATED DATA: script creates continuous signals that scales from -1 to 0 to 1 in amplitude. The phase shift is pre-programmed, specified in line 18. Cross-correlation analysis extracts relative amplitude and phase information.

```
clc;
clear;
close all;
```

Constants

```
EPSfreq=2.36e3;           % EPS carrier
N=10;
rate=(EPSfreq*N);        % Sample rate of real system
dt=1/rate;
dtheta=2*pi/N;
numcycles=10;           % Record length (default: 10)
pts=N*numcycles;        % Sample number (acts as sample rate)
phase=50;
phase=phase*pi/180;
numPoints = 41;         % Displacement points
```

Main Code

```
% Initialized Matrices
[c,noise,pci,t,tm,out]=deal(zeros(pts,numPoints));
% Position Loop
for j=1:numPoints
    %scale ranges from -1(@startpoint) to 0(@midpoint) to 1(@endpoint)
    scale(j)=(-21+j)*0.05;
    % Noise Creation Loop
    for i=1:pts
        noise(i,j)=(-0.5+rand);
        % current ripple noise 2.55=51/20 ratio from MSBS/EPS power supply at LaRC
        pci(i,j)=sin(i*2.55*dtheta);
    end

    % Signal Creation
    for i=1:pts
        t(i,j)=i*dt;
        %time in microseconds
        tm(i,j)=t(i)*1e6;
        %simulated reference signal
        c(i,j)=sin(i*dtheta);
        %simulated pickup signal, add noise: +0.1*noise(i)+0.5*pci(i)
        out(i,j)=scale(j)*sin(i*dtheta-phase);
```

```

    %When +phase, for first half out lags c, then out leads c. When
    %-phase, it's the opposite.
end
% Plotting Signals
figure
plot(tm(:,j),c(:,j),tm(:,j),out(:,j));
axis([0 2000 -1.2 1.2]);
title('Simulated Signals');
legend('reference','pickup');

%Autocorrelation of reference
Rx(j) = (1/pts)*sum(c(:,j).^2);
%Autocorrelation of measured
Ry(j) = (1/pts)*sum(out(:,j).^2);

%Amplitudes
Ax(j) = sqrt(2*Rx(j));
Ay(j) = sqrt(2*Ry(j));

%Crosscorrelation
R(j) = (1/pts)*sum(c(:,j).*out(:,j));

relphase(j)=acos(R(j)/sqrt(Rx(j)*Ry(j)));
if isnan(relphase(j)) == 1
    relphase(j) = 0;
end

relamp(j) = Ay(j)/Ax(j);
end
%Relative phase sign adjust loop: since acos only outputs angles from 0 to 180: last four
%values (all less than value at position 7) should be in Q4, not Q1
for k=1:41
    if k<21
        relphase(k) = 2*pi - relphase(k);
    end
    relphase(k) = relphase(k)*180/pi;
end
%Relative amplitude sign correction loop: negate first 5 values
for k=1:20
    relamp(k) = -relamp(k);
end
end

```

To plot first displacement point reference and pickup signals

```

figure
plot(tm(:,1),c(:,1),tm(:,1),out(:,1));
axis([0 2000 -12 12]);
legend('reference','pickup');
figure
plot(tm(:,21),c(:,21),tm(:,21),out(:,21));
axis([0 2000 -12 12]);
legend('reference','pickup');

```

Analysis plots

```

figure
subplot(2,2,1);
hold on;

```

```

plot(R);
plot(Rx);
plot(Ry);
title('Cross Correlation Sum');
xlabel('Displacement');
ylabel('Correlated Amplitude');
legend('R','X','Y');
grid on;
subplot(2,2,2);
hold on;
plot(Ax);
plot(Ay);
title('Amplitude');
xlabel('Displacement');
ylabel('Derived Amplitude');
legend('Ax','Ay');
grid on;
subplot(2,2,3);
figure
plot(scale,relphase);
title('Relative Phase');
xlabel('Amplitude');
ylabel('Phase Difference (Deg)');
grid on;
subplot(2,2,4);
figure
plot(scale,relamp);
ylim([-1 1]);
grid on;
title('Relative Amplitude');
xlabel('Actual Amplitude');
ylabel('Derived Amplitude');

```

Contents

- [Script for cross-correlation based EPS demodulation: Including all error sources](#)
- [Constants](#)
- [Main Code](#)
- [Downsampling - decimate\(x,r\) divides the sample rate by r](#)
- [Quantization](#)
- [Correlation - DOWNSAMPLING ONLY](#)
- [Correlation - QUANTIZATION ONLY](#)
- [Correlation - BOTH QUANTIZATION AND DOWNSAMPLING](#)
- [Signal Plots](#)
- [Analysis Plots](#)

Script for cross-correlation based EPS demodulation: Including all error sources

SIMULATED DATA To run with no error, dn = 1 There are four sources of error: Downsampling, quantization, noise, and window length. To change the downsampling level, change "dn", line 29. To change quantization level, change "bits", line 62. To change noise, add +n*noise(i)+m*pci(j), n and m < 1, to the output signal, line 47. To change the window length, change "numcycles", line 23. There are three options for processing: using only downsampling, line 76, using only quantization, line 86, or both, line 96. Comment out the others, use only one choice.

```
clc;
clear;
close all;
```

Constants

```
EPSfreq=2.36e3;           % EPS carrier
N=10;
rate=(EPSfreq*N);        % Sample rate of real system
dt=1/rate;
dtheta=2*pi/N;
numcycles=10;           % Record length
points=N*numcycles;      % Sample number (acts as sample rate)
phase=50;
phase=phase*pi/180;
numPoints = 41;         % Number of displacement data points

dn = 1;                  % Downsampling value - fails at 5 b/c violates Nyquist
dpoints = points/dn;     % Number of samples once downsampled
```

Main Code

Initialized Matrices

```
[c,noise,pci,t,tm,out]=deal(zeros(points,numPoints));
% Position Loop
for j=1:numPoints
```

```
    %scale ranges from -1(@startpoint) to 0(@midpoint) to 1(@endpoint)
    scale(j)=(-21+j)*0.05;
```



```

% Noise Creation Loop
for i=1:points
    noise(i,j)=(-0.5+rand);
    % current ripple noise 2.55=51/20 ratio of power supply ripple/operating frequency
    pci(i,j)=sin(i*2.55*dtheta);
end
% Signal Creation
for i=1:points
    t(i,j)=i*dt;
    %time in microseconds
    tm(i,j)=t(i)*1e6;
    %simulated reference signal
    c(i,j)=sin(i*dtheta);
    %simulated pickup signal, to add noise add +0.1*noise(i)+0.1*pci(i)
    out(i,j)=scale(j)*sin(i*dtheta-phase);
    %When +phase, for first half out lags c, then out leads c. When
    %-phase, it's the opposite.
end

```

Downsampling - decimate(x,r) divides the sample rate by r

```

DSO(:,j) = decimate(out(:,j),dn);
DSC(:,j) = decimate(c(:,j),dn);

```

Quantization

Use sig = out(:,j) (line 66) if quantizing signals with original sample rate. Use sig = DSO(:,j) (line 67) if quantizing signals with downsampled rate.

```

%   sig = out(:,j);
sig = DSO(:,j);
ampMax = 10; % max(sig)
bits = 10;
nLevels = 2^bits; % Assuming that num levels is total for entire wave
partition = linspace(-ampMax,ampMax,nLevels); % Length numPoints-1, to represent numPoint intervals
spaceW = 2*ampMax/nLevels;
codebook = [-ampMax-spaceW, partition]; % Length numPoints, one entry for each interval
[index,qTemp] = quantiz(sig,partition,codebook); % Quantize
quants(:,j) = qTemp';

```

Correlation - DOWNSAMPLING ONLY

```

Rx(j) = (1/dpoints)*sum(DSC(:,j).^2);
Ry(j) = (1/dpoints)*sum(DSO(:,j).^2);

Ax(j) = sqrt(2*Rx(j));
Ay(j) = sqrt(2*Ry(j));

R(j) = (1/dpoints)*sum(DSC(:,j).*DSO(:,j));

```

Correlation - QUANTIZATION ONLY

```

%   Rx(j) = (1/dpoints)*sum(c(:,j).^2);
%   Ry(j) = (1/dpoints)*sum(quants(:,j).^2);
%
%   Ax(j) = sqrt(2*Rx(j));

```

```

%   Ay(j) = sqrt(2*Ry(j));
%
%   R(j) = (1/dpoints)*sum(c(:,j).*quants(:,j));

```

Correlation - BOTH QUANTIZATION AND DOWNSAMPLING

```

%   Rx(j) = (1/dpoints)*sum(DSc(:,j).^2);
%   Ry(j) = (1/dpoints)*sum(quants(:,j).^2);
%
%   Ax(j) = sqrt(2*Rx(j));
%   Ay(j) = sqrt(2*Ry(j));
%
%   R(j) = (1/dpoints)*sum(DSc(:,j).*quants(:,j));

relphase(j)=acos(R(j)/sqrt(Rx(j)*Ry(j)));

if isnan(relphase(j)) == 1
    relphase(j) = 0;
end

relamp(j) = Ay(j)/Ax(j);

```

```

end
%Relative phase sign adjust loop: since acos only outputs angles from 0 to
%180 first 20 values should be in Q4, not Q1
for k=1:41
    if k<21
        relphase(k) = 2*pi - relphase(k);
    end
    relphase(k) = relphase(k)*180/pi;
end
%Relative amplitude sign correction loop: negate first 20 values
for k=1:20
    relamp(k) = -relamp(k);
end

```

Signal Plots

```

figure
plot(tm(:,1),c(:,1),'-*',tm(:,1),out(:,1),'-*');
axis([0 4300 -1.2 1.2]);
legend('reference','pickup');
title('Original Signal, first data point');

x = (1:dn:100)*dt*1e6;
figure
plot(x,DSc(:,1),'-*',x,DSo(:,1),'-*')
axis([0 4300 -1.2 1.2]);
legend('reference','pickup')
title('Downsampled Signal, first data point');

```

Analysis Plots

```

figure
subplot(2,2,1);
hold on;

```

```
plot(R);
plot(Rx);
plot(Ry);
title('Cross Correlation Sum');
xlabel('Displacement');
ylabel('Correlated Amplitude');
legend('R','X','Y');
grid on;
subplot(2,2,2);
hold on;
plot(Ax);
plot(Ay);
title('Amplitude');
xlabel('Displacement');
ylabel('Derived Amplitude');
legend('Ax','Ay');
grid on;
subplot(2,2,3);
plot(scale,relphase);
title('Relative Phase');
xlabel('Amplitude');
ylabel('Phase Difference (Deg)');
grid on;
subplot(2,2,4);
plot(scale,relamp);
ylim([-1 1]);
grid on;
title('Relative Amplitude');
xlabel('Actual Amplitude');
ylabel('Derived Amplitude');
```

D-2: Simulated Signal Graphs

41 displacement points: Time (ms) vs. Amplitude (Voltage)

Blue signal: Reference or excitation signal. Red signal: Measured or pickup signal.

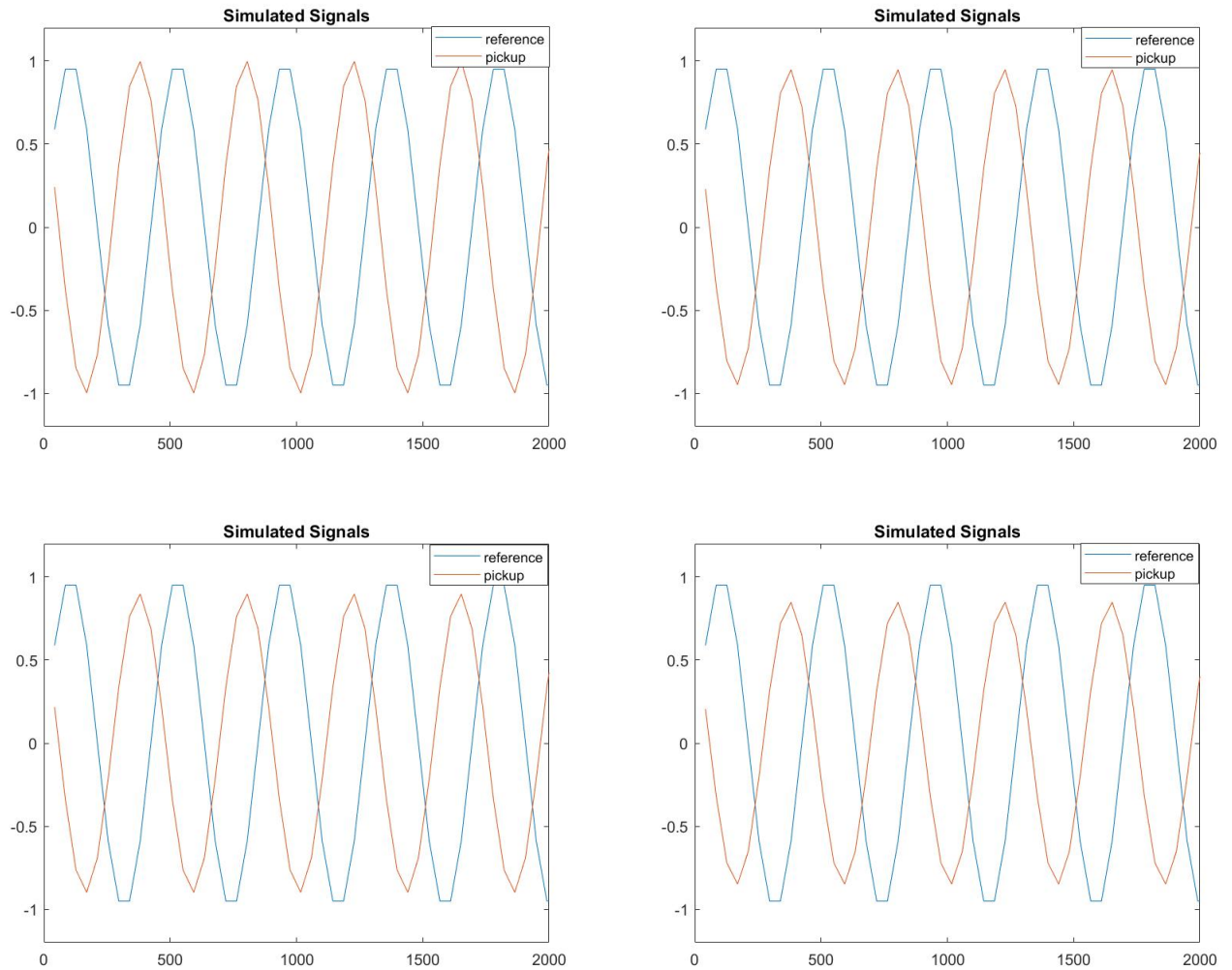


Figure 60: Simulated Signals, Displacement Points 1-4

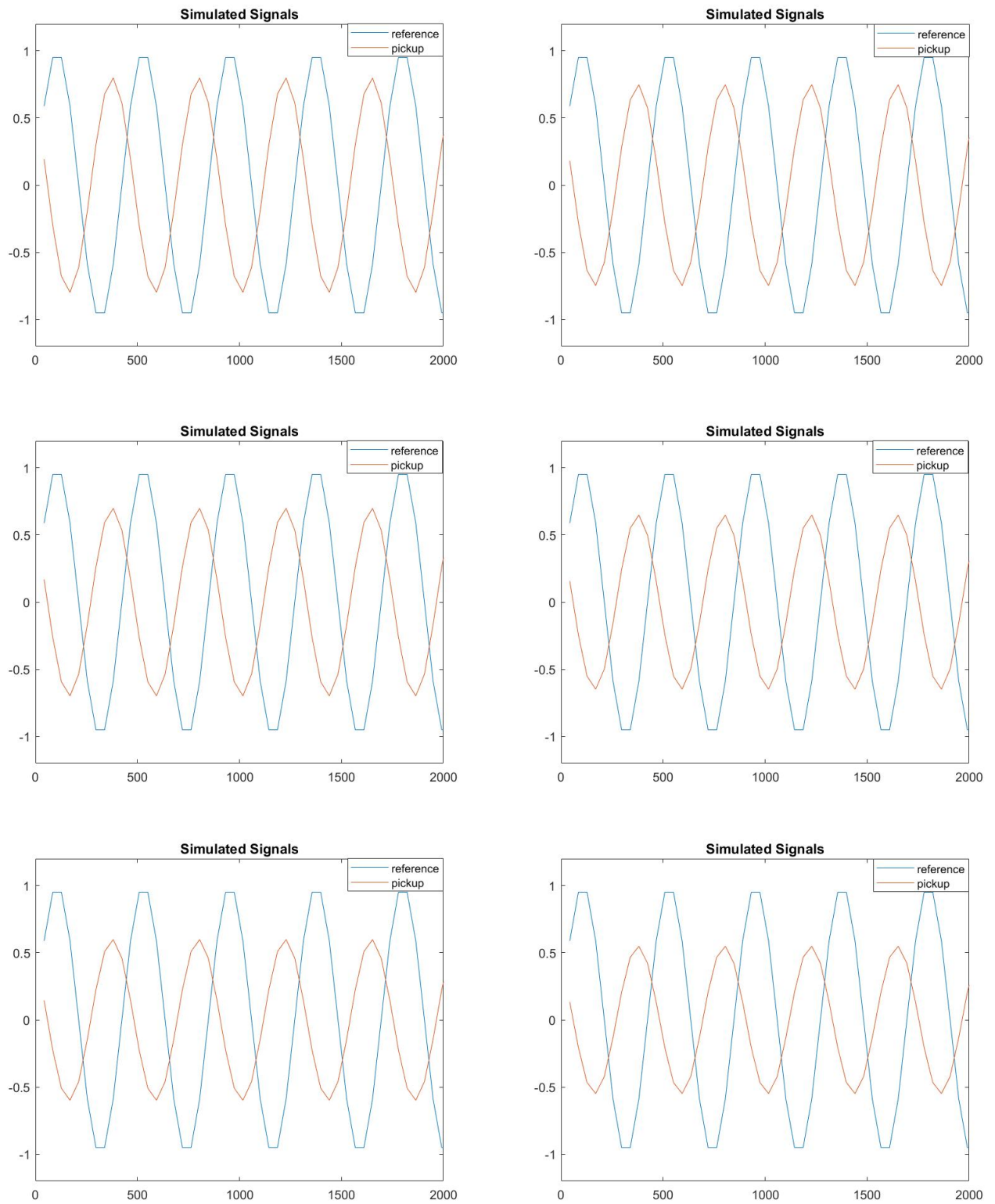


Figure 60: Simulated Signals, Displacement Points 5-10

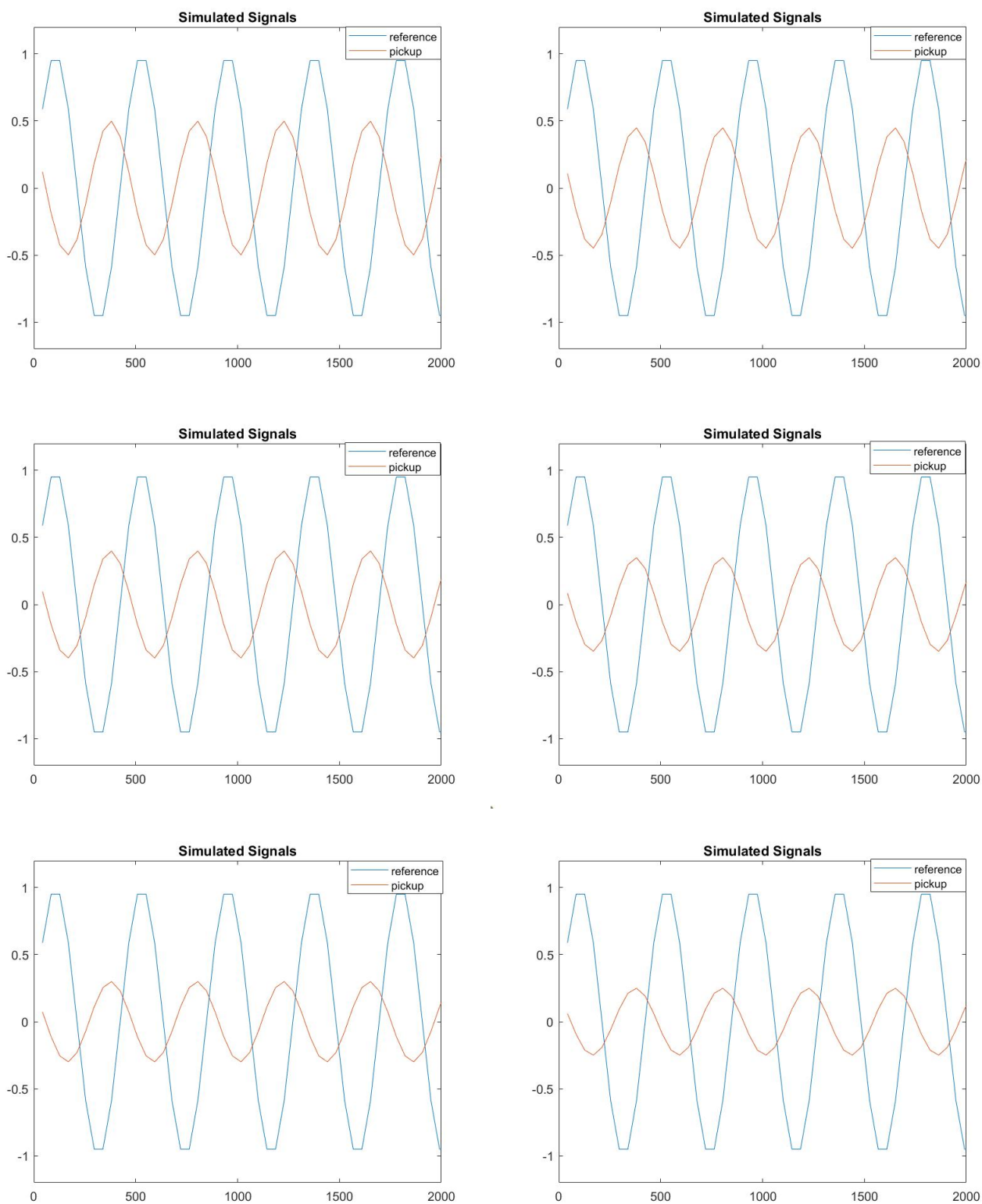


Figure 60: Simulated Signals, Displacement Points 11-16

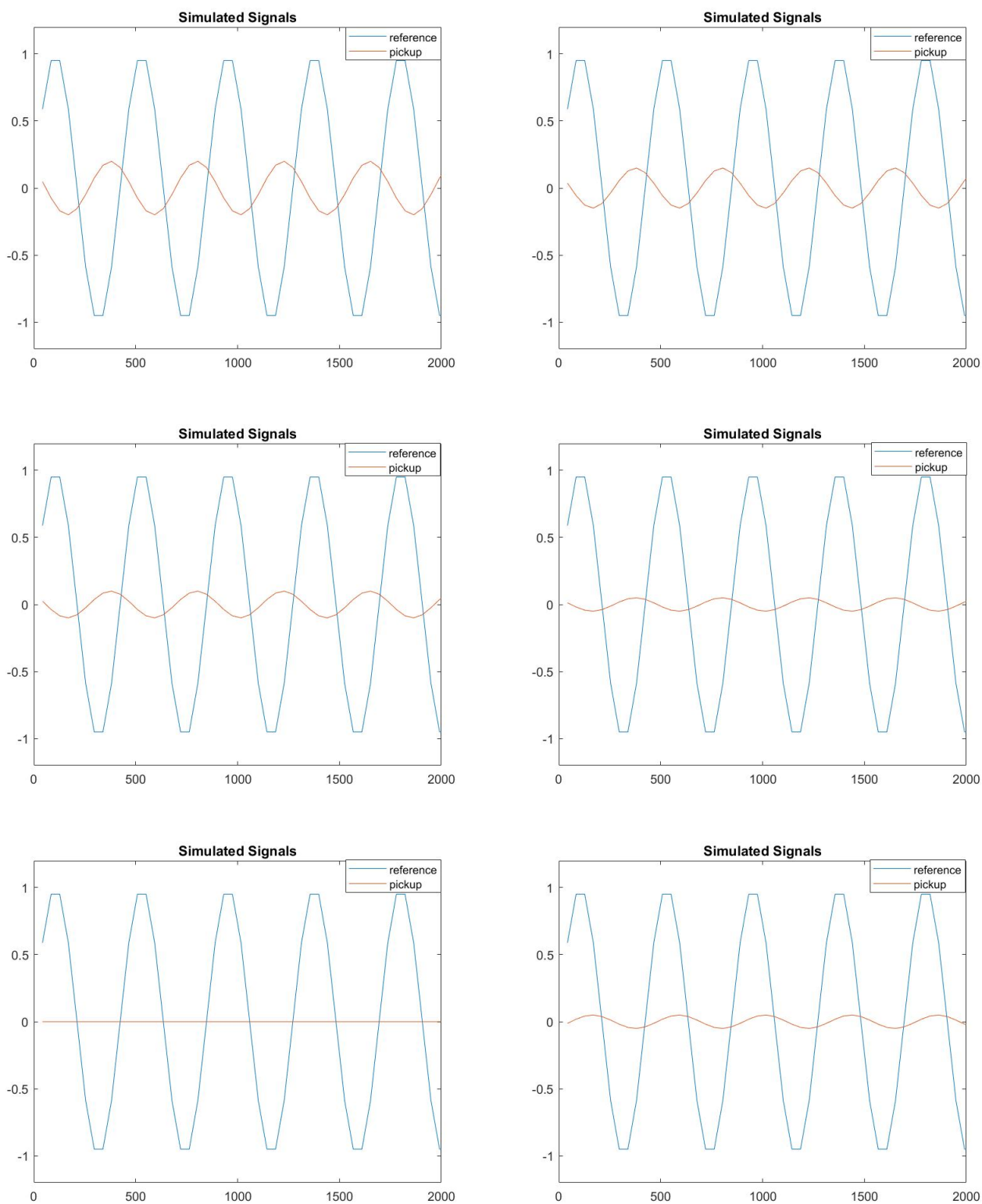


Figure 60: Simulated Signals, Displacement Points 17-22

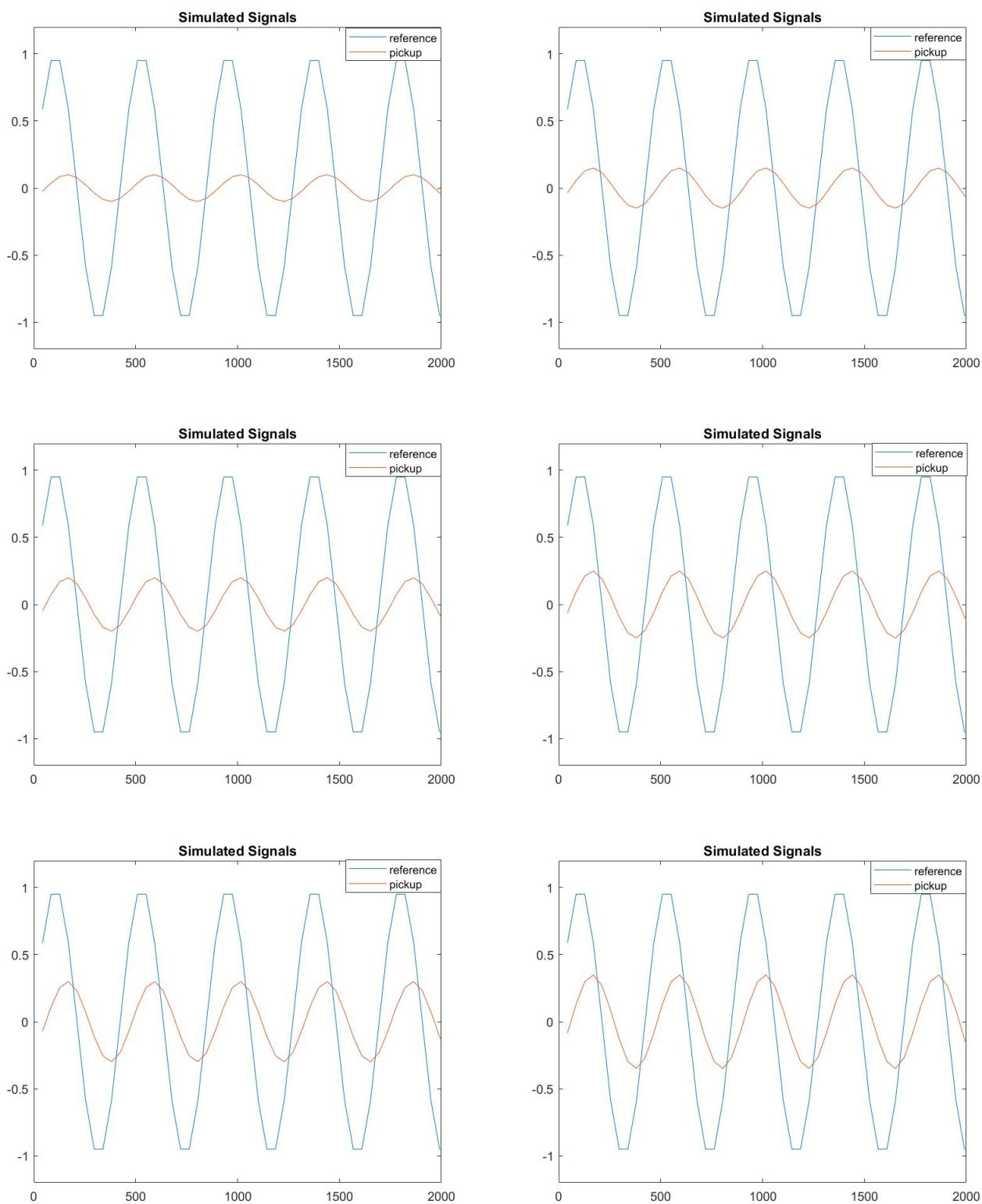


Figure 60: Simulated Signals, Displacement Points 23-28

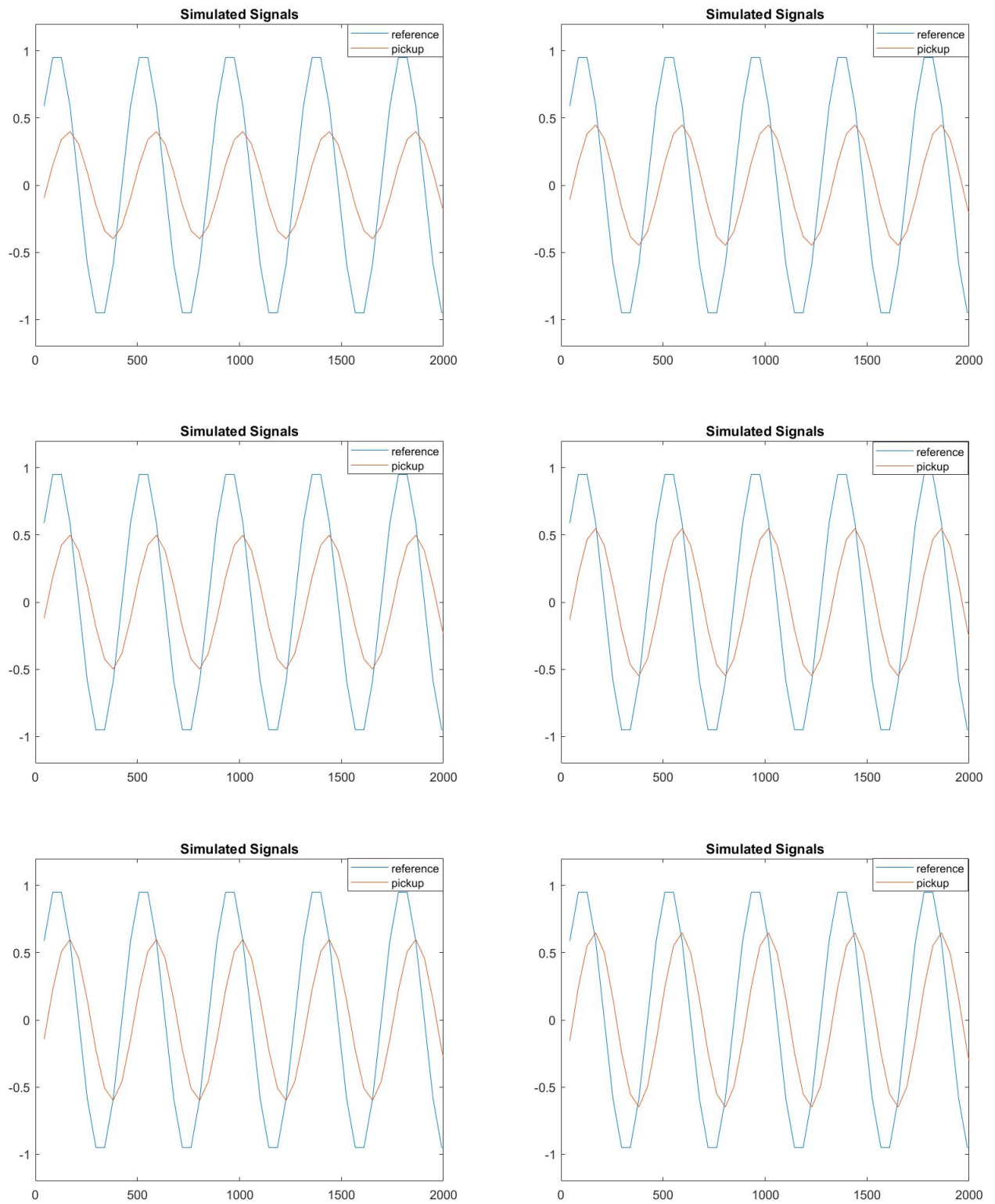


Figure 60: Simulated Signals, Displacement Points 29-34

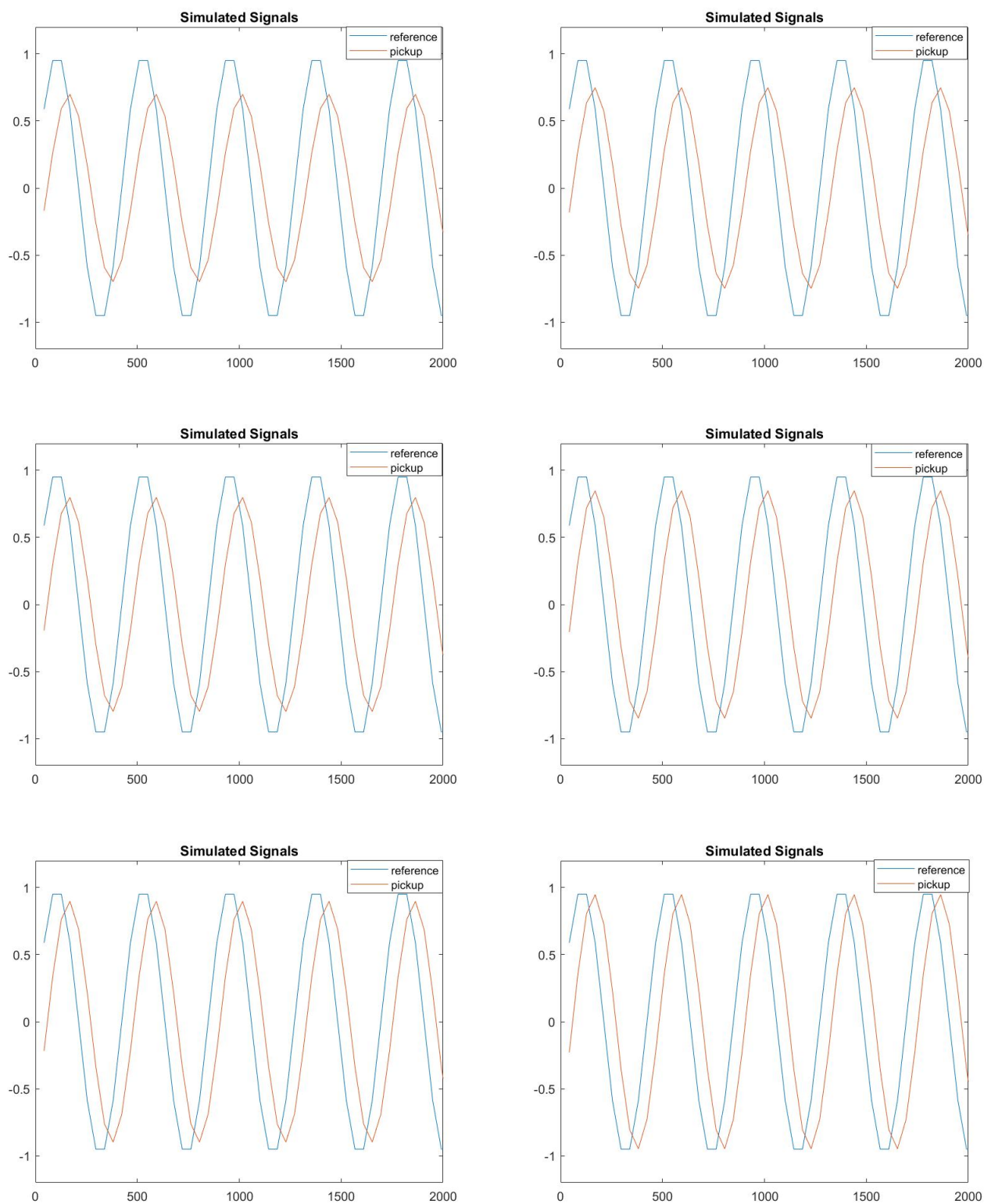


Figure 60: Simulated Signals, Displacement Points 35-40

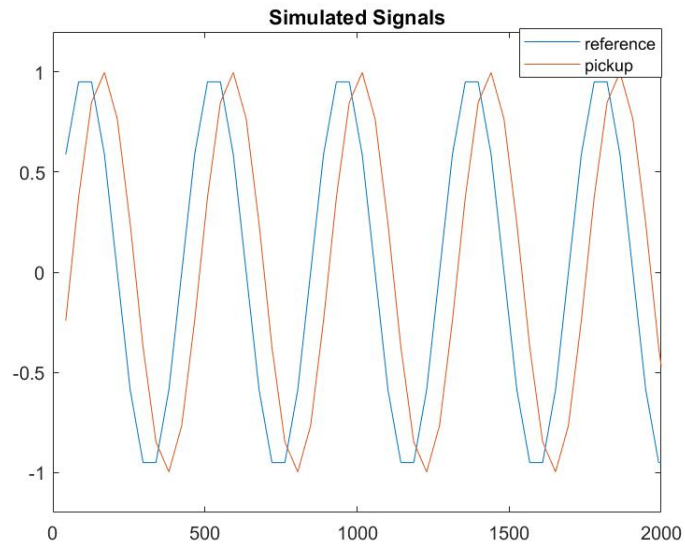


Figure 60: Simulated Signals, Displacement Point 41

D-3: MATLAB Data Processing Scripts

Contents

- [Script for cross-correlation based EPS demodulation](#)
- [Constants](#)
- [Main Code](#)
- [Correlation](#)
- [Analysis Plots](#)

Script for cross-correlation based EPS demodulation

DataF.mat has 250000 data points per position. Each point of data corresponds to 1mm of motion beginning at -5mm from center to 5mm from center ranging 10mm total, resulting in 11 data points. Output is relative amplitude and phase for each 11 displacement points.

```
clc;
clear;
close all;
```

Constants

```
EPSfreq=2.36e3;           %EPS carrier
rate=50000;              %LabView Sample Clock Rate = rate
dt=1/rate;
points=250000;          %Total number of samples
```

Main Code

```
load('DataF.mat');

for i=1:points
    t(i)=i*dt;
    %Time in microseconds
    tm(i)=t(i)*1e6;
end

%Position loop: 11 positions
for j=1:11
```

```
%Output from filter (pickup signal)
out(:,j) = DataF((j-1)*points+1:(j*(points)), 1);
%Reference from excitation coils
c(:,j) = DataF((j-1)*points+1:(j*(points)), 2);

%Function detrend removes data bias by computing mean values and subtrancing them from signals
%Use if DAQ system puts offset on signals
out_d(:,j) = detrend(out(:,j));
c_d(:,j) = detrend(c(:,j));
```

Correlation

```

%Auto correlation reference
Rx(j) = (1/points)*sum(c_d(:,j).^2);
%Auto correlation measured
Ry(j) = (1/points)*sum(out_d(:,j).^2);
%Amplitude of reference
Ax(j) = sqrt(2*Rx(j));
%Amplitude of measured
Ay(j) = sqrt(2*Ry(j));
%Cross-correlation
R(j) = (1/points)*sum(c_d(:,j).*out_d(:,j));

%Relative phase in radians
relphase(j)=acos(R(j)/sqrt(Rx(j)*Ry(j)));
%Amplitude ratio
relamp(j) = Ay(j)/Ax(j);

```

```

end
%Relative phase sign adjust loop: since acos only outputs angles from 0 to 180: last four
%values (all less than value at position 7) should be in Q4, not Q1
for k=1:11
    if relphase(k) < relphase(7)
        relphase(k) = 2*pi - relphase(k);
    end
end
%Background signal subtraction loop
for j = 1:11
    %Subtract background signal in complex plane
    V1(j)= relamp(j)*exp(1i*(relphase(j)));
    %Background signal
    V2 = -0.00005332875 + 0.0030995i;
    V3(j) = V1(j)-V2;
    %Update relamp values once background signal subtracted
    relamp(j) = abs(V3(j));
    relphase(j) = angle(V3(j))*180/pi;
end
%Relative amplitude sign correction loop: negate first 5 values
for k = 1:11
    if relamp(k)<6)
        relamp(k) = -relamp(k);
    end
end
end

```

Analysis Plots

```

figure
plot([-5:1:5],relamp,'*-');
grid
title('Pickup Relative Amplitude')
xlabel('Displacement')
ylabel('Derived relative amplitude')

figure
plot([-5:1:5],relphase,'*-');grid
title('Pickup Relative Phase')
xlabel('Displacement')
ylabel('Phase difference')

```


Contents

- [Background Signal Extraction](#)
- [Pickup signal subtraction](#)

Background Signal Extraction

This script constructs series of complex numbers using relamp and relphase. A line is fit through the complex points plotted.

```

clc; clear all; close all;

%Use relamp and relphase values from raw data processing
relamp = [0.0161,0.0135,0.0107,0.0081,0.0054,0.0031,0.0024,0.0042,0.0067,0.0094,0.0119];
relphase = [2.259,2.235,2.186,2.116,1.969,1.588,0.583,-0.2353,-0.414,-0.506,-0.542];

%Construct complex numbers
for k = 1:11
    V(k)= relamp(k)*exp(1i*(relphase(k)));
end

plot(real(V),imag(V));
hold on
plot(V,'o');
grid on
xlabel('Real Axis');
ylabel('Imaginary Axis');
title('Uncorrected Relative Amplitude, Complex Representation');

%Find vector of centerpoint from origin
%This data set V(6) = -0.00005332875 + 0.0030995i

% cpoint = V(6);

for j = 1:11

```

Pickup signal subtraction

```

V1(j) = V(j);
V2 = -0.00005332875 + 0.0030995i;
V3(j)= V1(j)-V2;
Relamp(j) = abs(V3(j));

```

```

end

figure
plot(real(V3),imag(V3));
hold on
plot(V3,'o');
grid on
xlabel('Real Axis');
ylabel('Imaginary Axis');
title('Nulled Relative Amplitude (Background Removed), Complex Representation');

for k = 1:11;
    if relphase(k) > 1

```

```
        Relamp(k) = -Relamp(k);
    end
end

%Use basic fitting in figure Tools drop down menu, select linear fit
figure
plot(Relamp,'o');
grid on;
xlabel('Displacement Points');
ylabel('Derived Relative Amplitude');
title('Corrected Relative Amplitude');
legend('Relative Amplitude');
```

Published with MATLAB® R2021a

Contents

- Combined Error Sources
- Constants
- Main Code: Divided according to three error sources - Window length, Downsampling, and Quantization. Correlation analysis is then applied.
- Window Length
- Position loop: 11 positions
- Downsampling - `downsample(x,r)` divides sample rate by `r`
- Quantization - change # of bits in line 64 to adjust quantization level
- Correlation
- Analysis Plots

Combined Error Sources

Uses noisy 250000 sample signal, then decrease window length, then downsample, then quantize. To change window length, divide total points (line 19). To downsample, change `r` in line 56. To quantize, change bits in line 64.

```
clc; clear all; close all;
```

Constants

```
EPSfreq=2.37e3;           %EPS carrier
rate=50000;              %LabView data file Sample Clock rate = rate
dt=1/rate;
tpoints=250000;         %Total number of samples
```

Main Code: Divided according to three error sources - Window length, Downsampling, and Quantization. Correlation analysis is then applied.

```
load('DataF.mat');
```

Window Length

```
plotSize = 250;
points = tpoints/1000; %250, ~10 cycles
for j=1:11
    jInd = (j-1)*250000;

    % First Data Indices based on Sample size
    data2Ind1 = jInd +1;

    % Second Data Index based on Sample size
    data2Ind2 = jInd + points;

    % First Result Index
    res2Ind1 = (j-1)*points+1;

    % Second Result Index
    res2Ind2 = (j)*points;
```

```
fprintf('Second Loop:\nj = %d \nind = %d:%d\n\n', [j; data2Ind1; data2Ind2])

%Output from filter (pickup signal)
out2(1:points,j) = DataF(data2Ind1:data2Ind2, 1);

%Reference from excitation coils
c2(1:points,j) = DataF(data2Ind1:data2Ind2, 2);
end
```

Second Loop:

```
j = 1
ind = 1:250
```

Second Loop:

```
j = 2
ind = 250001:250250
```

Second Loop:

```
j = 3
ind = 500001:500250
```

Second Loop:

```
j = 4
ind = 750001:750250
```

Second Loop:

```
j = 5
ind = 1000001:1000250
```

Second Loop:

```
j = 6
ind = 1250001:1250250
```

Second Loop:

```
j = 7
ind = 1500001:1500250
```

Second Loop:

```
j = 8
ind = 1750001:1750250
```

Second Loop:

```
j = 9
ind = 2000001:2000250
```

Second Loop:

```
j = 10
ind = 2250001:2250250
```

Second Loop:

```
j = 11
ind = 2500001:2500250
```

Position loop: 11 positions

```
for j=1:11
```

```

%Output from filter (pickup signal)
out(:,j) = out2(1:points,j);
%Reference from excitation coils
c(:,j) = c2(1:points,j);

%Function detrend removes data bias by computing mean values and subtracting them from signals
out_d(:,j) = detrend(out(:,j));
c_d(:,j) = detrend(c(:,j));

```

Downsampling - `downsample(x,r)` divides sample rate by `r`

```

r = 2;
dout(:,j) = downsample(out_d(:,j),r);
dc(:,j) = downsample(c_d(:,j),r);

```

Quantization - change # of bits in line 64 to adjust quantization level

```

sig = dout(:,j);
%max(sig)
ampMax = 10;
bits = 12;
% Assuming that num levels is total for entire wave
nLevels = 2^bits;
% Length numPoints-1, to represent numPoint intervals
partition = linspace(-ampMax,ampMax,nLevels);
spaceW = 2*ampMax/nLevels;
% Length numPoints, one entry for each interval
codebook = [-ampMax-spaceW, partition];
% Quantize
[index,qTemp] = quantiz(sig,partition,codebook);
quants(:,j) = qTemp';

```

Correlation

```

%Autocorrelation of reference
Rx(j) = (1/points)*sum(dc(:,j).^2);
%Autocorrelation of measured
Ry(j) = (1/points)*sum(quants(:,j).^2);
%Amplitude of reference
Ax(j) = sqrt(2*Rx(j));
%Amplitude of measured
Ay(j) = sqrt(2*Ry(j));
%Cross-correlation
R(j) = (1/points)*sum(dc(:,j).*quants(:,j));

%Relative phase in radians
relphase(j)=acos(R(j)/sqrt(Rx(j)*Ry(j)));
%Amplitude ratio
relamp(j) = Ay(j)/Ax(j);

```

```

end
%Relative phase sign adjust loop: since acos only outputs angles from 0 to 180: last four
%values (all less than value at position 7) should be in Q4, not Q1
for k=1:11

```

```

    if relphase(k>7)
        relphase(k) = 2*pi - relphase(k);
    end
end
%Background signal subtraction loop
for j = 1:11
    %Subtract background signal in complex plane
    V1(j)= relamp(j)*exp(1i*(relphase(j)));
    %Background signal
    V2 = -0.00005332875 + 0.0030995i;
    V3(j) = V1(j)-V2;
    %Update relamp values once background signal subtracted
    relamp(j) = abs(V3(j));
    relphase(j) = angle(V3(j));
end
%Relative amplitude sign correction loop: negate first 5 values
for k = 1:11
    if relamp(k<6)
        relamp(k) = -relamp(k);
    end
end
end

```

Analysis Plots

```

figure
plot([-5:1:5],relamp,'-*');grid
xlabel('Displacement')
ylabel('Derived relative amplitude')
figure
plot([-5:1:5],relphase,'-*');grid
xlabel('Displacement')
ylabel('Phase difference')

```

Contents

- [Window averaging - cuts 250000 sample record into 20000 sample chunks for each of the 11 positions.](#)
- [Divide total record into 20000 sample records](#)
- [Constants](#)
- [Main Code - cycle through assigned chunks in lines 40-65](#)
- [Correlation](#)
- [Analysis Plots](#)

Window averaging - cuts 250000 sample record into 20000 sample chunks for each of the 11 positions.

Assign the size of the divided records as chunk (line 12). All divided record are assigned unique names in lines 43-68. Correlation processing proceeds manually, cycling through each pair of chunks each time.

```
clc; clear all; close all;

load('DataF.mat');
```

Divide total record into 20000 sample records

```
plotSize = 250;
chunk = 20000;
% 20,000 sample chunks * 12 = 240,000: leaves out last 10,000 samples out per position
for i = 1:12
    %loop through 11 positions
    for j=1:11
        jInd = (j-1)*250000;

        % First Data Indices based on chunk size
        data1Ind1 = jInd + chunk*(i-1)+1;

        % Second Data Index based on chunk size
        data1Ind2 = jInd + chunk*i;

        % First Result Index
        res1Ind1 = (i-1)*chunk+1;

        % Second Result Index
        res1Ind2 = (i)*chunk;

        fprintf('i = %d \nj = %d \nind = %d:%d\n\n', [i; j; data1Ind1; data1Ind2])

        %Output from filter (pickup signal)
        out1(res1Ind1:res1Ind2,j) = DataF(data1Ind1:data1Ind2, 1);

        %Reference from excitation coils
        c1(res1Ind1:res1Ind2,j) = DataF(data1Ind1:data1Ind2, 2);
    end
end
points = chunk;

%Assigning chunks: pickup signal
outc1 = out1((0)*points+1:(1)*points,:);
outc2 = out1((1)*points+1:(2)*points,:);
outc3 = out1((2)*points+1:(3)*points,:);
```

```
outc4 = out1((3)*points+1:(4)*points,:);
outc5 = out1((4)*points+1:(5)*points,:);
outc6 = out1((5)*points+1:(6)*points,:);
outc7 = out1((6)*points+1:(7)*points,:);
outc8 = out1((7)*points+1:(8)*points,:);
outc9 = out1((8)*points+1:(9)*points,:);
outc10 = out1((9)*points+1:(10)*points,:);
outc11 = out1((10)*points+1:(11)*points,:);
outc12 = out1((11)*points+1:(12)*points,:);
```

```
%Assigning chunks: reference signal
```

```
cc1 = c1((0)*points+1:(1)*points,:);
cc2 = c1((1)*points+1:(2)*points,:);
cc3 = c1((2)*points+1:(3)*points,:);
cc4 = c1((3)*points+1:(4)*points,:);
cc5 = c1((4)*points+1:(5)*points,:);
cc6 = c1((5)*points+1:(6)*points,:);
cc7 = c1((6)*points+1:(7)*points,:);
cc8 = c1((7)*points+1:(8)*points,:);
cc9 = c1((8)*points+1:(9)*points,:);
cc10 = c1((9)*points+1:(10)*points,:);
cc11 = c1((10)*points+1:(11)*points,:);
cc12 = c1((11)*points+1:(12)*points,:);
```

```
i = 1
j = 1
ind = 1:20000
```

```
i = 1
j = 2
ind = 250001:270000
```

```
i = 1
j = 3
ind = 500001:520000
```

```
i = 1
j = 4
ind = 750001:770000
```

```
i = 1
j = 5
ind = 1000001:1020000
```

```
i = 1
j = 6
ind = 1250001:1270000
```

```
i = 1
j = 7
ind = 1500001:1520000
```

```
i = 1
j = 8
ind = 1750001:1770000
```

```
i = 1
j = 9
ind = 2000001:2020000
```

```
i = 12
j = 11
ind = 2720001:2740000
```

Constants

```
EPSfreq=2.37e3;           %EPS carrier
rate=50000;              %LabView data file Sample Clock rate = rate
dt=1/rate;
tpoints=250000;         %Total number of samples
```

Main Code - cycle through assigned chunks in lines 40-65

```
for i=1:points
    t(i)=i*dt;
    tm(i)=t(i)*1e6;           %Time in microseconds
end
%Position loop: 11 positions
for j=1:11
```

```
%Output from filter (pickup signal): cycle through chunks in lines 43-54 (ie outc1, outc2, etc)
out(:,j) = outc12(:,j);
%Reference from excitation coils: cycle through chunks in lines 57-68 (ie cc1, cc2, etc)
c(:,j) = cc12(:,j);

%Function detrend removes data bias by computing mean values and subtrancing them from signals
%Use if DAQ system puts offset on signals
out_d(:,j) = detrend(out(:,j));
c_d(:,j) = detrend(c(:,j));
```

Correlation

```
%Auto correlation reference
Rx(j) = (1/points)*sum(c_d(:,j).^2);
%Auto correlation measured
Ry(j) = (1/points)*sum(out_d(:,j).^2);

%Amplitude of reference
Ax(j) = sqrt(2*Rx(j));
%Amplitude of measured
Ay(j) = sqrt(2*Ry(j));

%Cross-correlation
R(j) = (1/points)*sum(c_d(:,j).*out_d(:,j));
%Relative phase in degrees
relphase(j)=acos(R(j)/sqrt(Rx(j)*Ry(j)));

%Amplitude ratio
relamp(j) = Ay(j)/Ax(j);
```

```

end
%Relative phase sign adjust loop: since acos only outputs angles from 0 to 180: last four
%values (all less than value at position 7) should be in Q4, not Q1
for k=1:11
    if relphase(k>7)
        relphase(k) = 2*pi - relphase(k);
    end
end
% Background signal subtraction loop
for j = 1:11
    %Subtract background signal in complex plane
    V1(j)= relamp(j)*exp(1i*(relphase(j)));
    %Background signal
    V2 = -0.00005332875 + 0.0030995i;
    V3(j) = V1(j)-V2;
    %Update relamp values once background signal subtracted
    relamp(j) = abs(V3(j));
    %relphase in degrees
    relphase(j) = angle(V3(j))*180/pi;
end
% Relative amplitude sign correction loop: negate first 5 values
for k = 1:11
    if relamp(k<6)
        relamp(k) = -relamp(k);
    end
end
end

```

Analysis Plots

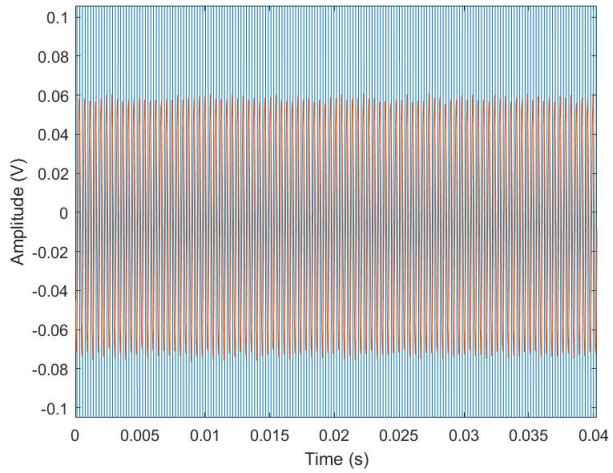
```

figure
plot([-5:1:5],relamp,'-*');grid
xlabel('Displacement')
ylabel('Derived relative amplitude')
figure
plot([-5:1:5],relphase,'-*');grid
xlabel('Displacement')
ylabel('Phase difference')

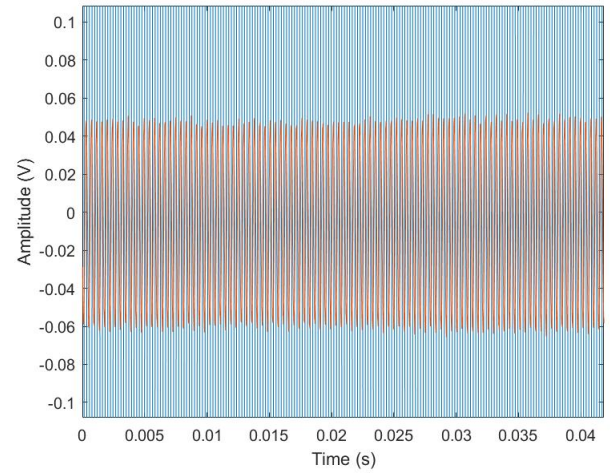
```


APPENDIX E - Graphs of Pickup Signals, Raw Data

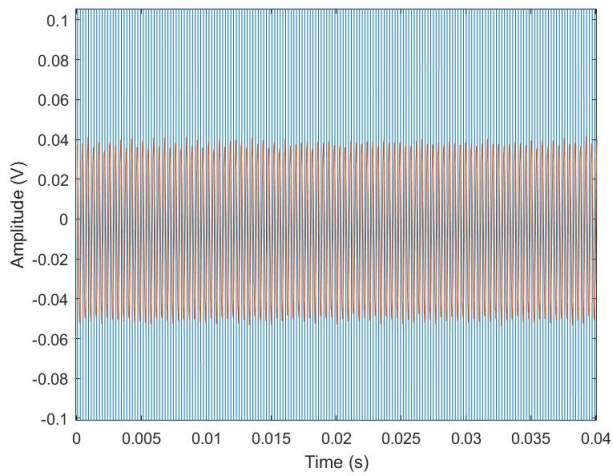
These graphs have been included to illustrate the slight negative offset put on the signals by the NI9205 data acquisition unit.



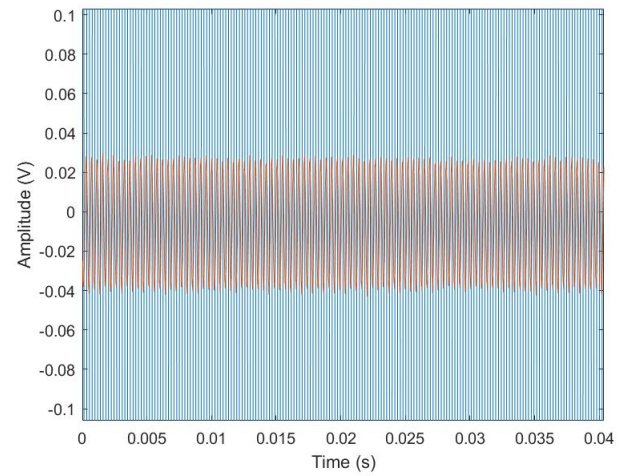
((a)) Core at -5 mm



((b)) Core at -4 mm

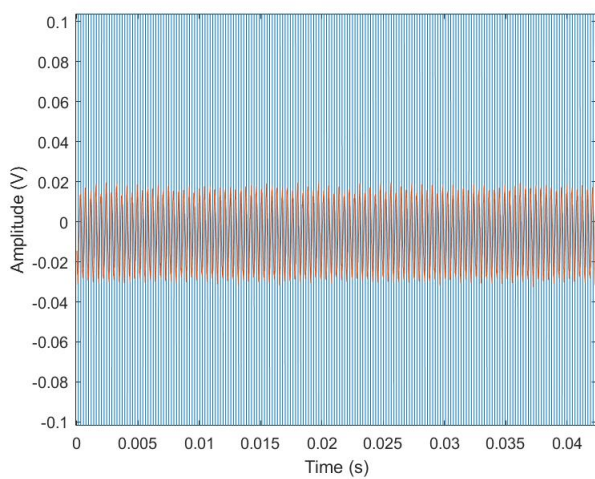


((c)) Core at -3 mm

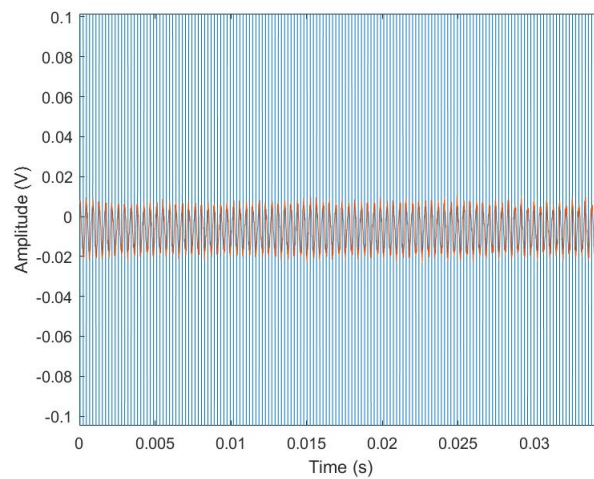


((d)) Core at -2 mm

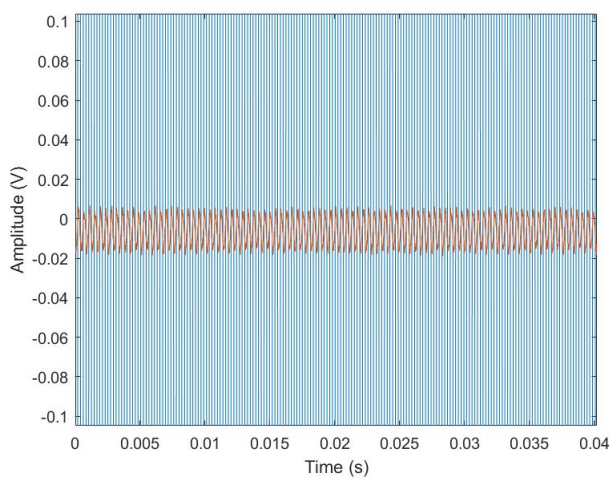
Figure 61: Raw Data Graphs, Displacement -5 to -2 mm



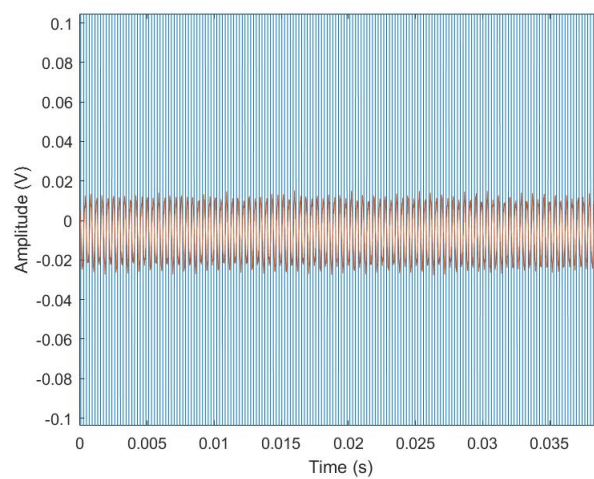
((e)) Core at -1 mm



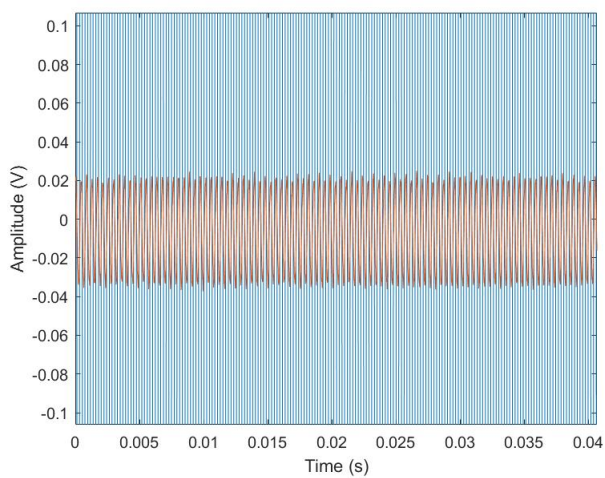
((f)) Core at 0 mm



((g)) Core at 1 mm

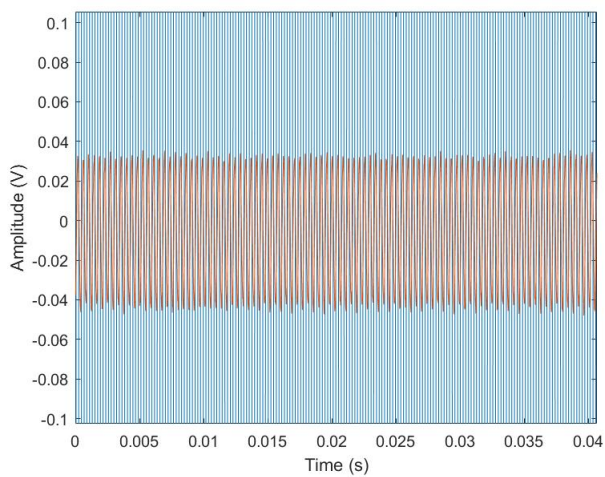


((h)) Core at 2 mm

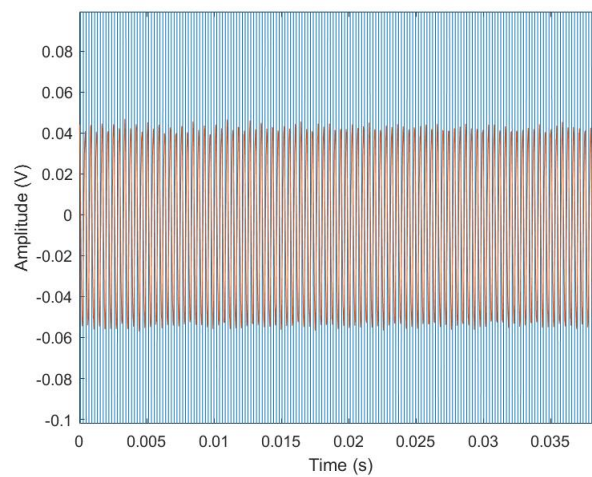


((i)) Core at 3 mm

Figure 61: Raw Data Graphs, Displacement -1 to 3 mm



((j)) Core at 4 mm



((k)) Core at 5 mm

Figure 61: Raw Data Graphs, Displacement 4 to 5 mm

APPENDIX F - Displacement Direction Relative Phase Difference

In this appendix, information about the model core displacing in the opposite direction is included. This data was not used for the final thesis because the record length was too short. However, it does illustrate that if the model core displaces from right to left (thinner end of EPS to connector end), the phase difference values will be between 0 and 90 degrees when the core is moving towards the center of the EPS and the values will be between 90 and 180 degrees when the model core is moving away from the center of the EPS. In this way it is shown to be consistent with the original formulation in Chapter 1.1 and Reference 7 if the motion is in this direction.

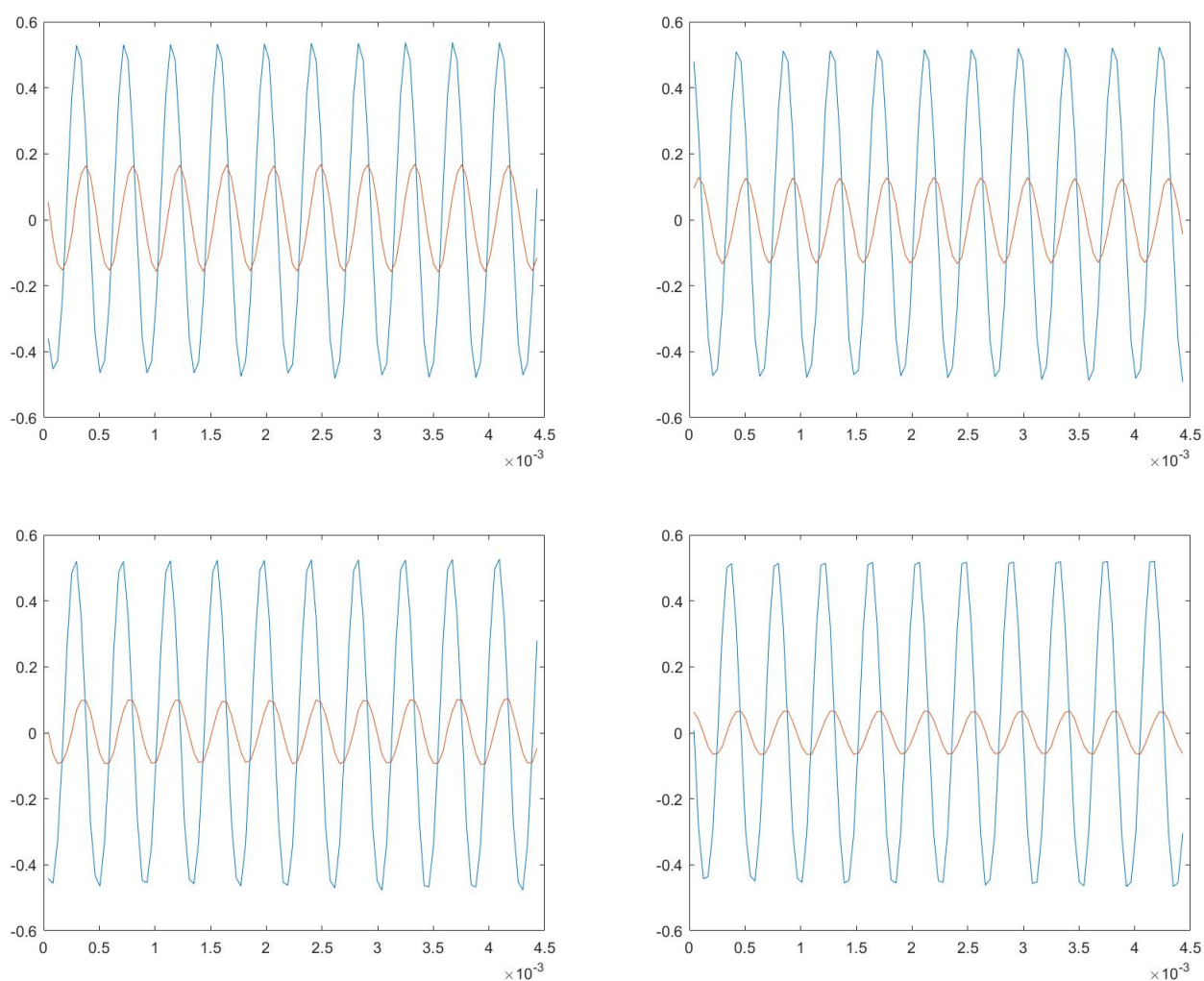


Figure 62: Displacement Right to Left, -5 to -2 mm

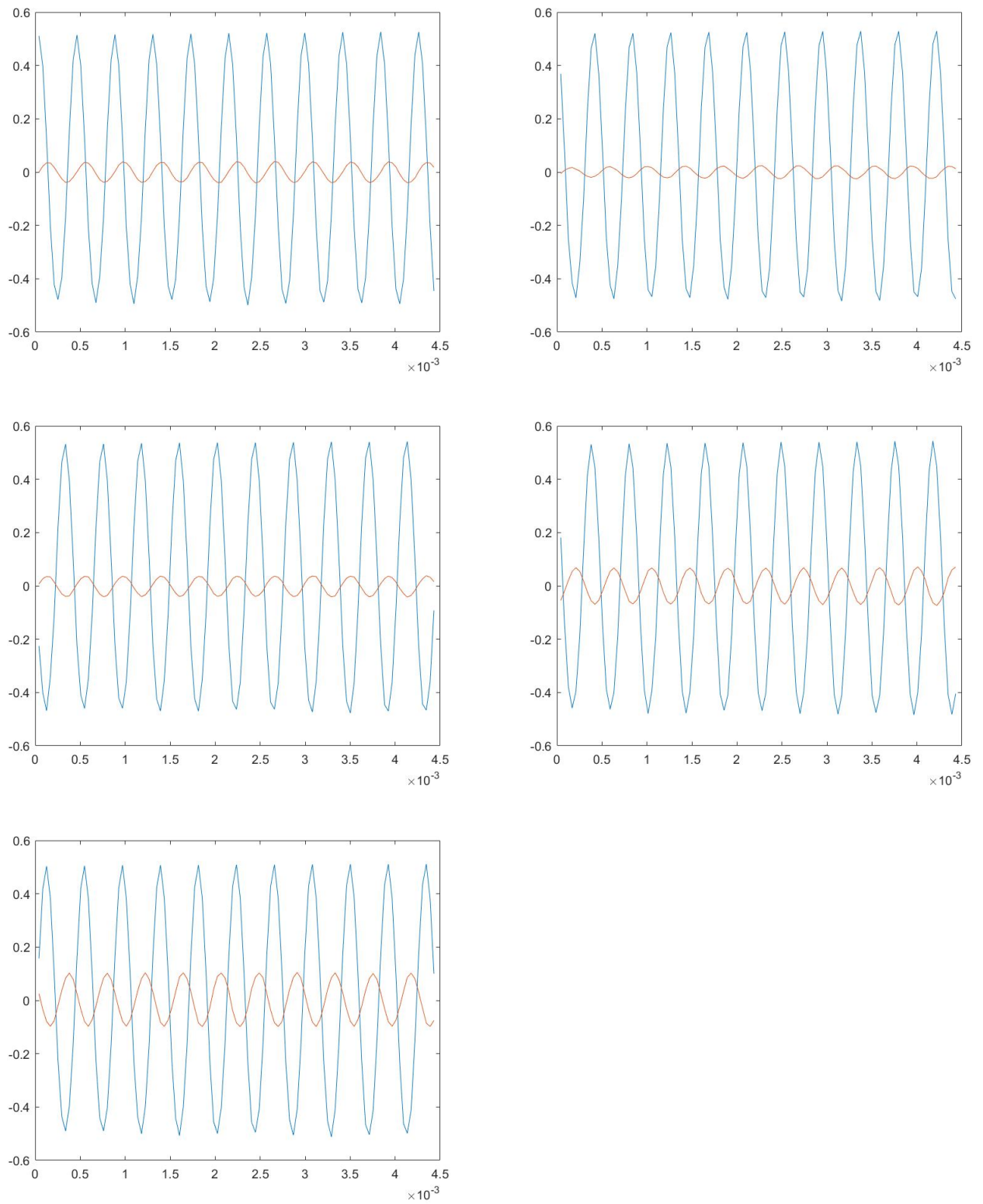


Figure 62: Displacement Right to Left, -1 to 3 mm

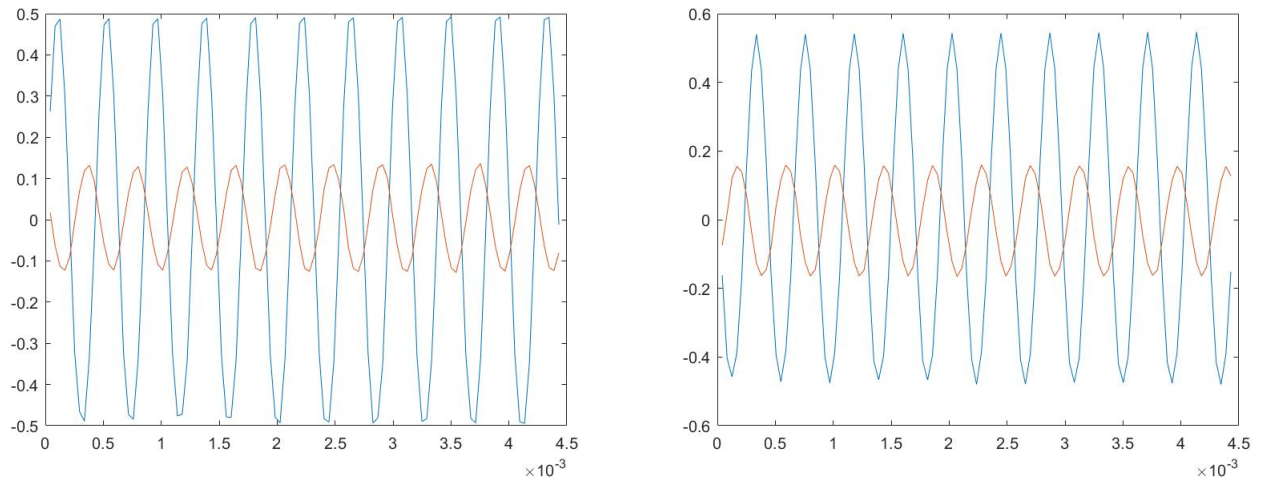


Figure 62: Displacement Right to Left, 4 to 5 mm

Table of Relative Amplitude and Phase Difference Values:

Displacement (mm)	Relative Amplitude	Phase Difference (Degrees)
-5	0.3128	58.9987
-4	0.2503	61.3886
-3	0.1937	63.0399
-2	0.1312	70.4944
-1	0.0759	86.4654
0	0.0447	142.2705
1	0.0758	165.7061
2	0.1340	148.0255
3	0.1947	142.7048
4	0.2531	139.3271
5	0.3135	138.3951

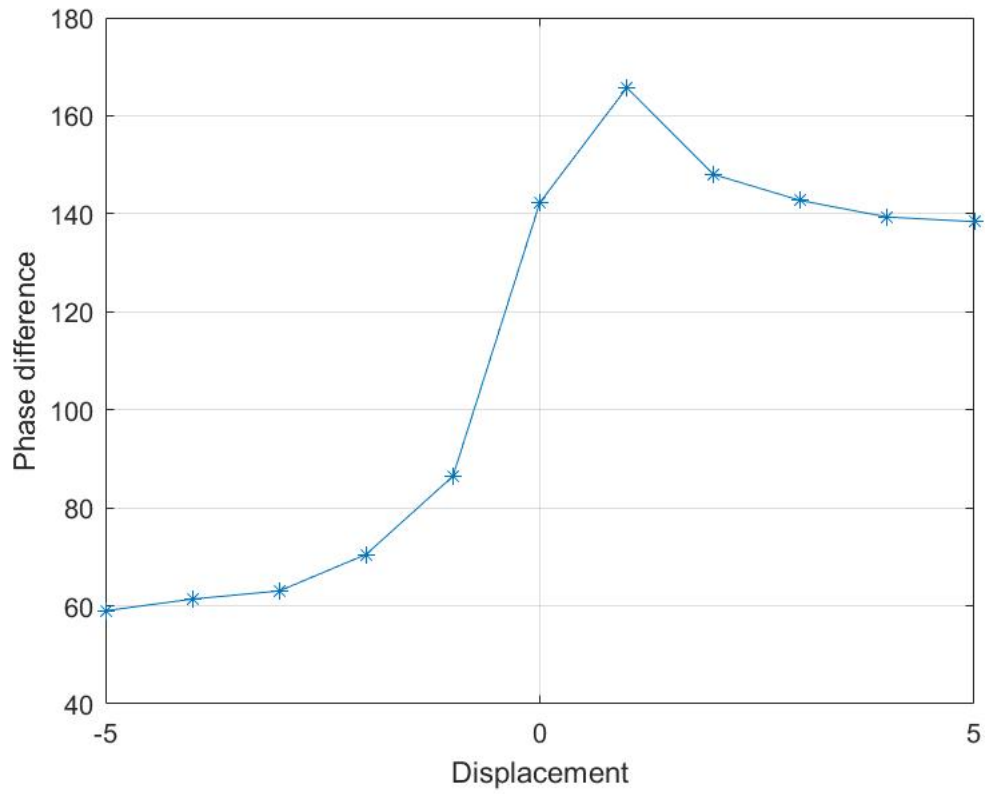


Figure 63: Relative Phase

After negating the first 5 data points, the relative amplitude graph below is the result.

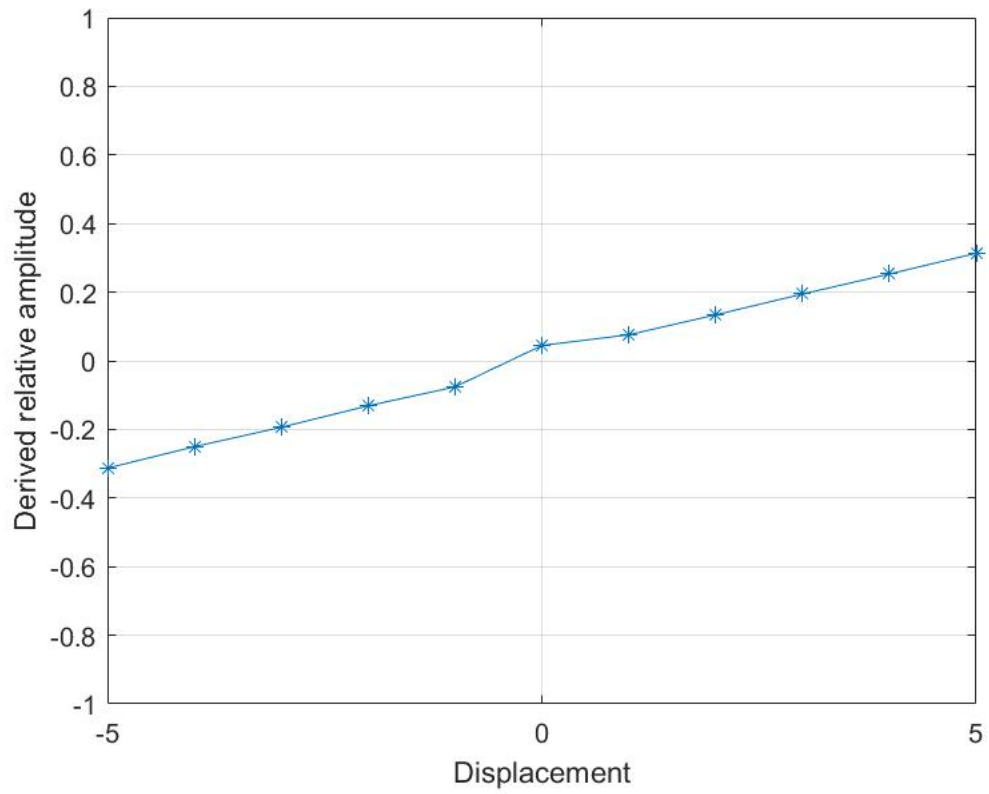


Figure 64: Relative Amplitude

VITA

Michelle Weinmann

811 Shirley Ave, Apt A5
 Norfolk, VA 23517
 Mobile: 631-276-2911
 Email: mwein006@odu.edu

Work Experience

NASA Langley **01/2020 - 12/2020**
 1 Nasa Dr
 Hampton, VA 23666

Semester Intern Hours per week: 40

Duties, Accomplishments, and Related Skills:

As an intern for the Structural Dynamics Branch working on composite booms for small satellite solar sail applications, my primary task is to set up a data acquisition system for a parabolic flight test. The test is being conducted in conjunction with DLR, a German aerospace center. In order to measure the deflection and deformation of the composite booms during the zero gravity test, a stereo system of multiple cameras is necessary to acquire visual data from multiple perspectives at the same time. Each camera must operate synchronously. LabVIEW and Aramis software is used for control of the cameras. Also assisted mentor with proposal related activities for the multifunctional deployable tower for payloads. This internship has been mostly carried out during the Covid-19 pandemic, requiring testing to be carried out from home. With much difficulty, camera set up in the home was achieved.
 Supervisor: Juan Fernandez (757-864-8876)

NASA Langley **06/2019 - 08/2019**
 1 Nasa Dr
 Hampton, VA 23666

Summer Intern Hours per week: 40

Duties, Accomplishments, and Related Skills:

I interned for the Atmospheric Flight and Entry Systems Branch working on a one and two degree model of the Magnetic Suspension and Balance System with an Electronic Position Sensor. In order to perform moment of inertia tests, an existing one degree of freedom magnetic levitation system must be tested for operation with an existing one degree of freedom position sensor I constructed for my Master's thesis. After successful integration, a second degree of freedom will be added using CAD designs and 3D printing methods. MATLAB Simulink software is used for control of the system. This project is collaborative in nature with another NIFS summer intern and a PhD student on the research team with myself leading the operations.
 Supervisor: Mark Schoenenberger (757-864-1833)

ODU Research Foundation & NASA Langley **01/2017 - 06/2019**
 4111 Monarch Way, Suite 204 1 Nasa Dr
 Norfolk, VA 23508 Hampton, VA 23666

Graduate Research Assistant Hours per week: 20

Duties, Accomplishments, and Related Skills:

Working on a one degree of freedom model of the Magnetic Suspension and Balance System's Electromagnetic Position Sensor housed at NASA Langley. Most work has been carried out at NASA

Langley, involves circuit design, fabrication, and signal processing, additionally using software such as MATLAB, LabView, and Comsol.
Supervisor: Dr. Colin Britcher (757-683-4916)

ODU Research Foundation

01/2015 - 12/2016

4111 Monarch Way, Suite 204
Norfolk, VA 23508

Undergraduate Research Assistant

Hours per week: 15

Duties, Accomplishments, and Related Skills:

Building a one degree of freedom model of the Magnetic Suspension and Balance System's Electromagnetic Position Sensor that was then housed at ODU, now housed at NASA. Introductory tasks included conditioning circuit design and fabrication and coil construction.
Supervisor: Dr. Colin Britcher (757-683-4916)

Education

Old Dominion University Norfolk, VA United States
Currently enrolled in Master of Science, expected graduation May 2021
GPA: 3.88 of a maximum 4.0
Credits Earned: 33 Semester hours
Major: Aerospace Engineering

Old Dominion University Norfolk, VA United States
Some College Coursework Completed 12/2016
GPA: 3.96 of a maximum 4.0
Credits Earned: 50 Semester hours
Major: Mechanical Engineering, Minor: Aerospace Engineering

St. John's College Annapolis, MD United States
Bachelor's Degree 05/2012
GPA: 3.4 of a maximum 4.0
Major: Liberal Arts

Professional Publications

A Digital One Degree of Freedom Model of an Electromagnetic Position Sensor
Submission for AIAA SciTech 2021
Michelle Weinmann
Department of Mechanical and Aerospace Engineering
Old Dominion University

Further Development of an Electromagnetic Position Sensor for a Wind Tunnel MSBS
Submission for ICFD 2018
Dr. Colin P Britcher, Michelle Weinmann
Department of Mechanical and Aerospace Engineering
Old Dominion University
Timothy Schott, Mark Schoenenberger
NASA Langley Research Center

A Multi-University Small Satellite Design Course: Systems Engineering Approach

Submission for AIAA Aviation and Aeronautics Forum and Exposition 2019
Shelly Mann, Michelle Weinmann
Department of Mechanical and Aerospace Engineering
Old Dominion University
Edmundson Effort
North Carolina A&T State University